Protecting Privacy for Group Nearest Neighbor Queries with Crowdsourced Data and Computing

Tanzima Hashem Mohammed Eunus Ali
Department of CSE, BUET
Dhaka 1000, Bangladesh
{tanzimahashem,eunus}@cse.buet.ac.bd

Lars Kulik Egemen Tanin Anthony Quattrone
Department of CIS, University of Melbourne
VIC 3010, Australia
{lkulik,etanin,anthony.quattrone}@unimelb.edu.au

ABSTRACT

User privacy in location-based services (LBSs) has become an important research area. We introduce a new direction to protect user privacy that evaluates LBSs with crowdsourced data and computation and eliminates the role of a location-based service provider. We focus on the group nearest neighbor (GNN) query that allows a group to meet at their nearest point of interest such as a restaurant that minimizes the total or maximum distance of the group. We develop a crowdsource-based approach, called PrivateMeetUp, to evaluate GNN queries in a privacy preserving manner and implement a working prototype of PrivateMeetUp.

Author Keywords
Group nearest neighbor queries; privacy; crowdsourcing

ACM Classification Keywords
H.2.8 Database Applications: Spatial databases and GIS.

INTRODUCTION

The increased adoption of location-based social networking services such as Facebook or Google+ facilitates a new class of applications that has *groups of users* instead of single users at its core. Location-based social networking services allow friends to remain connected virtually from anywhere at any time and to jointly enjoy location-based services (LBSs). For instance, a group of friends involved in a Facebook chat may decide to meet for a dinner at a restaurant that minimizes their overall trip time.

Users accessing LBSs reveal their locations to a location-based service provider (LSP) from which the LSP can identify a user's health, habits, and preferences, which may raise privacy concerns [10]. For example, if a user requests an LBS from a health clinic, then the LSP might identify the user's health condition if that health clinic specializes on certain medical procedures. Users may not trust LSPs and prefer to involve their peers instead of an unknown and impersonal LSP to answer their queries. Thus, we introduce a new concept to offering privacy preserving LBSs: crowdsourcing. In our approach, we eliminate the need for an LSP to answer of location based queries.

In general, an LSP has a large database of points of interests (POIs) such as restaurants and movie theaters. In the future, we envision that users may store their data and opinions about POIs on their local devices and share those with each other, in particular if they receive high levels of privacy in return. For example, users who enjoy lunch at a restaurant could store geocoded restaurant data on a GPS-enabled smartphone and make the data available to their peers for LBSs. In our approach both the data necessary to evaluate a location-based query and the actual processing of that query are crowdsourced. The role of an LSP replaced by users: the data set for an LBS is formed from local user data and a query answer is computed by splitting the evaluation tasks among the users in a privacy preserving manner.

We focus on an important class of LBSs, the group nearest neighbor (GNN) query [11], which enables a group of users to meet at a suitable place such as restaurant or movie theater. A GNN query returns the location of a POI that minimizes the aggregate distance (e.g., the total travel distance or maximum distance) of the group. We present a crowdsourced approach to evaluate the privacy preserving GNN queries without an LSP. In our approach, the data set for a GNN query is available through the shared POIs of every user in the group. One advantage of the group's POIs instead of the LSP's POIs is that a group may want to meet at their familiar and preferred places, a service an LSP cannot provide without requiring users to disclose their preferences. The computation of the GNN is divided among the crowd, i.e., the group members, to ensure user privacy.

In a straightforward approach to compute the aggregate distance of every POI without an LSP, every group member reveals their locations to other members. Although users might trust their peers, they may still prefer to keep their exact locations private. Consider a scenario, where a group of colleagues wants to meet for lunch at a restaurant but one of them is currently at a job interview and would like to keep her current location private. However, computing the group nearest POI while protecting each user's location privacy is a major challenge for processing GNN queries. To address this issue, we develop an approach called *PrivateMeetUp*. PrivateMeetUp evaluates GNN queries and enables users to control the degree of location precision disclosed to other group members. To evaluate our approach, we develop a prototype application in Android called PrivateMeetUp. In addition, we

UbiComp'13, September 8–12, 2013, Zurich, Switzerland.
Copyright © 2013 ACM 978-1-4503-1770-2/13/09...$15.00.
http://dx.doi.org/10.1145/2493432.2494524

run experiments to show the efficacy and efficiency of our approach in a simulated environment.

