Implementing Secure Group Queries for Collaborative Spatial Computing

AFRIN SYED¹, SUDHAKAR VENATI²

¹PG Scholar, Dept of CSE, Andhra Engineering College, Atmakur, AP, India.
²Assistant Professor, Dept of CSE, Andhra Engineering College, Atmakur, AP, India.

Abstract: With the speedy development of location-aware mobile devices, omnipresent net access and social computing technologies, scores of users’ personal data, like location information and social information, has been promptly accessible from numerous mobile platforms and on-line social networks. The convergence of those 2 kinds of information, called geo-social information, has enabled cooperative spatial computing that expressly combines each location and social factors to answer helpful geo-social queries for either business or social sensible. during this paper, we tend to study a brand new kind of Geo-Social K-Cover Group (GSKCG) queries that, given a group of question points and a social network, retrieves a minimum user cluster within which every user is socially associated with a minimum of k alternative users and therefore the users’ associated regions (e.g., acquainted regions or service regions) will together cowl all the question points. we tend to consequently explore a group of effective pruning methods to derive associate economical rule for locating the optimum answer. Moreover, we tend to style a unique index structure tailored to our drawback to more accelerate question process.

Keywords: Social data, Location data, Geo-Social Queries.

I. INTRODUCTION

The convergence of location data and social data, known as geo-social data, has enabled a new computing paradigm that explicitly combines both location and social factors to generate useful computational results for either business or social good. In this paper, we use the term collaborative spatial computing to represent this emerging paradigm. The idea of collaborative spatial computing has been widely used in various domains, including location based social networks. One of the most important applications of collaborative spatial computing in the database field is geo-social queries, which are attracting increasing interest from both industrial and academic communities. The study of geo-social queries is in its incipience. The pioneering studies consider geo-social queries that take as inputs a set of mobile users, a query location point and certain social acquaintance constraint and that return a set of users with the minimum location distance while satisfying the social constraint. In this paper, we propose a novel type of geo-social queries, called Geo-Social K-Cover Group (GSKCG) queries, which is based on spatial containment and a new modeling of social relationships. Intuitively, given a set of spatial query points and an underlying social network, a GSKCG query finds a minimum user group in which the members satisfy certain social relationship and their associated regions can jointly cover all the query points. We formally define a GSKCG query to capture the natural search requirements driven by the real-life applications. GSKCG queries differ from the existing geo-social queries in both the spatial and social factors.

II. BACKGROUND WORK

A. Spatial query processing

Spatial question process supported R-tree or its extensions has been extensively studied over the past 20 years. the present analysis has centered on varied styles of queries, as well as k-nearest-neighbor queries, vary Queries, among others. Roussopoulos et al. conferred associate degree economical branch-and-bound R-tree traversal algorithmic rule to go looking the closest neighbor object to a question purpose, then extended it to k-nearest-neighbor search. Recently, spatial queries have been extended to incorporate text keywords, known as spatial keyword queries in the literature. Zhou et al. proposed a hybrid index structure to handle both textual and spatial queries. Cong et al. presented a new indexing scheme called IR-tree, which integrates the R-tree and inverted files for location-aware top- k object retrieval. Li et al. proposed a novel spatial-aware interest group (SIG) query and presented two kinds of IR-tree based algorithms, interest-oriented and diameter oriented, to tackle SIG queries efficiently. However, these works cannot deal with queries considering the social factor. In this paper, we also present a novel index structure, Enhanced Social aware R-tree, which integrates the user’s social relationships into the R-tree, to process GSCKG queries efficiently.

B. Social query processing

There have been some studies on group and team queries over social networks with the goal of finding a user group with a certain social relationship. Social groups or teams are usually cohesive subgraphs formed by users with acquaintance relations. Their acquaintance levels can be measured by several classical graph models, such as clique, k-core, and k-plex.Yang et al. proposed the social-temporal group query to find a group of activity attendees with the minimum total social distance to the query issuer. Lappas et al. and Li et al. studied the problem of expert team formulation which aims to find a group of experts covering all required skills and minimize the communication cost.
among them. In this paper, we use k-core to model users’ social relations, which is different from the previous studies. In addition, a GSKCG query takes into consideration the spatial factor.

C. Geo-Social Query Processing

Efficiently process queries considering each spacial and social constraints attracts more and more attention recently. A main steam is to mine users’ location and social network knowledge to seek out the relationships between the users and their locations. Liu et al. planned the circle-of-friend question to realize minimal-diameter social teams. They extended the density-based cluster paradigm and applied it on places that area unit visited by users of a geo-social network. Armenatzoglou et al. proposed a general framework that offers flexible data management and algorithmic design for Geo-Social Networks (GeoSNs) queries. They designed a new index structure called Social R-tree to integrate the users’ social relationships into an R-tree for efficient query processing. This index is different from our Enhanced SaR-tree in that it is used to reduce the checking states during the enumeration. Zhu et al. presented a new family of geo-social group queries with minimum acquaintance constraint (GSGQs), and also designed a new index structure named SaR-tree to accelerate the GSGQs queries. However, the SaRtree cannot be directly adopted by our GSKCG queries due to our regional spatial factor which differs from the point spatial factor.

III. PROPOSED WORK

In this section, we present our KCGFinder algorithm and a set of pruning strategies for answering GSKCG queries.

A. Basic Algorithm

To satisfy the minimum cardinality requirement of a GSKCG query, the general idea of KCGFinder is to process the user groups in increasing order of group size and return the current group as soon as it is valid. Algorithm 1 gives the pseudo code of the KCGFinder algorithm.

