The Flexible Socio Spatial Group Queries

Bishwamittra Ghosh
BUET, Bangladesh
bishwamitra.ghosh@gmail.com

Sajid Hasan Apon
BUET, Bangladesh
sajidhasanapon@gmail.com

Mohammed Eunus Ali
BUET, Bangladesh
eunus@cse.buet.ac.bd

Timos Sellis
Swinburne University of Technology, Australia
tsellis@swin.edu.au

Farhana M. Choudhury
RMIT University and University of Melbourne, Australia
fchoudhury@unimelb.edu.au

Jianxin Li
The University of Western Australia
jianxin.li@uwa.edu.au

ABSTRACT

A socio spatial group query finds a group of users who possess strong social connections with each other and have the minimum aggregate spatial distance to a meeting point. Existing studies limit to either finding the best group of a fixed size for a single meeting location, or a single group of a fixed size w.r.t. multiple locations. However, it is highly desirable to consider multiple locations in a real-life scenario in order to organize impromptu activities of groups of various sizes. In this paper, we propose Top k Flexible Socio Spatial Group Query (Top k-FSSGQ) to find the top k groups w.r.t. multiple POIs where each group follows the minimum social connectivity constraints. We devise a ranking function to measure the group score by combining social closeness, spatial distance, and group size, which provides the flexibility of choosing groups of different sizes under different constraints. To effectively process the Top k-FSSGQ, we first develop an Exact approach that ensures early termination of the search based on the derived upper bounds. We prove that the problem is NP-hard, hence we first present a heuristic based approximation algorithm to effectively select members in intermediate solution groups based on the social connectivity of the users. Later we design a Fast Approximate approach based on the relaxed social and spatial bounds, and connectivity constraint heuristic. Experimental studies have verified the effectiveness and efficiency of our proposed approaches on real datasets.

PVLDB Reference Format:
DOI: https://doi.org/10.14778/3282495.3282497

1. INTRODUCTION

Various social network sites now allow users to capture their locations through GPS-enabled devices and share them through check-ins or mentions in posts. As a result, socio spatial networks are emerging where each user is associated with a physical location along with the connectivity with other members of the network.

Given such a network, socio spatial group queries [7, 25, 30, 33] aim to find the ‘best’ group against a Point Of Interest (POI) where the users possess social tightness within the group and have spatial closeness to the POI. An example of such queries is to find a group of three members, who are located close to a particular restaurant and socially connected to at least one of the other members, so that the group is competent for a targeted advertisement of a “20% discount for a table of three” offer running at that restaurant.

Although existing works have contributions towards finding an important class of group queries, there exists several gaps with the real-life applications, particularly the following major limitations:

(i) Impracticality of specifying group size: In each of the existing socio spatial group queries (7, 25, 30, 33), a single value as the size of the group (i.e., fixed size group) needs to be specified by the query issuer a priori. However, without prior knowledge of the social connections and users’ locations, it is difficult to provide an exact and explicit size for the desired group. For example, “buy two get one” is a traditional offer. However, the advertiser may find that most of the groups close to the POI are generally of four people. So the group size of three members may not be a feasible offer.

(ii) Finding only the best group for only one POI: The algorithms in [7, 30, 33] can find only one group against only one query POI, where the solutions are not easily extendable for multiple groups or POIs (Section 2). Finding multiple groups are important for advertisers, and multiple POIs are important to suggest the best meeting location. For example, given multiple event locations of a festival, each resulting group can get advertisement for its nearest event. To the best of our knowledge, the only existing work that incorporates multiple POIs is [25]. However, their proposed algorithm can find only one group (i.e., \( k = 1 \)) as the result. They use an R-tree and a Ball-tree as index, and the algorithm is not easily extendable for \( k > 1 \), which limits the applicability of the work.

(iii) Profit optimization - a trade-off between the group effectiveness and the group size: Advertisers want to offer the best deal to attract closely connected users who are located nearby to the POI; and at the same time they want to maximize their profit by preferring a larger group, as increasing the group size is more profitable. However, increasing the group size may decrease the users’ satisfaction in the meet-up as it increases the chance of meeting with more unknown people. Thus, there should have a trade-off between satisfaction and cost; and finding the optimal group size is essential for such scenarios. In literature, the group score is generally defined as the weighted combination of social and spatial scores of the group, but the size of the group is ignored in the scoring function. Hence, the existing work are not suitable to find the balance between the group effectiveness and group size trade-off.
To address the above limitations, we propose a novel Top-k Flexible Socio Spatial Group Query (Top-k-FSSGQ) that finds top k groups with flexible group sizes for a given set of POIs (meeting points). Consider a scenario, where a chain-shop has multiple branches and they want to find k best groups to offer total k coupons in different locations. Since the owner does not have any prior knowledge on the groups and users around each branch, she cannot decide on the number of coupons required in different branches, and what should be the formation of the coupons, e.g., “buy two get one” or “buy three get one”. Thus the objective is to find top k groups in total w.r.t. any query POIs (branches) with the highest socio spatial ranks, where the group size can vary within a query minimum and maximum size, specified by the query issuer based on the application. For example, if the POI is a restaurant, its maximum table size can be the maximum allowable group size. Formally, for a given socio-spatial graph, the minimum and maximum number of users allowed in a group, a minimum acquaintance constraint, a set of candidate meeting points, and a maximum distance constraint, the Top-k-FSSGQ query returns k best groups and their corresponding meeting points, where each group satisfies the minimum acquaintance and group size constraints, and the location of each member satisfies the distance constraint. We also guarantee that a top-k group cannot be fully contained in another top-k group, for example, given a set of users \{a, b, c, d, e\}, then groups like \{a, b, c\} and \{a, b\} both cannot be in the result.

We now present an example to illustrate the Top-k-FSSGQ query, which also highlights the limitations of the existing studies.

**Example 1.** Figure 1 presents a graph of users \(V = \{a, b, c, d, e\}\), where the upper and lower part represent the social and spatial layer, respectively. Each edge in the social layer represents the connectivity between two users. Each user has a location, shown with an arrow from the social to the spatial layer. Here, \(O = \{o_1, o_2\}\) is a set of POIs. Let, we want the top 2 groups with group size minimum 3 and maximum 4, and the resultant groups should satisfy the minimum acquaintance constraint, 1.

As the existing studies [7, 30, 33] can only find the best group of only one fixed size for only one POI, their algorithm needs to be repeatedly reissued for each of the POIs and for each of the group sizes between the minimum and maximum values. Although Shen et al. [25] allows multiple POIs as the query input, they can only find one single group (i.e., top-1 group) of a fixed size. Therefore, their approach cannot be directly applied to find the top-2 groups. From this example, it is evident that, (i) If the group size is strictly fixed, potential profitable groups of other sizes may get excluded, (ii) A high computational cost is incurred for the existing approaches by reissuing query with different sizes for different POIs. Here, our aim is to find the top-2 groups \(\{b, a, d, c\}\) and \(\{b, a, e, c\}\) in this example, detailed calculations are shown in Section 5.2) efficiently by avoiding the repeated unnecessary calculations.

To process Top-k-FSSGQ query, we extend the approach presented in [30] as our baseline. In this baseline, we repeat the approach for all possible groups of size between the minimum and maximum value against all meeting points that satisfy the constraints, rank the groups and then return the top k groups as result. As the baseline requires a high computational cost (details in Section 3), we propose multiple efficient solutions: (i) an efficient Exact approach that finds optimal groups with much less computation overhead, (ii) a heuristic based Approximate approach which further improves the efficiency by selecting members of the groups based on the social connectivity, (iii) a Fast Approximate approach that answers the query much efficiently by sacrificing the quality of the groups slightly, and (iv) a Greedy Approximate approach.

The key idea of the approaches is to derive theoretical bounds on spatial distance and social connectivities to effectively prune a large number of candidate groups that cannot be an answer. In details, we expand our search for all meeting points in parallel and select users for possible solution group for each POI. Selection of users is processed by prioritizing both spatial and social aspect so that the formed groups can satisfy the acquaintance constraint and possess the minimum aggregate spatial distance. For Exact approach, we exploit the connectivity and locations of already fetched users w.r.t. each meeting point to derive upper bounds that can safely determine whether we need to further explore the space for a higher rank group. We also define a bound for the familiarity constraint for a user that must be satisfied in order to qualify the group as an answer for our heuristic based Approximate approach. For the Fast Approximate approach, we design more powerful pruning strategies to reduce the exploration. We develop an upper bound on spatial distance and a lower bound on social connectivity of a member (in contrast to all members in the exact approach). Based on these bounds, we develop an early terminate strategy.