The applicability of our crowdsource-based approach is not limited to only group LBSs; it can be also applied to other LBSs such as finding a nearest POI for an individual as long as friends share their POIs with the individual. In such a case, an individual knows her own location and can easily compute the answer privately. Therefore, we focus on GNN queries, where it is hard to find the answer without knowing locations of others. Our privacy solution can be extended for other group LBSs such as group trip planning queries.

PROBLEM SETUP

In our system architecture the users are connected through the cellular network or the Internet. A user who initiates a MeetUp request to a set of connected users is called the *initiator*. The initiator and users who accept the initiator's request form a group and propose the data set locations for possible meeting places (POIs). A group nearest neighbor (GNN) query computes the POI from the data set (i.e., candidate POIs) that minimizes the aggregate distance of the group. We focus on the aggregate functions SUM and MAX, i.e., the group's total and maximum travel distance, respectively. We study the *k*GNN query, which returns *k* POIs that have *k* smallest aggregate distances. A group could select from the *k* POIs a single POI that is based on other criteria such as votes received, cost or preference. On the other hand, if it happens that a group has higher priority on preference of Italian food than the distance for selecting a restaurant then the group can request a more selective query, e.g., may ask for an Italian restaurant that minimize the group distance. In privacy preserving *k*GNN queries, all group members hide their exact locations from each other.

Adversary Model

In our approach, a *k*GNN query is evaluated with crowdsourced data and computation. We consider the users involved in processing *k*GNN queries as potential adversaries. We do not consider eavesdropping separately as adversarial behavior, because the maximum knowledge of an eavesdropper about a group members's location is never greater than the maximum knowledge of the other group members. We make following assumptions about our system:

- Adversaries do not have any background knowledge about user locations and do not know their distribution.
- Group members know each other's identity.
- We exclude wireless adhoc networks for communication as their limited range may refine user locations.

Threat Model

Evaluating *k* GNNs from a data set depends on the locations of all group members. For privacy preserving *k*GNN queries, users do not reveal their locations to others. Existing work [7] shows that even revealing a user's distances instead of exact locations to three or more POIs enables an adversary to compute the user's exact location using 2D trilateration. The solution proposed in [7] to avoid this the distance intersection attack does not apply in our scenario as it assumes an LSP. Thus, we develop an LSP-free solution that does not reveal users' distances to POIs and thus prevent the distance intersection attack.

Privacy Model

Researchers have developed different techniques [9] for user privacy in LBSs. The concept of *k*-anonymity [5] hides a user's identity and is not applicable as users reveal identities to peers. To protect a user's location privacy, existing approaches often reveal a user's imprecise location [6] to evaluate location-based queries. However, they use a fixed degree of imprecision (or inaccuracy) to hide a user's location. Since users do not know how much information they need to reveal, they may reveal more than required, which could unnecessarily reduce their location privacy. Cryptographic approaches [4] have also been proposed to access LBSs without disclosing user locations but they require complex infrastructures and incur high processing overhead [3].

In PrivateMeetUp, users disclose their imprecise distances to POIs to their peers to identify the POI with the minimum aggregate distance from the data set. The highest level of imprecision is imposed at the start of the evaluation and then gradually decreases as long as a user's privacy is not violated and *k* GNNs are determined. Thus, a user only reveals location information that is required for processing GNN queries. From the revealed imprecise distances to POIs, the user's location can be determined as an area. We measure a user's privacy level in terms of the *obfuscation level* [6], which is defined as the *percentage area of the total space* that can be inferred as the user's location from the user's revealed distance information. The larger the inferred area for a user's location, the higher is the level of privacy. We show that our approach ensures high obfuscation levels, i.e., a user's imprecise location includes a range of diversified places such as homes, restaurants, hospitals, which makes it difficult to use additional context information to pinpoint a user's location.

