Middle-East Journal of Scientific Research 20 (4): 527-531, 2014
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2014.20.04.12040

Mining Continuously Moving Objects with PAM Framework Using K Nearest Neighbor Algorithm

[1]T. Judes Divya and [2]S. Vinoth Lakshmi

[1]Department of Information and Technology, Tagore Engineering College, Bharath University, India
[2]Department of Information and Technology, Anna University, Bharath University, India

Abstract: This paper is to monitor continuously moving objects location and minimize the number of location updates. This paper proposes a privacy-aware monitoring (PAM) framework that addresses both issues efficiency and privacy which are two fundamental issues in moving object monitoring. As for efficiency, PAM framework minimizes the number of location updates. As for privacy, the accuracy of location updates determines how much the client's privacy is exposed to the server. We develop efficient kNN (k Nearest Neighbor algorithm) in the PAM framework. The experimental results show that PAM substantially outperforms traditional schemes in terms of monitoring accuracy, CPU cost and scalability while achieving close-to-optimal communication cost.

Key words: Privacy Aware Monitoring · k Nearest Neighbor

INTRODUCTION

The PAM framework is to monitor continuously moving objects location and minimize the number of location updates. Efficiency and privacy are two fundamental issues in moving object monitoring [1]. This paper proposes a privacy-aware monitoring (PAM) framework that addresses both issues. The framework distinguishes itself from the existing work by being the first to holistically address the issues of location updating in terms of monitoring accuracy, efficiency and privacy, in particular when and how mobile clients should send location updates to the server.

Based on the notions of safe region and most probable result, PAM performs location updates only when they would likely alter the query results. Furthermore, by designing various client update strategies, the framework is flexible and able to optimize accuracy, privacy or efficiency.

The fundamental problem in a monitoring system is when and how a mobile client should send location updates to the server because it determines three principal performance measures of monitoring-accuracy, efficiency and privacy. Accuracy means how often the monitored results are correct and it heavily depends on the frequency and accuracy of location updates. As for efficiency, two dominant costs are: the wireless communication cost for location updates and the query evaluation cost at the database server, both of which depend on the frequency of location updates. As for privacy, the accuracy of location updates determines how much the client's privacy is exposed to the server. The existing system focused on static dataset. So for dynamic updation of data we go for PAM framework.

The efficient kNN (k Nearest Neighbor algorithm) in the PAM framework is developed. The experimental results show that PAM substantially outperforms traditional schemes in terms of monitoring accuracy, CPU cost and scalability while achieving close-to optimal communication cost. Henceforth the project is simulated using relational database instead of spatial database.

Related Work: In order to carry out this project several references and white papers are referred, from which many valuable information are identified. The following sections provide those information.

Corrseponding Author: T. Judes Divya, Department of Information and Technology, Tagore Engineering College, Bharath University, India.

Indexing the Positions of Continuously Moving Objects: On tracking the changing positions of objects capable of continuous movement is becoming increasingly feasible and necessary. R tree based indexing technique [2] that supports the efficient querying of the current and projected future positions of such moving objects. The technique is capable of indexing objects moving in one-two-and three-dimensional space. Update algorithms enable the index to accommodate a dynamic data set, where objects may appear and disappear and where changes occur in the anticipated positions of existing objects. It contains static dataset and affects update performance.

Anonymous Usage of Location-Based Services: Location cloaking was proposed to blur the exact client positions into bounding boxes. By assuming a centralized and trustworthy third-party server that store all exact client positions, various location cloaking algorithms [3] were proposed to build the bounding boxes while achieving the privacy measure such as k-anonymity. However, the use of bounding boxes makes the query results no longer unique. As such, query evaluation in such uncertain space is more complicated.