The contributions of this paper are described as follows.

(i) We re-define the socio spatial group query for multiple meeting points, and design a flexible ranking function consisting of social cohesiveness, spatial closeness, and group size.

(ii) We develop an Exact approach which is significantly faster than the baseline and multiple Approximate approaches that are highly efficient with almost similar quality of result.

(iii) We develop an early termination strategy and several pruning rules for improving the efficiency of the Exact approach and the Approximate approach based on the upper bound of spatial distance and lower bound of social connectivity.

(iv) We conduct extensive experiments to verify the efficiency of our developed algorithms and effectiveness of our approximate solutions by using real datasets.

2. RELATED WORK

Different variants of group queries in social network have been studied in literature recently. We can categorize the queries based on (i) group with social connectivity constraints only; (ii) group based on spatial distance; and (iii) group with both constraints.

2.1 Group Queries by Social Connectivity

Social connectivity based group queries can further be classified as team formation [16, 18], community detection [9, 22], and community search [5, 6, 8, 12, 19, 26]. Team formation [16, 18] aims at finding a group of experts in a social graph with required skills while minimizing communication cost within that group. Community detection aims at finding all communities in a given graph.
based on specific criteria such as modularity [15] or the other different contexts [9, 22]. Whereas, given a query node, community search aims at finding a group (community) of nodes in a large graph where the resulting community must contain the query node. Here, the query can be a single node [5–8, 12, 19, 26, 33] or multiple nodes [26] in a graph. Minimum degree metric or k-core is often being used as the social constraint in defining communities [6, 7, 26, 33]. Besides, k-clique [5], k-truss [12], and connectivity [11] have also been considered in online community search. In addition to the social factor, a socio temporal group query [29] emphasizes the availability of all attendees in an impromptu activity.

### 2.2 Group Queries Based on Spatial Distance

Group nearest neighbor queries that find a meeting point with the smallest aggregate distance (summation, maximum, etc.) from the group have been extensively studied in different contexts [2, 3, 20, 23, 31]. The studies [14, 32] explored optimal location query for a group, where given a set of users and a set of POIs, the query finds the location of a new meeting point that minimizes the average distance from each user to the closest meeting point [14, 32]. Other similar works [13, 24, 27, 28] find a location for a new server such that the maximum distance between the server and any client is minimized. Papadias et al. [23] find a location that minimizes the sum of the distances from the users. Ali et al. [3] find optimal subgroups and the meeting point for each subgroup that minimizes the aggregate spatial distance for the subgroup; whereas Ahmed et al. [2] extend [3] to include both spatial proximity and keyword similarity while selecting a meeting point for a subgroup.

### 2.3 Socio Spatial Group Queries

For a given socio spatial graph, Yang et al. [30] propose a socio spatial group query (SSGQ) that finds a set of members against a fixed rally point where the aggregate spatial distance between members and the rally point is minimized and each member is allowed to be unfamiliar with at most a maximum number of members in that group. Members’ locations are indexed using an R-tree, and new members are added to an initially empty set based on distance ordering and familiarity checking. Finally, a resultant group of predefined fixed size is returned. Our work is different from this work in several aspects. Our aim is to find top k groups of variable size for multiple POIs and each member has minimum level of acquaintance with the other members. In contrast, SSGQ only considers one rally point, which is impractical as for a large socio spatial graph multiple rally points may exist. Moreover, the average minimum familiarity constraint [30] cannot always be preferable for an individual where that member possesses weak social connectivity within the resulting group. Lastly, potential candidate members may be excluded due to the fixed size group.

Shen et al. [25] propose the multiple rally-point social spatial group query (MRGQ) that chooses a suitable rally point from the multiple points and the corresponding best group, which exhibits the minimized spatial distance between group members to the best rally point. The resulting group is of fixed size and satisfies the average minimum familiarity constraint. To efficiently process the query, an R-tree is used to index member locations and a Ball-tree is used to index rally points. As only the best group is returned, indexing both activity locations and member locations makes solution efficient. Incorporating the idea of multiple rally points enhances the acceptance of MRGQ, but finding only the best group for multiple POIs limits the applicability of socio spatial group queries. The MRGQ also considers an average minimum social connection as social cohesiveness, which limits individual’s satisfaction. Moreover, the resulting group has fixed size limitation.

Zhu et al. [33] propose a new class of geo-social group queries with minimum acquaintance constraint (GSGQs), where the result group guarantees the worst-case acquaintance of all users. GSGQs takes three parameters: query issuer, spatial constraint, and social constraint. Query issuer is a member in the given graph. Minimum degree c is the social constraint. GSGQs considers three different spatial constraints, i.e., GSGQs returns largest c-core within a range or a c-core of more than a fixed size or a c-core of a fixed size. Fang et al. [7] propose a spatial-aware community (SAC), which is a connected c-core where the members in the resulting group are located within a spatial circle having a minimum radius. SAC also maintains the minimum acquaintance constraint. c-core is experimented effectively [7, 33] when a single query issuer is a member in resulting group. However, in a large socio spatial graph, different sized groups with different social and spatial configuration exist. Socio spatial query issued by a member [7, 33] serves individual’s purpose but does not help to understand the presence of various socio spatial clusters in a given graph. Armenatzzoglou et al. [4] propose a set of geo-social ranking (GSR) functions to combine both social and spatial factors and find top-k users with respect to each of these functions. They introduce a general GSR framework and propose four functions that cover several practical scenarios. In our solution, we have adopted similar technique like [4] to score social tightness and spatial closeness of a group. Although GSR functions help understanding basic ranking criteria in assessing the score of social and spatial features, ranking users cannot serve the goal of socio spatial group queries. For this reason, we are computing the score of groups rather than users which helps to find top k groups in a given socio spatial graph.

### 3. Problem Formulation & Baseline

Let a socio-spatial graph be $G = (V, E)$ where $V$ is the set of members and $E$ is the set of edges representing the social connections. Let $l_v$ be the location for $v \in V$ and $O$ be a set of candidate meeting points. We will first define our socio-spatial group score function. A group with strong social connection among members and less aggregate spatial distance to a meeting point is preferred, where a user is interested in meeting points within a spatial range. Group size is also important as a large group with the same connectivity and aggregate spatial aspect is more preferable. We adopt the social and spatial score measures from [4]. Now consider a subgraph of $G$ as $G' = (V', E')$ and a meeting point $o' \in O$. The socio-spatial group score of $(G', o')$ can be measured as follows.

**Definition 1. (Social connectivity score)** The social score is computed based on the average social connectivity, provided that each member satisfies a minimum acquaintance constraint, $c$. The social connectivity score $S_{sc}$ is the density of $G'$ and computed as $\frac{|E'|}{|V'|(|V'| - 1)}$.

**Definition 2. (Spatial closeness score)** The spatial closeness score $S_{sp}$ of $G'$ is inverse to the normalized average distance of the group members to $o'$ and computed as $1 - \frac{\sum_{v \in V'} d(v, o')}{\sum_{v \in V'} d(v, o')}$, where $d(v, o')$ is the spatial distance from $v_i$ to $o'$ and $d_m$ is the maximum spatial distance of a user from a meeting point.

**Definition 3. (Group size score)** The group size score $S_{gs}$ of $G'$ is directly proportional to its group size $|V'|$ and is computed as $\frac{|V'|}{n''}$, where $n''$ is the maximum group size.

**Definition 4. (Socio-spatial group score)** Given a subgraph $G' = (V', E')$ in $G$, a meeting point $o' \in O$, a maximum group
size $n''$ and a maximum spatial range $d_{m}$. We measure the socio spatial group score $S(G', a')$ as a linear weighted combination of the individual scores, which is the most common type of combination used in the literature [4, 21]. Note that, any other types of combinations of the scores can also be used in our solutions.

$$S = \alpha * S_{sc} + \beta * S_{sp} + \gamma * S_{gs}$$

(1)

Here, $\alpha + \beta + \gamma = 1$, and the values of $\alpha$, $\beta$ and $\gamma$ can be set based on priority of social, spatial, and group size, respectively. Based on the socio-spatial group score $S(\ldots)$, the Top $k$-FSSGQ query can be formalized as follows.