RELATED WORK

GNN queries have been introduced in [11]. In [7], the first privacy preserving approach for GNN queries was developed; the approach trusts the LSP to protect user privacy from other group members. Our approach protects user privacy without relying on an LSP. Recently, two cryptographic approaches [1] and [8] have been proposed to address privacy protection problem for GNN queries, but have the same limitations of all cryptographic methods. In recent years, a few approaches have been developed for processing LBSs with crowdsourcing [12]. These approaches neither focused on GNN queries nor addressed user privacy for LBSs. To the best of our knowledge, we develop the first approach using crowdsourcing to preserve user privacy.

PRIVATEMEETUP

Our key idea is to convert the 2-dimensional (2D) space in to an 1-dimensional (1D) imprecise distance space. An accurate distance space ranges from 0 to the maximum possible distance between any two locations in the 2D Euclidean space. To introduce imprecise distances, the distance space is iteratively divided into buckets, where each bucket describes a range of distances. A user only reveals the bucket in which the user's actual distance to a POI falls into. The smaller the number of buckets, the higher the imprecision imposed in the distance space and the higher is the level of user privacy. Starting from a high degree of imprecision, our approach iteratively reduces the degree of imprecision in the distance space until the group decides on their meeting place. The steps of PrivateMeetUp are summarized as follows:

1. The initiator sends a MeetUp request to a group of users.
2. Users who accept the initiator's request, propose their POIs as candidate meeting places.
3. The initiator computes the imprecise distance space and builds the data set from the received POIs.
4. The initiator and the members compute their distances for each POI given the current imprecise distance space and update the imprecise aggregate distances of each POI, if the users' privacy levels are respected.
5. On receiving the returned data set, the initiator refines the data set based on the aggregate imprecise distances, i.e., prunes the POIs that cannot be GNNs.
6. If the data set size equals k, the initiator announces the GNNs to the group. Otherwise, the initiator refines the distance space and repeats steps 4 to 6.

In PrivateMeetUp, a user's current privacy level is determined based on the maximum knowledge that an adversary can have about the user. The maximum knowledge of an adversary about a user is derived by combining all group members' knowledge from every iteration of the evaluation process. Thus, in our approach, a user has full control about their privacy levels. An important benefit of our approach is that the group does not need to stop the query execution if some users leave at different stages of the execution due to their privacy requirements. Our approach will still guarantee an optimal meeting place for the remaining subgroup without discarding the intermediate computation results.

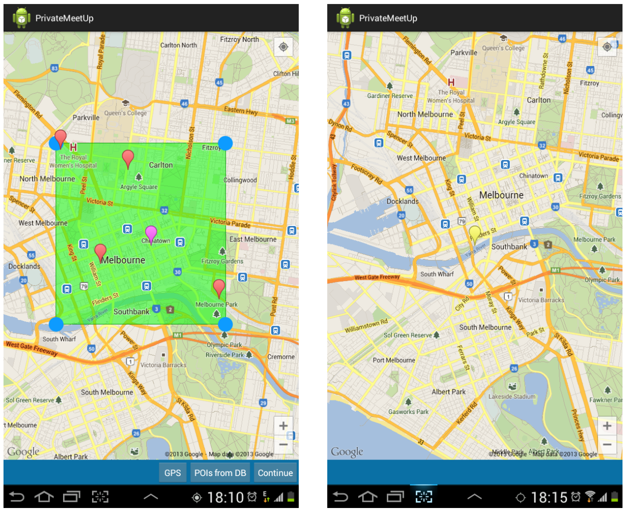
Security Analysis

Our system assumes a semi-honest model: every group member in PrivateMeetUp follows our algorithms but may try to violate other members' privacy based on the revealed member data during the query evaluation. A malicious group member may intentionally propose a POI set to maximize the number of iterations during the query evaluation and thus cause members to reveal more precise locations. However, identifying such a POI set is difficult as other member locations are unknown. More importantly, in our approach group members can leave the evaluation process any time if their privacy is violated, i.e., precise locations would be disclosed. This could affect the accuracy of the query answer slightly for the group members who left early.