Algorithm 1 KCGFinder (Query points P , Integer k , LBSN G)

1. $S \leftarrow$ The set of users in $G$ that each covers at least one point in $P$;
2. $U_k \leftarrow$ The set of users belonging to $S$ that may appear in a $k$-core;
3. $H \leftarrow$ All connected components of $G[U_k]$ that each fully covers $P$;
4. $M \leftarrow \min_{C \in H} |C|$;
5. for $x$ from $k-1$ to 1 do
6. for each $C^x$ in $H$ do
7. if $|C^x| \geq x$ then
8. $C^x \leftarrow$ GetOptimalGroup $(C^x, k, a, P)$
9. if $C^x \neq 0$ then
10. Return $C^x$;
11. Return $\emptyset$;

It can be observed that the main complexity of KCGFinder comes from the GetOptimalGroup function. Therefore, in the rest of this section, we focus on how to optimize GetOptimalGroup via a set of pruning techniques. We give the general idea of GetOptimalGroup in Algorithm 2.

Algorithm 2 GetOptimalGroup (Component $G$, Integer $k$, Query points $P$)

1. for each size-$a$ user group $G$ do
2. if the number of edges of $G[C]$ \( \geq k(a+1)/2 \) then
3. if $G[C]$ is $k$-core and $P \subseteq \bigcup_{C \in G} u. R$ then
4. Return $C$;
5. Return $\emptyset$;

B. Basic Pruning

We start with two basic pruning strategies, k-core (KC) based pruning and spatial query-point coverage (SQPC) based pruning, based on the degree constraint in a k-core and the spatial query point coverage constraint, respectively. By the definition of k-core, we know that the minimum degree of each vertex in a k-core should be no less than $k$. Therefore, in the branch and bound search, if the minimum degree constraint cannot be satisfied after adding any new users from SU to SI, the search process should backtrack to the previous state of SI. Any valid user group should cover all query points $P$. If SU cannot fully cover the rest query points in $P - S_I$, where $S_I$ is the set of points covered by SI, adding any user from SU to SI cannot form a valid group. In this case, the search process can safely prune the subtree rooted at SI without missing the optimal solution. In some cases, even though SU can cover all query points in $P - S_I$, the users of SU are still not a member of any valid group.

C. Hybrid Index

In this section, we design a novel index structure, the Enhanced Social-aware R-tree (SaR-tree), to further accelerate query processing. The SaR-tree structure is a variant of R-tree that indexes both spatial locations and social relations. Figure 4 illustrates a simple SaR-tree. Different from a classical R-tree, each entry of an SaR-tree contains two major pieces of information: a set of core bounding rectangles (CBRs) (see Definition 5) that encodes the social information and a minimum bounding rectangle (MBR) that encodes the spatial information as in an R-tree. Intuitively, a CBR bounds the users by the social constraint while an MBR bounds the users by the spatial constraint, and therefore an SaR-tree gains the ability of both social-based and spatial-based pruning for GSKCG query processing. We propose a novel index structure, known as the enhanced SaR-tree, to address this problem. To construct an Enhanced SaR-tree over an LBSN, we first construct a standard R-tree and then compute the CBR for each entry in rtree. To compute the CBR of an entry, we should know how to build a user’s CBR. The general idea of constructing a user’s CBR includes two steps. First, as the users’ associated regions may intersect with each other, we calculate the user’s internal CBR. Second, given the user’s internal CBR, we expand it to obtain the corresponding external CBR, from which the user’s CBR will be selected. We give the formal definitions of these two types of CBRs below. Algorithm 3 describes how to construct a CBR of a user.
Then we use the GetInternalCBRs function to generate the internal CBRs on both directions. Based on these internal CBRs, we invoke the GetExternalCBRs function to calculate the corresponding external CBRs. Finally, the external CBR with the maximum area is returned as u’s CBR. Next we elaborate GetInternalCBRs and GetExternalCBRs.

The GetInternalCBRs function is shown Algorithm 4 & 5 respectively.

Finally, we discuss how to compute the CBR of each entry in the enhanced SaR-tree by a bottom-up approach. A leaf entry’s CBR is the CBR of the user it represents.

D. GSKCG Query Processing

In this section, we present our integrated algorithm SaRBasedKCGFinder. Generally, the algorithm consists of two steps: 1) filter impossible users based on the enhanced SaR-tree; 2) feed the remaining users to KCGFinder. We give the details of SaRBasedKCGFinder in Algorithm 6.

IV. CONCLUSION

In this paper, we have introduced a new practical type of GSKCG queries that considers both users’ associated spatial regions and their social acquaintance levels. A GSKCG query aims to find a minimum user group that covers all query points and that is a k-core. We have proposed an efficient algorithm SaRBasedKCGFinder to find the optimal solution, whose success lies in a set of effective pruning strategies and a novel index structure.

V. FUTURE ENHANCEMENT

As for future work, we plan to work on the following two extensions. First, the social graph used in this paper is unweighed, we intend to extend our algorithm to support a weighted social graph. Second, in some cases, we need not an exact solution. How to design an efficient approximation algorithm with a tight approximation bound is also our future work.

VI. REFERENCES


Author’s Profile:
Afrin Syed has received has B.Tech Degree in Computer Science Engineering from Sri Raghavendra Institute of science & Technology, Vinjumur, affiliated to JNTUA in 2013 and Pursuing M.Tech degree in Computer Science & Engineering in Andhra Engineering College, Atmakur, affiliated to JNTUA, Anantapur in 2016.

Sudhakar Venati has received the M.Tech (CS) degree from JNTU-K and B.Tech (CSIT) from JNTU. And he is also done the PGDCA from ET&T, New Delhi. Currently working as an assistant professor in Andhra Engineering College, Atmakur, and NelloreDt-524322(A.P).