**Definition 5.** (Top $k$-FSSGQ) Given a socio spatial graph $G = (V, E)$, a set $O$ of locations as the meeting points, a minimum group size $n'$, a maximum group size $n''$, a minimum acquaintance constraint $c$, a maximum spatial range $d_{m}$, and a parameter $k$, the Top $k$-FSSGQ finds a ranked list of top-$k$ groups and their corresponding meeting points from $O$, each as a tuple of the form $(G_{1}, o_{1}, S(G_{1}, o_{1}))$. Here, each $G_{i}$ is a subgraph of $G$, and each $(G_{i}, o_{i}, S(G_{i}, o_{i}))$ must satisfy the following conditions.

1. $(G_{i}, o_{i})$ is considered as an eligible candidate if and only if it meets the spatial and social constraints, i.e., $n' \leq |V(G_{i})| \leq n''$ holds, the minimum degree of any node in $G_{i}$ is no less than $c$, and the maximum distance of any user in $G_{i}$ to $o_{i}$ is no larger than $d_{m}$.
2. $S(G_{i}, o_{i}) \geq S(G_{j}, o_{j})$ for any $1 \leq i \leq k - 1$;
3. If $\bar{o} \in O$, $G_{j} \subseteq G$ and $S(G_{j}, o) > S(G_{i}, o)$ where $1 \leq i \leq k$, $j > k$ and $(G_{j}, o)$ is an eligible candidate.

**Parameter settings and choice of default values.** The problem formulation is made generalized using different parameters to cater for different needs of groups. As a component of the scoring function, the group size is the group size constraints $n'$ and $n''$ are optional. Here, $n'$ can be set as default to ‘1’, and $n''$ can be set to the number of users of the social graph. Similarly, the minimum acquaintance $(c)$ and the maximum distance $(d_{m})$ are useful to filter out the groups that the user does not want in the result. If the user is unsure of these values, $c$ can simply be set to default ‘1’ and $d_{m}$ to infinity so the constraints do not have any affect on the result. As value of all the components in Equation 1 is normalized between $[0,1]$, $\gamma$ can be replaced with $(1 - \alpha - \beta)$ without losing generality.

**Theorem 1.** The Top $k$-FSSGQ problem is NP-hard.

**Proof.** We prove this by the reduction from the $n''$-clique. Given a graph $G_{c}$, $n''$-clique decision problem determines whether the graph contains a clique, i.e., a complete graph of $n''$ vertices. For Top $k$-FSSGQ problem, assume that $G = G_{c}$, $c = n'' - 1$, $n'' = n''$, $d_{m} = \infty$, $O = \{o\}$ and $\forall v \in V$, $d(v, o) = 1$. We first prove the necessary condition. If $G_{c}$ contains a $n''$-clique, there must exist a group with the same vertices in the $n''$-clique such that every member has social connectivity with all the other members in that group. Hence the total spatial distance is $n''$. We then prove the sufficient condition. If $G$ in Top $k$-FSSGQ has a group of minimum size of $n'$ and maximum size of $n''$ and $c = n'' - 1$, $G_{c}$ in problem $n''$-clique must contain a solution of size $n''$ too. Therefore, the Top $k$-FSSGQ is an NP-hard problem.

**Baseline approach:** We develop a baseline by extending one of the most relevant work proposed by of Yang et al. [30]. Yang et al. developed a sub-optimal solution for a socio spatial group query that finds the best group of a fixed size against a single meeting point, while the resultant group follows the required average minimum acquaintance constraint among members. Thus to find the answer of Top $k$-FSSGQ, for each meeting point, we find the best groups for each allowable group size (between the minimum and maximum group size), where the groups follow our required minimum acquaintance constraint $c$. Then, we rank all the groups that are found w.r.t. all meeting points to find the top $k$ groups. Note that in the above steps we only consider members that fall within the maximum spatial range $d_{m}$.

### Table 1: Basic notation

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d(v, o)$</td>
<td>The spatial distance between $v \in G$ and $o \in O$</td>
</tr>
<tr>
<td>$n'$</td>
<td>Minimum (maximum) query group size</td>
</tr>
<tr>
<td>$n''$</td>
<td>Minimum acquaintance constraint</td>
</tr>
<tr>
<td>$d_{m}$</td>
<td>Maximum spatial distance constraint</td>
</tr>
<tr>
<td>$k$</td>
<td>No. of results to be returned</td>
</tr>
<tr>
<td>$\alpha, \beta, \gamma$</td>
<td>Preference parameters in the scoring function</td>
</tr>
<tr>
<td>$V_{eq}$ ($V_{eq}$)</td>
<td>The set of already explored members (remaining members) of a candidate group for $o_{eq} \in O$</td>
</tr>
<tr>
<td>$f_{c}$ ($f_{k}$)</td>
<td>Current social connectivity in $V_{eq}$ ($k^{th}$ group)</td>
</tr>
<tr>
<td>$d_{s}$ ($d_{k}$)</td>
<td>The current aggregate spatial distance of all members in $V_{eq}$ ($k^{th}$ group)</td>
</tr>
<tr>
<td>$\delta_{f}$ ($\delta_{o}$)</td>
<td>The additional increase of the total social connectivity (aggregate spatial distance)</td>
</tr>
<tr>
<td>$m$</td>
<td>The maximum additional social connectivity of new members to the group</td>
</tr>
<tr>
<td>$d_{c}$</td>
<td>Spatial upper bound for a group of size $n$</td>
</tr>
<tr>
<td>$\maxdeg$</td>
<td>The maximum degree of the members in $V_{eq}$</td>
</tr>
<tr>
<td>$\mindeg$</td>
<td>The minimum spatial distance of the members in $V_{eq}$ from the meeting point $o_{eq}$</td>
</tr>
<tr>
<td>$f^{\uparrow}$</td>
<td>The lower bound on social connectivity</td>
</tr>
<tr>
<td>$d^{\uparrow}$</td>
<td>The upper bound on spatial distance</td>
</tr>
<tr>
<td>$f(v, V_{eq})$</td>
<td>The number of social connectivity of $v$ in $V_{eq}$</td>
</tr>
</tbody>
</table>

### 4. AN EFFICIENT EXACT APPROACH

We propose an efficient exact solution for answering the Top $k$-FSSGQ. The key idea is to develop an early termination strategy based on our derived upper bound on spatial distance that avoids the exploration of a large number of groups. Next, we present our advance termination strategy that determines when we should stop the exploration of members w.r.t. a meeting point.

#### 4.1 An Advance Termination Strategy

Let us assume that we have already found $k$ initial groups that satisfy the necessary constraints of our query, and let $k^{th}$bestscore be the current score of the $k^{th}$ group. Now we need to find whether any un-explored group has a higher score than the $k^{th}$bestscore. Let for any meeting point $o_{eq} \in O$, $V_{eq}$ be the set of already explored members for a candidate group and $V_{eq}$ be the set of remaining members that are yet to be explored. If we can guarantee that including more members from $V_{eq}$ to $V_{eq}$ will not yield any group having a score greater than $k^{th}$bestscore, we can safely terminate the search process for the meeting point $o_{eq}$. Based on the above observation, we will now formulate some bounds that can ensure the safe termination of the search.

Let $f_{k}$ be the total social connectivity of the $k^{th}$ group, $n_{k}$ be the number of members in the group, and $d_{k}$ be the aggregate spatial distance of the group members from the corresponding meeting point. Also, let $f_{c}$ be the current total social connectivity of members in $V_{eq}$ and $d_{c}$ be current aggregate spatial distance of all members of $V_{eq}$ from the meeting point $o_{eq}$. The ranking score of a group is computed based on the weighted combination of three scores: social score, spatial score and group size score. Thus, we first determine the score gains that can be obtained by the new group w.r.t. the $k^{th}$ best group in these three scoring measures.