EXPERIMENTS

PrivateMeetUp Prototype

We have developed a prototype, PrivateMeetUp as a proof of concept of our algorithm being applied in real world scenarios. We have evaluated PrivateMeetUp using *four* Samsung Galaxy Tab 2 Android devices connected to Internet through GPRS. We integrate PrivateMeetUp with Facebook and use the Extensible Messaging and Presence Protocol (XMPP) of the Facebook-Chat API for the communication purpose.



(a) POIs of a user (b) Result POI

Figure 1. PrivateMeetUp Prototype

After logging into PrivateMeetUp, a user can see her friend list from Facebook. To *initiate* a PrivateMeetUp gathering, a user invites a group of friends. Each group member responds to the invitation by sending a list of POIs and a rectangular area: the MBR (minimum bounding rectangle) of the suburbs in which the user and the POIs are located. A user can select POIs by pointing to different places on the map, or can outsource already visited or locally stored POIs on a user's mobile device. Figure 1(a) shows the screen of selected POIs with red marks and the MBR of a user.

The initiator's device forms a data set from the received POIs and computes a total space as an MBR that includes MBRs of all users. Then, the initiator's device computes an imprecise distance space and starts the process of computing k GNNs. Since every user participates in computing k GNNs, the data set is passed among users in a round-robin fashion (i.e., a user u_i forwards the data packet to the user $u_{i+1 \mod n}$, where n is the group size). After one complete iteration, the initiator's mobile device has the imprecise aggregate distance of all POIs and it applies the pruning technique to remove the POIs from the data set that can never be a part of k GNNs. The process continues until the system finds k GNNs. Finally, the initiator's device sends the k GNNs to all participants, and the notification is displayed on each user's screen (Figure 1(b)).

Experiments using PrivateMeetUp Prototype

We measure the performance of our approach in terms of the processing time, communication overhead and privacy level. The communication overhead is expressed as the number of iterations and the privacy level is determined as the average obfuscation level. We observe that for a group of 4 users, $k = 2$, and aggregate function SUM, when each user gives one POI, i.e., 4 POIs in total, it takes 10 iterations and 44.1 seconds to answer 2 GNNs. Similarly, when 4 users crowdsource 8 POIs, it takes 12 iterations and 88.6 seconds to find the answer. For 4 and 8 POIs, the average obfuscation level achieved by users are 0.55% and 0.25%, respectively, which are sufficient to protect a user's location privacy. For example, a small suburb in Melbourne is about 4 km^2, which is about 0.05% with respect to the total area of Melbourne.

The entire computing process runs as a background process on every participating mobile device. Thus, even if the whole process may take a couple of minutes to complete, the delay is acceptable for users as PrivateMeetUp does not prevent them from performing other activities during that period.

Experiments in Simulated Environment

In the simulation, we vary a wide range of parameters: group size as 2, 4, 8 and 16, data set size as 4, 8, 16 and 32, and answer set size k as 1,2,4, and 8. The default values for group size, data set size, and k are set to 8, 16, and 2, respectively. We use POIs of tourist attraction of Victoria as our POIs [13]. We run the experiments on a desktop with a Intel Core 2 Duo 2.40 GHz CPU and 4 GBytes RAM. We run each set of experiments for 25 samples of kGNN queries and measure the average performance in terms of number of iterations and obfuscation level. Since in a PC environment, the processing time does not reflect the actual processing time on a real device, we have not shown the processing time in experiments.

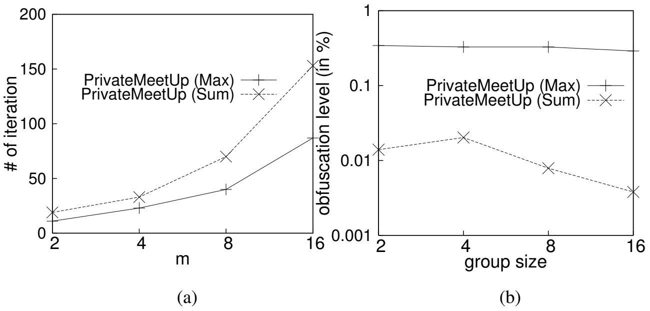


Figure 2. Effect of group size

Figures 2(a) and 2(b) show the required number of iterations and the average obfuscation level for varying group size, respectively. The number of iterations linearly increases with the increase of group size for both SUM and MAX, because a higher number of users in a group means more constraints on the selection of POIs. We also see that the achieved privacy level slightly decreases with the increase of group size. The average obfuscation level achieved by users for SUM and MAX are approximately .58% and 0.032%, respectively.