Monitoring Continuous Spatial Queries over Moving Objects: A monitoring framework where the clients are aware of the spatial queries being monitored, so they send location updates only when the results for some queries might change. Our basic idea is to maintain a rectangular area, called safe region, for each object. The safe region is computed based on the queries in such a way that the current results of all queries remain valid as long as all objects reside inside their respective safe regions. A client updates its location on the server only when the client moves out of its safe region. This significantly improves the monitoring efficiency and accuracy compared to the periodic or deviation update methods. However, this framework [4] fails to address the privacy issue, that is, it only addresses "when" but not "how" the location updates are sent.

Monitoring k-Nearest Neighbor Queries over Moving Objects: The constant monitoring of kNN queries over moving objects within geographic area [5]. Existing approaches to this problem have focused on predictive queries and relied on the assumption that the trajectories of the objects are fully predictable at query processing time. We relax this assumption and propose two efficient and scalable algorithms using grid indices. One is based on indexing objects and the other on queries. For each approach, a cost model is developed and a detailed analysis along with the respective applicability is presented. The Object-Indexing approach is further extended to multi-levels to handle skewed data. We found the problem of monitoring k-NN queries over moving objects. We presented an analysis of two proposed approaches namely, query-indexing and object indexing. We have validated the results of our analysis, presented extensions of the basic methods to handle non-uniform data efficiently and conducted a variety of experiments to explore the benefits of our approach in a variety of parameter settings.

Anonymous Location-Based Queries in Distributed Mobile Systems: To protect location privacy, various anonymizing techniques [6] have been proposed to hide client actual location. The anonymizer must be updated continuously with the current locations of all users. Moreover, the complete knowledge of the entire system poses a security threat, if the anonymizer is compromised. Existing approaches may fail to provide spatial anonymity for some distributions of user locations and describe a novel technique which solves this problem. We propose prive a decentralized architecture for preserving the anonymity of users issuing spatial queries to LBS.

Protecting Location Privacy with Personalised k-Anonymity Architecture: A scalable architecture for protecting the location privacy from various privacy threats resulting from uncontrolled usage of LBSs [7]. This architecture includes the development of a personalized location anonymization model and a suite of location perturbation algorithms. A unique characteristic of our location privacy architecture is the use of a flexible privacy personalization framework to support location k-anonymity for a wide range of mobile clients with context-sensitive privacy requirements. This framework enables each mobile client to specify the minimum level of anonymity that it desires and the maximum temporal and spatial tolerances that it is willing to accept when requesting k-anonymity-preserving LBSs. The personalized location k-anonymity model, together with our location perturbation engine, can achieve high resilience to location privacy threats without introducing any significant performance penalty.

Algorithm

kNN (k Nearest Neighbor) Algorithm: Object is classified by a majority vote of its neighbors. Here if we don't know mobile client information we use nearest neighbor client information to access it. Assign weights to the neighbors based on their 'distance' from the query point. Weight 'may' be inverse square of the distances. An arbitrary instance is represented by (a1(x), a2(x), a3(x),.., an(x)) ai(x) denotes features. Euclidean distance between two instances d(xi, xj)=sqrt (sum for r=1 to n (ar(xi) - ar(xj))2). Continuous valued target function mean value of the k nearest training examples.

The k Nearest Neighbor Rule (kNN) is a very intuitive method that classifies unlabeled examples based on their similarity to examples in the training set. For a given unlabeled example xu RD, find the k "closest" labeled examples in the training data set and assign xu to the class that appears most frequently within the k-subset. The kNN only requires an integer, A set of labeled examples (training data), a metric to measure "closeness". The algorithm is analytically tractable, simple implementation and it uses local information which can yield highly adaptive behaviour [8].

Design: In the existing system, clients are always willing to provide their exact positions to the server so the privacy issue is simply ignored. For example, knowing that a user is inside a heart specialty clinic during business hours, the adversary can infer that the user might have a heart problem. This has been cited as a major privacy threat in location-based services. In the existing system, privacy is not maintained. Location resolution increases ie maximum no of location updates are possible, so searching is tedious. Large amount of information will be stored in database server.