102
Social score gain: The social score of the $k^{th}$ group is $f_k/(n_k \cdot (n_k - 1))$ (Definition 1). Let, we are considering some more members to be included in the currently explored group $V_{kq}$ so that final group size becomes $n$ where $n' \leq n \leq n''$. In this process, let $\delta_t$ be the additional increase of the total social connectivity if we add more members to $V_{kq}$ from $V_{R_k}$. Then, the social score gain of any newly formed group w.r.t. the social score of the $k^{th}$ group is:

$$\Delta S_{sc} = \frac{f_c + \delta_t}{n \cdot (n - 1)} - \frac{f_k}{n_k \cdot (n_k - 1)}$$

Spatial score gain: Similarly, the spatial score of the $k^{th}$ group is $1 - d_k/(n_k \cdot d_m)$ (Definition 2). Let $\delta_d$ be the additional increase of the aggregate spatial distance of the currently explored group if we add more members to $V_{kq}$ from $V_{R_k}$. Thus, the spatial score of the newly formed group will be $1 - (d_c + \delta_d)/(n \cdot d_m)$. Hence the spatial score gain of the newly formed group from the spatial score of the $k^{th}$ group is $\Delta S_{sp} = \frac{d_k}{n_k} - \frac{d_c + \delta_d}{n}$.

Group size score gain: Similarly, the gain of the group size score of the current group w.r.t. the $k^{th}$ group (Definition 3) is as follows.

$$\Delta S_{gs} = \frac{n - n_k}{n''}$$

Based on the above formulations, we can derive the combined score gain of the new group over the $k^{th}$ group. If the gain is positive, that implies that the new group may have a better score than the $k^{th}$ group, thus a candidate for the result. Therefore, the new group must satisfy the following equation.

$$\text{Gain} = \alpha \cdot \Delta S_{sc} + \gamma \cdot \Delta S_{sp} + \beta \cdot \Delta S_{gs} > 0$$

$$\Rightarrow \alpha \left( \frac{f_c + \delta_t}{n \cdot (n - 1)} - \frac{f_k}{n_k \cdot (n_k - 1)} \right) + \beta \left( \frac{d_k}{n_k} - \frac{d_c + \delta_d}{n} \right) > 0$$

$$\Rightarrow \text{Gain} = \frac{\alpha \left( \frac{f_c + \delta_t}{n \cdot (n - 1)} - \frac{f_k}{n_k \cdot (n_k - 1)} \right) + \beta \left( \frac{d_k}{n_k} - \frac{d_c + \delta_d}{n} \right)}{n''} \cdot (n - n_k) + \gamma$$

$$\Rightarrow \text{Gain} = \frac{\alpha \left( \frac{f_c + \delta_t}{n \cdot (n - 1)} - \frac{f_k}{n_k \cdot (n_k - 1)} \right) + \beta \left( \frac{d_k}{n_k} - \frac{d_c + \delta_d}{n} \right)}{n''} \cdot (n - n_k) + \gamma$$

Here, we have two unknown variables: additional total social connectivity $\delta_t$ and additional aggregate spatial distance $\delta_d$ that a group can achieve. To find the maximum gain in Eq. (2), we need to find the maximum possible $\delta_t$ and the minimum possible $\delta_d$.

4.2 Computing $\delta_t$

Let $f_m$ be the maximum additional social connectivity that a new group can achieve if we include new members from $V_{R_k}$ to $V_{kq}$ for $o_q \in O$. The next member from $V_{R_k}$ to be included in $V_{kq}$, can be connected with maximum $|V_{kq}|$ members, and the second next member can be connected with maximum $|V_{kq}| + 1$ members, and so on. Thus, we get:

$$f_m = |V_{kq}| + (|V_{kq}| + 1) + (|V_{kq}| + 2) + \ldots + (n - 1)$$

$$= \frac{(n - |V_{kq}|) \cdot (n + |V_{kq}| - 1)}{2}$$

In this case, we assume that the maximum degree, maxdeg, of the complete initial set of members $V_{R_k}$ for $o_q$ is greater than or equal to $n - 1$. However, since a member $v \in V_{R_k}$ cannot be connected with more than maxdeg members, we can tighten the $f_m$ bounds. Thus, we can derive $f_m$ as follows.

$$f_m = \frac{|V_{kq}|}{2} \cdot (|V_{kq}| + 1) + \frac{(|V_{kq}| + 2) \cdot (|V_{kq}| + 1)}{2} + \ldots + \frac{(n - |V_{kq}|) \cdot (n + |V_{kq}| - 1)}{2}$$

$$= \frac{(n - |V_{kq}|) \cdot (n + |V_{kq}| - 1)}{2} + \frac{n \cdot (n + |V_{kq}| - 1)}{2} - \frac{|V_{kq}| \cdot (|V_{kq}| + 1)}{2}$$

$$= \frac{n \cdot (n + |V_{kq}| - 1)}{2} - \frac{|V_{kq}| \cdot (|V_{kq}| + 1)}{2}$$

If the current candidate group $V_{kq}$ has already more than maxdeg members, the next included members can be socially connected with at most maxdeg members in $V_{kq}$. Thus, when $\text{maxdeg} < n - 1$ and $|V_{kq}| > \text{maxdeg}$, then, $f_m = (n - |V_{kq}|) \cdot \text{maxdeg}$. A new edge in a socio spatial graph increases the social connectivity of the graph by two. Since $f_m$ represents the maximum social connectivity that can be achieved by adding more members in $V_{kq}$, the additional social connectivity of $V_{kq}$ can be increased by at most $2 \cdot f_m$. Thus, we get $\delta_t = 2 \cdot f_m$.

4.3 Distance Upper Bound

Based on the computed upper bound of $\delta_t$, i.e., $\delta_t = 2 \cdot f_m$, we can derive the distance upper bound, $d'_n$, for the group by replacing the value of $\delta_t$ in Eq. (2) as:

$$n \left( \frac{d_m}{\beta} \cdot \alpha \left( \frac{f_c + \delta_t}{n \cdot (n - 1)} - \frac{f_k}{n_k \cdot (n_k - 1)} \right) + \frac{(n - n_k) \cdot \gamma}{n''} \right) + \frac{d_k}{n_k} - d_c = d'_n$$

Hence, $d'_n$ is the upper bound for the aggregate spatial distance of the members in $V_{R_k}$ who can be included in $V_{kq}$ to form a feasible group of size $n$.

4.4 Advance Termination

Based on the computed $d'_n$, we deduce an early termination strategy for the group search. Let $d_{min}$ be the minimum spatial distance of the members in $V_{R_k}$ from $o_q$. Let us assume that we want to form a new group of size $n'$, where $n' \leq n \leq n''$. Thus, the minimum aggregate spatial distance of the new $n - |V_{kq}|$ members can be computed as $d_{min} \ast (n - |V_{kq}|)$. Hence, $\delta_d = d_{min} \ast (n - |V_{kq}|)$. Consequently, the new group cannot be included in the answer list, if the following condition holds.

$$d'_n \leq d_{min} \ast (n - |V_{kq}|)$$

We can compute $d'_n$ for each valid group size of $n$. If Eq. (4) holds for all values of $n$, we terminate the search for $o_q$ as no better group is possible by adding new members to $V_{kq}$. We formalize the above termination process in the following lemma.

**Lemma 1.** Let $o_q$ be a meeting point, $V_{kq}$ be the list of members already included in the process of forming a group. If for all $n$, $d'_n \leq d_{min} \ast (n - |V_{kq}|)$, where $n' \leq n \leq n''$, then no group with a higher ranking score than the current $k^{th}$ best group is possible by including more members to $V_{kq}$, and thus the search can be terminated w.r.t. $o_q$.