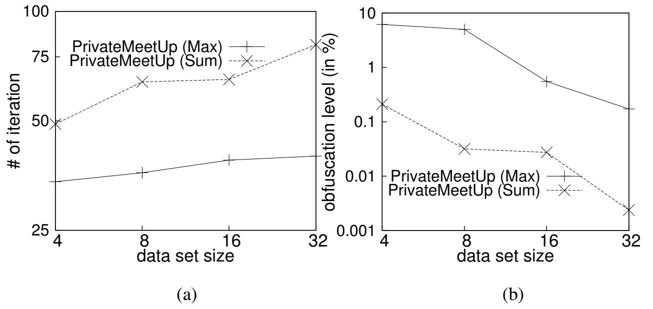


Figure 3. Effect of data set size

Figure 3(a) shows that the iteration increases with the increased number of POIs. For a large data set, the algorithm needs to prune more POIs to find the answer and thus requires increased number of iterations. With the increase of the number of POIs, a user needs to reveal higher number of imprecise distances and thus, the user's location could be more precisely identified (Figure 3(b)).

We also observe that the number of iterations increases and the average obfuscation level of users decreases with the increase of k (not shown). Since the number of iteration increases for a large k, the imprecision of the distance space decreases and users need to reveal precise distances to POIs.

CONCLUSION

We have introduced a new concept to evaluate LBSs with crowdsourced data and computation: PrivateMeetUp, which efficiently evaluates k group nearest neighbor (kGNN) queries in a privacy preserving manner without the need for an LSP. We have built a prototype of PrivateMeetUp based on the Android platform to show the applicability of our approach in real environments. Experimental results show that our approach is scalable and can ensure high level of privacy for large group sizes. Though we have focused on Euclidean space, our approach can be also adopted for road networks.

ACKNOWLEDGEMENTS

This research was supported under Australian Research Councils Discovery Projects funding scheme (project number DP110100757).

REFERENCES

1. Ashouri-Talouki, M., Baraani-Dastjerdi, A., and Seluk, A. A. Glp: A cryptographic approach for group location privacy. *Computer Communications 35*, 12 (2012), 1527–1533.
2. Ghinita, G. Private queries and trajectory anonymization: a dual perspective on location privacy. *TDP 2*, 1 (2009), 3–19.
3. Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., and Tan, K.-L. Private queries in location based services: anonymizers are not necessary. In *SIGMOD* (2008), 121–132.
4. Gruteser, M., and Grunwald, D. Anonymous usage of location-based services through spatial and temporal cloaking. In *MobiSys* (2003), 31–42.
5. Hashem, T., and Kulik, L. Safeguarding location privacy in wireless ad-hoc networks. In *Ubicomp* (2007), 372–390.
6. Hashem, T., Kulik, L., and Zhang, R. Privacy preserving group nearest neighbor queries. In *EDBT* (2010), 489–500.
7. Huang, Y., and Vishwanathan, R. Privacy preserving group nearest neighbour queries in location-based services using cryptographic techniques. In *GLOBECOM* (2010), 1–5.
8. Krumm, J. A survey of computational location privacy. *PUC 13* (2009), 391–399.
9. Microsoft. Location & privacy: Where are we headed?, (accessed Sept 2, 2011). **http://www.microsoft.com/privacy/dpd**.
10. Papadias, D., Shen, Q., Tao, Y., and Mouratidis, K. Group nearest neighbor queries. In *ICDE* (2004), 301–310.
11. Shankar, P., Huang, Y.-W., Castro, P., Nath, B., and Iftode, L. Crowds replace experts: Building better location-based services using mobile social network interactions. In *PerCom* (2012), 20–29.
12. TouristPOIs. **http://downloads.cloudmade.com/oceania/australia_and_new_zealand/australia#downloads_breadcrumbs**.