In the proposed system, clients do not want to expose their genuine point locations to the database server to avoid spatiotemporal correlation inference attack, by which an adversary may infer users' private information such as political affiliations, alternative lifestyles, or medical problems. This proposes a PAM framework that addresses both issues efficiency and privacy. Compute safe region for each object. The PAM framework in which privacy is maintained and minimizes the number of location updates.

In PAM framework where the clients are aware of the spatial queries being monitored, so they send location updates only when the results for some queries might change. Our basic idea is to maintain a rectangular area, called safe region, for each object. The safe region is computed based on the queries in such a way that the current results of all queries remain valid as long as all objects reside inside their respective safe regions. A client updates its location on the server only when the client moves out of its safe region. This significantly improves the monitoring efficiency and accuracy compared to the periodic or deviation update methods. However, this framework fails to address the privacy issue, that is, it only addresses "when" but not "how" the location updates are sent. This project constructs PAM architecture using relational database instead of using spatial queries.

In this, we take a more comprehensive approach-instead of dealing with "when" and "how" separately like most existing work, we propose a privacy-aware monitoring (PAM) framework that incorporates the accuracy, efficiency and privacy issues altogether. We adapt for the monitoring environment the privacy model that has been employed by location cloaking and other privacy-aware approaches. More specifically, a client encapsulates its exact position in a bounding box and the timing and mechanism with which the box is updated to the server are decided by a client-side location updater as part of PAM. However, the integration of privacy into the monitoring framework poses challenges to the design of PAM.

To our knowledge, this is the first comprehensive framework that addresses the issue of location updating holistically with monitoring accuracy, efficiency and privacy altogether. As for efficiency, the framework significantly reduces location updates to only when an object is moving out of the safe region and thus, is very likely to alter the query results. As for accuracy, the framework offers correct monitoring results at any time, as opposed to only at the time instances of updates in systems that are based on periodic or deviation location update. The framework is generic in the sense that it is not designed for a specific query type. Rather, it provides a common interface for monitoring various types of spatial queries such as range queries and kNN queries. Moreover, the framework does not presume any mobility pattern on moving objects. The framework is flexible in that by designing appropriate location update strategies, accuracy, privacy, or efficiency can be optimized. The proposed system has the following modules.

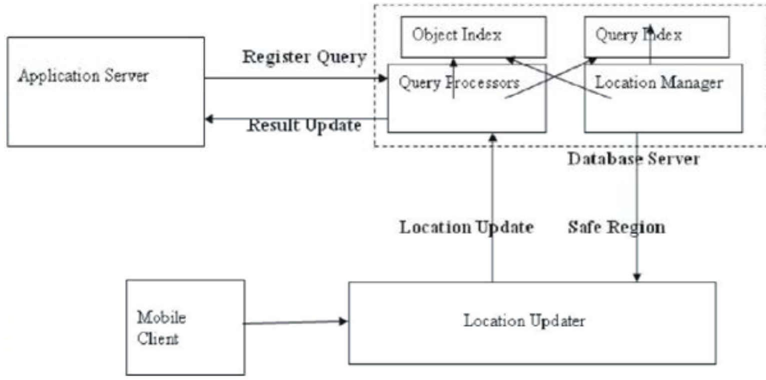

Fig. 1: Architecture of PAM framework

Location Updater: The clients maintain good connection with the database server through location updater. A client updates its location on the server only when the client moves out of its safe region. At the first client send its details are send to the location updater after the client receive the safe region from location updater. The location updater sends the client location to database server and receive the safe region from location manager. Each time a client detect its genuine point location, it is encapsulated in to a bounding box. Then the client side location updater decides whether or not to update the client to the server.

Database Server: The database server manages the location information of the objects. At the database server side, we have the moving object index, the query index, the query processor and the location manager. The database server handles location updates sequentially; in other words, updates are queued and handled on a first-come-firs-server basis. The object index is the server-side view on all objects. More specifically, to evaluate queries, the server must store the spatial range, in the form of abounding box, within which each object can possibly locate. The query index for each registered query, the database server stores the query parameters, the current query results and the quarantine area. The database server receives the client details from location updater and query received from application server. The query processor evaluates the most probable result when a new query is registered, or reevaluates the most probable result when a query is affected by location updates. The location manager computes the safe region for corresponding client and send to the location updater.