**Proof.** Proof is omitted for the brevity of the presentation.
Algorithm 1: Top k-FSSGQ($G, O, n', n'', c, d_m, k$)

1. Initialize a global empty list $L$
2. Initialize a min priority queue $Q$
3. For each meeting point $o_i$ in $O$
   4. Initialize $V_i = \emptyset$
   5. $V_R \leftarrow \text{retrieveFromRtree}(o_i, d_m)$
   6. $V_R \leftarrow \text{pruneUnqualifiedMembers}(V_R, c)$
   7. Q.push($o_i, V_i, V_R, \text{dist}(V_i, \text{get}(0), o_i)$)
   8. generateRankList($Q, L, k$)

Procedure generateRankList($Q, L, k$)

1. While $Q$ is not empty do
   2. If $|V_i| = n'$ or $|V_i| + |V_R| < n'$ then
      3. Continue
   4. Mark all members of $V_R$ as unvisited
   5. While $|V_i| < n''$ and $|V_R| > 0$ do
      6. If there is any unvisited member in $V_R$ then
         7. $v, d_v \leftarrow \text{nextMember}(V_R, o_i)$
      8. Else break
      9. If advanceTerminate($V_i, d_v, n', n'', L, k$) then
         10. break
      11. If $\text{flag} = \text{true}$ then
         12. If $|V_i| \geq n'$ and score $> L.get(k).score$ then
            13. $\text{updateRankList}(L, V_i)$
         14. $Q.push(o_i, V_i, V_R, \text{dist}(V_i, \text{get}(0), o_i))$
         15. $Q.push(o_i, V_i - \{v\}, V_R, \text{dist}(V_i, \text{get}(0), o_i))$
         16. break;
      17. Else
         18. Mark $v$ as visited

4.5 Algorithm

Algorithm 1 provides the pseudo code of the Exact approach. Given a socio-spatial graph $G$ and the set $O$ of meeting points, the Exact approach for the Top $k$-FSSGQ returns a list $L$ of top $k$ groups ranked in increasing order of their socio-spatial scores. The Top $k$-FSSGQ also takes the following input: the minimum $n'$ and the maximum $n''$ allowable group size, the maximum spatial distance, $d_m$, of members from a meeting point, and the minimum acquaintance constraint $c$. Here, the locations of the users are indexed with an R-tree [10].

For each meeting point $o_i$, we initialize an intermediate solution group $V_i$, as empty and a remaining set $V_R$, containing all members within $d_m$ distance from $o_i$ through retrieveFromRtree procedure (Lines 4–5). Members in $V_R$ are sorted in ascending order of their distance to $o_i$. In Line 6, members whose social connectivity do not satisfy the acquaintance constraint $c$ are excluded from $V_R$ using pruneUnqualifiedMembers procedure.

The algorithm works in a best-first manner, where groups are formed by incrementally retrieving users w.r.t. different points. A priority queue $Q$ is maintained, where each entity contains a tuple of a meeting point $o_i$, an intermediate group $V_i$, and remaining set of members $V_R$, for $o_i$. Entities in $Q$ are maintained in ascending order of the minimum spatial distance between $o_i$ and locations of the members in $V_R$. We initially push the list of entities w.r.t. all meeting points in $Q$. Then, we call generateRankList procedure to find the desired groups and ranks.

In generateRankList procedure, the top entity $(o_i, V_i, V_R)$ of queue $Q$ is popped in each iteration. Then an inner loop starts that fetches the next unvisited member $v$ from $V_R$, and check the feasibility of including $v$ in $V_i$. The inner loop breaks when a member is successfully included in $V_i$. Procedure advanceTerminate checks whether exploration of $V_i$ and $V_R$ w.r.t. $o_i$ can generate a group that can be in the final rank list $L$, and returns true when $V_i$ can be pruned in advance according to Lemma 1.

5. HEURISTIC-APPROXIMATE APPROACH

In our Exact approach, we use distance upper bound based advanced termination strategy for pruning. However, since the problem is NP-hard, it may not be scalable for large datasets. Thus, we propose a familiarity constraint satisfaction heuristic function that calculates a lower bound on connectivity of each individual while considering as a potential group member. This heuristic further prunes a larger number of members based on connectivity constraint, which makes it a scalable solution.

5.1 Familiarity Constraint Satisfaction

In our Exact approach, for each meeting point, we include members in the intermediate group $V_i$ from $V_R$, in ascending order of spatial distance without considering their social connections. As a result, when a valid size group is formed, the group may not satisfy the minimum query acquaintance constraint. To overcome this problem, in the member inclusion process from $V_R$ to $V_i$, we prioritize the users having strong social connectivity with the other members in $V_i$, and define a familiarity constraint filtering function to filter out groups that cannot be a result.

Let $|V_{tq}| < n'$, and $f(v, V_{tq})$ be the number of social connections that $v$ already has with other members in $V_{tq}$. If $v$ needs to satisfy constraint $c$, $v$ requires to have an additional $c - f(v, V_{tq})$ connectivity. After including $v$ in $V_{tq}$, additional $n'' - 1 - |V_{tq}|$ members need to be included so that $V_{tq}$ becomes an $n''$ size group. If $c - f(v, V_{tq}) > n'' - 1 - |V_{tq}|$, $v$ cannot have at least $c$ connectivity because the necessary additional connectivity is greater than the number of members to be added. As a result, $c - f(v, V_{tq}) < n'' - 1 - |V_{tq}|$, which can be expressed as $f(v, V_{tq}) \geq c - n'' + 1 + |V_{tq}|$. Also, when $|V_{tq}| \geq n'$, $v$ needs to be connected with $c$ other members so that the group $V_{tq} \cup \{v\}$ can satisfy the familiarity constraint. Formally, we write the familiarity constraint function as:

$$f(v, V_{tq}) \geq \begin{cases} c - n'' + 1 + |V_{tq}|, & \text{when } |V_{tq}| < n' \\ c, & \text{otherwise} \end{cases}$$

(5)
Algorithm. We can slightly modify Algorithm 1 to introduce the above familiarity constraint. The code in Line 20-28 will be executed only if checkFamiliarity(v, Vt) returns true. The procedure checkFamiliarity verifies whether v satisfies the familiarity constraint satisfaction function as discussed. If v fails to satisfy the constraint, v is marked as visited. Next, we explain the working procedure of our Approximate (which also includes all the steps of our Exact approach) with a running example.

5.2 Detailed Steps with an Example

We explain our algorithm with the example in Figure 1. Let \( V = \{a, b, c, d, e\} \) be the set of members and \( O = \{a_1, o_2\} \) be the set of meeting points. We want to find the top 2 groups, i.e., \( k = 2 \) where group size can vary between 3 and 4 (i.e., \( n' = 3, n'' = 4 \)). Each resulting group needs to maintain minimum acquaintance constraint \( c = 1 \). Let \( \alpha = \beta = \gamma = 0.33 \). Figure 1 shows the social connectivity of the users, where the table presents the distance of the users from each meeting point in kms. Let the maximum spatial distance constraint \( d_m \) be 15kms. Initially, for each meeting point we retrieve the members within \( d_m \) distance from the R-tree in increasing order of spatial distances (Line 1.5).

Figure 2 presents the step-by-step illustration of the member exploration while forming groups. Each state (node in tree) is marked with a number denoting the sequence of exploration step. The users within \( d_m \) distance from \( o_1 \) and \( o_2 \) are \{b, a, e, d, c\} and \{c, e, a, d, b\}, respectively, where the users are sorted in ascending order of their distances from the corresponding meeting point.

We explore the groups in a best first manner with a min-priority queue \( Q \), where \( Q \) is maintained based on the minimum distance between the meeting point \( o_i \) and the locations of the members in \( V_{R} \). Initially, there are two entries in \( Q \): \((o_1, \{c, e, a, d, b\}, 6)\) and \((o_2, \{b, a, e, d, c\}, 2)\) (Refer to Section 4.5 for entries in \( Q \)). The entry \((o_2, \{b, a, e, d, c\}, 2)\) is dequeued first from \( Q \) based on the minimum distance. Let, we begin exploring from the member \( b \). This state of exploration is shown in Figure 2 (marked as 1). As the terminating conditions are not satisfied, according to Lines 1.25-1.26, the entries \((o_2, \{b, a, e, d, c\}, 3)\) and \((o_1, \{c, e, a, d\}, 3)\) are then pushed to \( Q \) for further exploration. The subsequent explorations are shown in Figure 2.

From Figure 2, we see that less number of groups are explored for \( o_1 \) than \( o_2 \). For \( o_1 \), all of the subsequent states of \( \{c, e\}, \{c, d\}, \{e, d\} \) and \( \{c, a, d\} \) are pruned by early termination (Lemma 1). Similarly, the state \( \{a\} \) and its subsequent states are also not generated for the same reason. State \( \{e, a, d\} \) is filtered by familiarity constraint function (Subsection 5.1) and its subsequent states are pruned due to early termination. For meeting point \( o_2 \), Exact approach stops generating unnecessary states, i.e., state \{d\} or \{e\} is not generated since it cannot produce any group satisfying the minimum size constraint. Familiarity constraint satisfaction function stops generating some states, for example \( \{b, e, c\}, \{e, c, d\}, \{a, e, d\} \). At the end of the process, group \( \{b, a, c, d\} \) and \( \{b, a, e, d\} \) are the top 2 groups with score 83.6 and 79.2 respectively.

6. A FAST APPROXIMATE APPROACH

In the Exact approach, we defined upper bounds and the terminating condition based on all the remaining set of members to ensure that all feasible groups are considered, hence the Exact approach needs to explore a large number of members. To expedite the group search further, we propose a Fast Approximate (FA) approach. The key idea is to develop an upper bound on spatial distance and a lower bound on social connectivity of a member (in contrast to all members in the exact approach) to be included in a feasible group. Based on the bounds, we early terminate when there is no remaining member who can increase the group rank. Moreover, we impose a strict familiarity constraint, i.e., we only include a member if her connectivity with the existing members of the group is greater than the expected connectivity of the group (cf. Section 6.4). In this approach, we only include a member to the initially formed group if it results in a higher ranked group. Let, for a meeting point \( o_0 \) we have got an initial feasible group \( V_{i} \) where \(|V_{i}| \geq n'\). A member \( v \in V_{i} \) will be included in \( V_{i} \) if \( V_{i} \cup \{v\} \) has a higher rank score than \( V_{i} \). Similar to the exact approach (Section 4), we derive the gains in social score, spatial score, and the group size score for a new member as follows.

Social score gain: Let \( f_s \) be the total social connectivity of members in \( V_{i} \) and \( \delta_{S} \) be the additional social connectivity if a new member \( v \) is included in \( V_{i} \). Thus, the social score gain of the group \( V_{i} \cup \{v\} \) can be expressed as follows.

\[
\Delta S_{Sc} = f_s + \delta_f - \frac{f_s}{|V_{i}| + 1} - \frac{f_s}{|V_{i} - 1|} = 1 - \frac{f_s}{|V_{i} + 1|} - \frac{f_s}{|V_{i} - 1|}
\]

Spatial score gain: Let \( d_{c} \) be the aggregate spatial distance from members in \( V_{i} \) to \( o_{j} \) and \( \delta_{d} \) be the additional spatial distance between new member \( v \) in \( V_{i} \) and \( o_{j} \). The spatial score gain is:

\[
\Delta S_{Sp} = \frac{1}{|V_{i} + 1|} - \frac{1}{|V_{i} - 1|} = \frac{d_{c}}{|V_{i} + 1|} - \frac{d_{c}}{|V_{i} - 1|}
\]

Group score gain: If we include a new user \( v \) to \( V_{i} \), the group size will increase by one. Thus, the group size score gain is:

\[
\Delta S_{gs} = |V_{i} + 1| - |V_{i} - 1| = \frac{1}{n''} - \frac{1}{n'}
\]

Total score gain: To ensure that the new group \( V_{i} \cup \{v\} \) has a higher rank score than the initial group \( V_{i} \), the summation of the gains of the above three scores must be positive.

\[
\alpha \cdot \Delta S_{Sc} + \beta \cdot \Delta S_{Sp} + \gamma \cdot \Delta S_{gs} > 0
\]

\[
\Rightarrow \frac{\alpha}{|V_{i} + 1|} \left( \delta_l - \frac{2f_{s}}{|V_{i} - 1|} \right) + \frac{\beta}{|V_{i} + 1|} \left( \delta_{d} - \frac{d_{c}}{|V_{i} - 1|} \right) + \frac{\gamma}{n''} \left( \delta_{d} - \frac{d_{c}}{|V_{i} + 1|} \right) > 0
\]

\[
\Rightarrow \frac{d_{m}}{|V_{i} + 1|} \left( \frac{\alpha}{|V_{i} + 1|} \delta_l - \frac{2f_{s}}{|V_{i} - 1|} \right) + \frac{\beta}{|V_{i} + 1|} \left( \delta_{d} - \frac{d_{c}}{|V_{i} - 1|} \right) + \frac{\gamma}{n''} \left( \delta_{d} - \frac{d_{c}}{|V_{i} + 1|} \right) > 0
\]

\[
\Rightarrow \frac{d_{m}}{|V_{i} + 1|} \left( \delta_{l} - \frac{2f_{s}}{|V_{i} - 1|} \right) + \frac{\beta}{|V_{i} + 1|} \left( \delta_{d} - \frac{d_{c}}{|V_{i} - 1|} \right) + \frac{\gamma}{n''} \left( \delta_{d} - \frac{d_{c}}{|V_{i} + 1|} \right) > 0
\]
6.1 Distance Upper Bound for an User

Based on the above formulation, we can derive a distance bound for a user \( v \). In Eq. (6), we have two unknown variables, \( \delta_f \) and \( \delta_l \). As the new group must satisfy constraint \( c \), if we add \( v \) (from \( V_{R_u} \)) to \( V_q \), the connectivity of group \( V_q \) will increase at least by \( 2 \cdot c \). Thus, we replace \( \delta_f \) with \( 2 \cdot c \) in Eq. (6). Let left hand side (L.H.S) of Eq. (6) be \( d^{\uparrow} \). Thus, we get \( d^{\uparrow} > \delta_d \). Therefore, \( d^{\uparrow} \) is the upper bound on spatial distance of a user \( v \) to be considered as a candidate member in \( V_q \) w.r.t. \( \alpha_q \). Formally,

\[
\frac{d_m * (|V_q| + 1)}{\beta} \left( \frac{\alpha}{|V_q| * (|V_q| + 1)} \right)^{2 \cdot c} - \frac{2f}{(|V_q| - 1)} + \frac{\gamma}{n^2} + \frac{d_c}{|V_q|} = d^{\uparrow}
\]

(7)

Here, if \( d(v, \alpha_q) < d^{\uparrow} \), the new group \( V_q \cup \{ v \} \) will have higher score than that of the previous group.

**Lemma 2.** For a \( v \in V_{R_u} \), if \( |V_q| \geq n' \), \( d(v, \alpha_q) < d^{\uparrow} \), and \( f(v, V_q \cup \{ v \}) \geq c \), then \( V_q \cup \{ v \} \) guarantees a higher scoring group than the current group \( V_q \).

6.2 Lower Bound Social Connectivity of a User

When \( v \in V_{R_u} \) satisfies Lemma 2, \( V_q \cup \{ v \} \) becomes higher scoring than \( V_q \). However, if \( v \) cannot satisfy Lemma 2, \( V_q \cup \{ v \} \) can still have higher score as \( v \) can have more than \( c \) connection with members of \( V_q \) (as opposed to our previous assumption that \( v \) can have \( c \) social connection within \( V_q \)). We can re-write our score gain formulation, and can get the following equation for \( \delta_f \).

\[
\alpha \cdot \Delta S_{ac} + \beta \cdot \Delta S_{sp} + \gamma \cdot \Delta S_{qs} > 0
\]

\[
\Rightarrow \frac{\alpha}{|V_q| * (|V_q| + 1)} \left( \delta_f - \frac{2f}{(|V_q| - 1)} \right) + \frac{\gamma}{n^2} + \frac{d_m}{(|V_q| + 1)} > 0
\]

\[
\Rightarrow \delta_f > \frac{|V_q| * (|V_q| + 1)}{\alpha} \left( \frac{d_m}{(|V_q| + 1)} \right) \left( \frac{\alpha}{|V_q|} \right)^{2 \cdot f}
\]

(8)

\[
\left( \delta_d - \frac{d_c}{|V_q|} \right) - \frac{\gamma}{n^2} + \frac{2f}{(|V_q| - 1)}
\]

Here, we put \( \delta_d = d(v, \alpha_q) \), which is the lowest possible \( \delta_d \) as \( v \) has the minimum distance in \( V_q \) from \( \alpha_q \), to get the lower bound on social connectivity \( f^{\downarrow} \) from the R.H.S of the above equation. Thus, \( \delta_f > f^{\downarrow} \). Eventually, if including \( v \) in \( V_q \) results in increased social connectivity more than \( f^{\downarrow} \), \( V_q \cup \{ v \} \) will have higher score than \( V_q \). Since \( f(v, V_q) \) denotes the number of connectivity of \( v \) in \( V_q \), we have \( \delta_f = 2 \cdot f(v, V_q) \). Formally,

\[
f^{\downarrow} = \frac{|V_q| * (|V_q| + 1)}{\alpha} \left( \frac{d_m}{(|V_q| + 1)} \right) \left( \frac{\alpha}{|V_q|} \right)^{2 \cdot f}
\]

(8)

\[
\left( d(v, \alpha_q) - \frac{d_c}{|V_q|} \right) - \frac{\gamma}{n^2} + \frac{2f}{(|V_q| - 1)}
\]

**Lemma 3.** If \( |V_q| \geq n' \) and \( 2 \cdot f(v, V_q) > f^{\downarrow} \), \( V_q \cup \{ v \} \) guarantees a higher scoring group than the current group \( V_q \).