Application Server: The application servers gather monitoring requests and register spatial queries at the data base server, which then continuously updates the query results until the queries are deregistered. The application server is first authenticated. At any time, application servers can register spatial queries to database server. The query processor identifies those queries that are affected by this update using the query index and the reevaluates them using the object index. The updated query results are then reported to the application server. In this module we will be using kNN algorithm.

MATERIALS AND METHODS

PAM framework consists of components located at both the database server and the moving objects. At the database server side, we have the moving object index, the query index, the query processor and the location manager. At moving objects' side, we have location updaters. Accommodate all moving objects in secondary memory. This assumption has been widely adopted in many existing proposals .The database server handles location updates sequentially in other words, updates are queued and handled on a first-come-first-serve basis. This is a reasonable assumption to relieve us from the issues of read/write consistency. The moving objects maintain good connection with the database server. Furthermore, the communication cost for any location update is a constant. With the latter assumption, minimizing the cost of location updates is equivalent to minimizing the total number of updates.

At any time, application servers can register spatial queries to the database server. When an object sends a location update, the query processor identifies those queries that are affected by this update using the query index and then, reevaluates them using the object index.

The updated query results are then reported to the application servers who register these queries. Afterward, the location manager computes the new safe region for the updating object, also based on the indexes and then, sends it back as a response to the object.

The object index is the server-side view on all objects. More specifically, to evaluate queries, the server must store the spatial range, in the form of a bounding box, within which each object can possibly locate. For each registered query, the database server stores: the query parameters and the current query result.

In the PAM framework, based on the object index, the query processor evaluates the most probable result when a new query is registered, or reevaluates the most probable result when a query is affected by location updates. Obviously, the reevaluation is more efficient as it can be based on previous results. Location manager create safe region for each moving object. Safe region is nothing but the numerical value.

CONCLUSION

The real time PAM architecture is implemented using spatial queries. Here, PAM architecture is simulated using relational database. The proposed work is focused on the PAM framework in which privacy is maintained and minimizes the number of location updates. The efficient kNN (k Nearest Neighbor algorithm) in the PAM framework is developed.

REFERENCES

1. Haibo Hu, Jianliang Xu, Senior Member, IEEE and Dik Lun Lee, 2010. PAM: An Efficient and Privacy-Aware Monitoring Framework for Continuously Moving Objects Haibo Hu, 22(3).
2. Saltenis, S., C.S. Jensen, S.T. Leutenegger and M.A. Lopez, 2000. Indexing the Positions of Continuously Moving Objects, Proc.ACM SIGMOD.
3. Gruteser, M. and D. Grunwald, 2003. Anonymous Usage of Location-Based Services through Spatial and Temporal Cloaking, Proc. Mobi Sys.
4. Hu, H., J. Xu and D.L. Lee, 2005. A Generic Framework for Monitoring Continuous Spatial Queries over Moving Objects, Proc. ACM SIGMOD, pp: 479-490.
5. Yu, X., K.Q. Pu and N. Koudas, 2005. Monitoring K-Nearest Neighbor Queries over Moving Objects, Proc. IEEE Int'l Conf. Data Eng. (ICDE).
6. Ghinita, G., P. Kalnis and S. Skiadopoulos, 2007. Prive: Anonymous Location-Based Queries in Distributed Mobile Systems, Proc. Int'l World Wide Web Conf. (WWW '07), pp: 371-380.
7. Gedik, B. and L. Liu, 2008. Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms, IEEE Trans. Mobile Computing, 7(1): 1-18.
8. Thooyamani, K.P., V. Khanaa and R. Udayakumar, 2013. A frame work for modelling task coordination in Multi-agent system, Middle-East Journal of Scientific Research, 15(12): 1851-1856. ISSN: 1990-92.