**Proof.** Let \( d(v, \alpha_q) \) be the minimum spatial distance between any \( v \in V_{R_u} \) to \( \alpha_q \). We compute the lower bound on social connectivity \( f^{\downarrow} \) by putting the value \( \delta_d = d(v, \alpha_q) \) in Eq. (8). Since \( v \) has the minimum spatial distance \( d(v, \alpha_q) \) to \( \alpha_q \), the social connection of \( v \) in \( V_q \) must be greater than the lower bound of social connectivity to guarantee that the score of \( V_q \cup \{ v \} \) is greater than current \( V_q \). As a new connection increases the social connectivity of the graph by two, if \( 2 \cdot f(v, V_q) \) is greater than \( f^{\downarrow} \), then \( V_q \cup \{ v \} \) is guaranteed to be a higher scoring group than \( V_q \).

6.3 Early Termination using Distance Bound

According to Lemma 2, we can decide whether we should add a member \( v \in V_{R_u} \) to \( V_q \). To terminate our search for any potential members in \( V_{R_u} \), we have to ensure that no other members in \( V_{R_u} \) can generate any better group. However, checking such constraint for each member in \( V_{R_u} \) is computationally expensive. To overcome this, we compute an upper bound on spatial distance based on the social connectivity assumption that the member \( v \) in \( V_{R_u} \), which is considered to be included in \( V_q \), will be socially connected to every other members of \( V_q \). Moreover, if \( maxdeg \) denotes the maximum degree of the initial set \( V_{R_u} \) of all members, then a member cannot be connected to more than \( maxdeg \) members. Thus, we get \( f_{min} = \min(maxdeg, |V_q|) \). We put \( \delta_f = 2 \cdot f_{min} \) in Eq. (6), and get the L.H.S of Eq. (6) as \( d^{\downarrow} \). Hence, \( d^{\downarrow} > \delta_d \). Therefore \( d^{\downarrow} \) is the upper bound on spatial distance for a member in \( V_{R_u} \) w.r.t. \( \alpha_q \). Formally, we get \( d^{\downarrow} \) as follows.

\[
\frac{d_m * (|V_q| + 1)}{\beta} \left( \frac{\alpha}{|V_q| * (|V_q| + 1)} \right)^{2 \cdot f_{min} - 1} - \frac{2f_c}{(|V_q| - 1)} + \frac{\gamma}{n^2} + \frac{d_c}{|V_q|} = d^{\downarrow}
\]

(9)

We fetch members from \( V_{R_u} \) in increasing order of their spatial distance from \( \alpha_q \). Let \( v \in V_{R_u} \) be the member with the minimum spatial distance from \( \alpha_q \). Then \( v \) cannot provide any higher rank score if \( d(v, \alpha_q) > d^{\downarrow} \).

**Lemma 4.** Let \( v \in V_{R_u} \) be the next fetched member from \( V_{R_u} \). If \( |V_q| \geq n' \), \( d(v, \alpha_q) > d^{\downarrow} \), the search for a better group w.r.t. \( \alpha_q \) can be terminated as including any member from \( V_{R_u} \) to \( V_q \) does not result in a higher scoring group.

**Proof.** Any member from \( V_{R_u} \) is expected to have spatial distance to \( \alpha_q \) less than \( d^{\downarrow} \) so that including one more member from \( V_{R_u} \) to \( V_q \) ensures positive score gain. Let \( v \) be the next retrieved member from \( V_{R_u} \). As the members are retrieved in increasing order of their distances from \( \alpha_q \), \( v \) has the minimum distance from \( \alpha_q \) than any other member in \( V_{R_u} \). If \( d(v, \alpha_q) > d^{\downarrow} \), then no other subsequent member in \( V_{R_u} \) can have a distance less than \( d^{\downarrow} \). Thus we can safely terminate as no better scoring group can be formed.

6.4 A Heuristic for Familiarity Constraint

To expedite the group search process in the fast approximate approach, we also propose a heuristic that prioritizes members with higher social connectivity to the intermediate solution group. Since the minimum group size is \( n' \), each member in \( n' \) size group must be connected with at least \( c \) members from the remaining \( n' - 1 \) members to satisfy the acquaintance constraint. According to unitary method, a member \( v \in V_{R_u} \) will be included in \( V_q \) if \( v \) has social connectivity with at least \( \frac{c}{n'(n'-1)} \) members. Thus we get the social connectivity of member \( v \) in \( V_{R_u} \), \( f(v, V_{R_u}) \) \( \geq \frac{c}{n'(n'-1)} \) when \( |V_q| < n' \). On the other hand, when \( |V_q| \geq n' \), \( v \) must know \( c \) members to form a group \( V_q \cup \{ v \} \) that satisfies the minimum acquaintance constraint. In summary, we can express our strict familiarity constraint satisfaction function as follows.

\[
f(v, V_q) \geq \frac{c|V_q|}{n'-1} \quad |V_q| < n'
\]

(10)
6.5 Algorithm

In the FA approach, we incrementally fetch members from $V_0$ in increasing order of their distances from a meeting point, and include a member to the intermediate solution group $V_i$, when the member satisfies the strict familiarity constraint function. Once the size of $V_i$ reaches $n'$, we compute both spatial and social bounds to decide on whether we can include more members to the group to form higher scoring groups. The steps of the algorithm is quite similar to the Exact algorithm. We need to make the following changes in Algorithm 1: (i) In the \texttt{advanceTerminate} procedure (Line 19), we need to incorporate the early termination condition as prescribed in Lemma 4. (ii) The block (Line 20 - 28) is executed only when $|V_i| < n'$ is true, otherwise when either $d_u < d_v + \gamma$ (Lemma 2) or $2 \times f(v, V_i) > f^\text{opt}$ (Lemma 3) is true.

6.6 Approximation Ratio

In this section, we derive a theoretical bound on the approximation ratio of our approximate approach. We compute the ratio as the score of a group retrieved by our \texttt{Fast Approximation} algorithm divided by the score of the best possible group which might be missed by our algorithm in the worst case scenario.

In our approach, the set of unexplored members, $V_0$ is sorted according to distance of the members from a meeting point. Let $G$ be a group retrieved from $V_0$, where $d_1$ and $d_2$ are the nearest and the farthest distances of members from the meeting point $o$, respectively. As per Equation 1, the score of group $G$ w.r.t. $\alpha$, $S(G, o)$ (or simply $S(G)$), will be the lowest when each member in $G$ has a social connectivity of exactly $\gamma$ (the lowest connectivity) and distance $d_1$ from $o$. Let us denote such lowest scoring group (sub-optimal) as $G_{opt}$. Similarly, the highest scoring group is formed when each member is connected to every other member in the group and has a distance of $d_n$ from the meeting point. Let us denote this group (optimal) as $G_{opt}$. Let us assume that the scores of $G_{opt}$ and $G_{opt}$ are $n$ and $n_{opt}$, respectively. Since we assume that members of $G_{opt}$ are at $d_1$ distance, and $G_{opt}$ are at $d_m$ distance, and our algorithm retrieves members in order of distance, we can retrieve members of $G_{opt}$ instead of members of $G_{opt}$ if $d_{opt} = d_L$.

We compute the score of each of the three constituents of our $G_{opt}$ and $G_{opt}$ as follows:

\[
\text{Score} = \frac{2 \times n \times s/2 \times m(n - 1)}{n - 1} = \gamma \times \frac{\alpha}{n} + \beta \times \frac{(1 - d_m)}{n} + \gamma \times \frac{n}{n^*},
\]

Hence, we have the following scores: $S(G_{opt}) = \alpha \times \frac{n}{n - 1} + \beta \times \frac{(1 - d_m)}{n} + \gamma \times \frac{n}{n^*}$, and $S(G_{opt}) = \alpha \times 1 + \beta \times (1 - \frac{d_m}{n}) + \gamma \times \frac{n_{opt}}{n^*}$

If we set $n_{opt} = n^*$ (the maximum group size), we get the maximum possible score of $S(G_{opt})$ as: $S(G_{opt}) = \alpha \times 1 + \beta \times (1 - \frac{d_m}{n^*}) + \gamma \times 1$. Hence, for any group returned by the FA algorithm, the approximation ratio will be bounded by the following value:

\[
S(G_{opt})
\]

We also show the approx. ratio bound for different scenarios:

\[
\begin{array}{ccc}
\text{Emphasis} & \text{Weights} & \text{Approximation ratio} \\
\text{Social score} & \alpha = 1, \beta = 0 & \frac{n}{n^*} \\
\text{Spatial score} & \beta = 1, \alpha = 0 & 1 \\
\text{Size score} & \gamma = 1, \alpha = \beta = 0 & \frac{n^*}{n^*}
\end{array}
\]

6.7 A Greedy Approximation Approach

To expedite the search, we propose a greedy approximation approach that works as our baseline for the approximate approach. In the greedy approximation approach, we avoid the backtracking, and only progressively add members from $V_0$ to $V_i$ if they satisfy familiarity and other constraints. Thus, when a member is included in $V_i$ we do not exclude it for forming other possible groups.

7. EXPERIMENTAL EVALUATION

In this section we present the experimental evaluation for the baseline and our proposed approaches to answer the Top-$k$-FSSGQ queries. Specifically, we compare the performance among the following five methods: (i) the baseline (B) as presented in Section 3, (ii) the exact approach (E) (iii) the approximate approach (A), (iv) the fast approximate approach (FA), and (v) the greedy approximate approach (GA) as presented in Section 6.7.

7.1 Experimental Settings

These algorithms are implemented in Java and run on a server with Intel Xeon E5-2630, 6 cores X 2 threads per core @2.3 Ghz, 15360 kB of cache and 256 GB of RAM.

Dataset. We conduct extensive experiments with three real datasets (i) Brightkite [17], (ii) Gowalla [17], and (iii) Twitter [1].

Brightkite and Gowalla, each contains the social connections of the users and their check-in locations. As a user may have multiple check-ins, we consider the most frequent check-in location as the location of that user. If a user does not have any check-in, that user along with her social connections is discarded. The meeting point locations are generated by using the same distributions of check-in locations, i.e., the locations with higher check-ins have higher chance of selecting as meeting locations. Table 2 contains the details information of the datasets after applying these processing.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>#Nodes</th>
<th>#Edges</th>
<th>Check-ins</th>
<th>Time Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brightkite</td>
<td>58,228</td>
<td>214,078</td>
<td>4,491,143</td>
<td>Apr 08 - Oct 10</td>
</tr>
<tr>
<td>Gowalla</td>
<td>196,586</td>
<td>950,327</td>
<td>6,442,890</td>
<td>Feb 09 - Oct 10</td>
</tr>
<tr>
<td>Twitter</td>
<td>10M</td>
<td>84,744,091</td>
<td>-</td>
<td>May 11</td>
</tr>
</tbody>
</table>

The intuition behind selecting the meeting points in this way is, if many people frequently visit a place, the place is more likely to be their meeting point in real life. We also ran experiments by randomly selecting the meeting points with uniform distribution. However, in that case, there are many instances where no valid group is formed due to the sparsity of check-ins. Therefore, we exclude such experiments results.

The Twitter dataset contains the ‘follow’ relations among users and the locations of users in their profiles. As only a fraction of the users (appx. 1.5 millions of 10 millions) have their meaningful locations mentioned in the profile, we generate the locations of the other users following the same distribution. We consider the ‘follow’ relationship as an undirected social connection. We use this augmented twitter dataset to show the scalability of our approaches.

Evaluation Metrics and Parameters. We evaluate the efficiency, scalability, and effectiveness of our algorithms by varying different parameters. The list of parameters with their ranges and default values are shown in Table 3. For all experiments, a single parameter is varied while keeping the rest at default. To determine efficiency and scalability, we study (i) total runtime and (ii) total number of
of relevant instances. We measure the recall as ratio of the $k^{th}$ group rank in top $k$-A and the rank of the same group using the $E$ approach. For example, when $k = 16$, if $16^{th}$ group in top 16-A appear as the $20^{th}$ ranked group in $E$, recall is $16/24 = 0.67$.  

(iv) Percentage of user overlap. We also compare the percentage of user overlaps in two rank lists. For example, when $k = 16$, if the same set of users appear in both top 16-A and top 16-E, then percentage of user overlap will be 100%.

7.2 Performance evaluation

7.2.1 Varying Minimum Group Size, $n'$

Efficiency and scalability evaluation: Figure 3 shows the effect of varying $n'$ for Brightkite and Gowalla. The runtime of the baseline, the exact approach, and the approximate approach gradually increase with $n'$, as more groups are likely to be explored for a higher $n'$. The runtime of $FA$ and $GA$ does not vary much, as these processes can terminate earlier by exploring less number of groups. As shown in Figure 3, on average $E$ takes 3.67 times and 9.3 times less run time than $B$ in Gowalla and Brightkite, respectively. The approximate approach runs 84 times and 12.67 times (on average) faster than $B$ in Gowalla and Brightkite, respectively. $FA$ runs 737 times and 23.56 times (on average) faster than $B$ in Gowalla and Brightkite, respectively. The number of nodes explored also shows similar trend. As $GA$ avoids backtracking, the approach is faster than the other approximate approaches. Since the density of members around meeting points is higher in Gowalla than in Brightkite, the number of explored nodes w.r.t. the same group size is larger in Gowalla than that of Brightkite.

Effectiveness Evaluation: Figure 4 presents the effectiveness for varying $n'$. For each $n'$, we have nine values (shown with bars). The first three values denote the percentage of the groups of top $k$-Approximate (topk-A) appearing in top $k$-Exact (topk-E), $1.5 * k$-$E$, and top $2 * k$-$E$ rank list, respectively. We also compare the percentage of user overlaps in two rank lists. For example, when $k = 16$, if the same set of users appear in both top 16-A and top 16-E, then percentage of user overlap will be 100%.

Figure 3: Effect on varying $n'$

Figure 4: Effectiveness for varying $n'$

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min group size ($n'$)</td>
<td>4,5,6,7</td>
<td>6</td>
</tr>
<tr>
<td>Max group size ($n''$)</td>
<td>6,7,8,9</td>
<td>8</td>
</tr>
<tr>
<td>Min acquaintance</td>
<td>2.3,4,5</td>
<td>3</td>
</tr>
<tr>
<td>Max distance constraint</td>
<td>16,20,24,28</td>
<td>20</td>
</tr>
<tr>
<td>$\alpha, \beta, \gamma$</td>
<td>[0, 1]</td>
<td>.33</td>
</tr>
<tr>
<td>No. of meeting points</td>
<td>50,100,200,400</td>
<td>100</td>
</tr>
<tr>
<td>$k$</td>
<td>4,8,16,32</td>
<td>8</td>
</tr>
</tbody>
</table>
7.2.2 Varying Meeting Points

Efficiency and scalability: As the search space increases with the increase of $|O|$, the costs for the baseline and exact approach increase (Figure 5). The costs of the approximate algorithms do not vary much as many calculations are avoided by considering bounds on individual members than the group. The benefit of our approaches are higher for higher $|O|$. Similar pattern is seen w.r.t. number of nodes explored.

Effectiveness: Figure 7 shows the percentage of group appearance when we vary $|O|$. Both A and FA demonstrate a very high effectiveness for both datasets. GA has a very low percentage (0%) for Gowalla but high percentage (84%) for Brightkite. GA produces similar results while varying other parameters too.

From these experiments we consistently find that, although GA has almost constant efficiency, the effectiveness of GA is not competitive in many cases.

7.2.3 Varying Maximum Distance Threshold ($d_m$)

As more users become eligible to be included in the result with the increase of $d_m$, costs increase rapidly with the increase in $d_m$.
and $\beta > \gamma$, both the processing time and the number of nodes explored decrease. Since we explore members based on the increasing order of spatial distance w.r.t. meeting points, it is expected that the nearby groups are found quickly for a large $\beta$.

### 7.2.6 Experiments with Twitter Dataset

To show our scalability, we use augmented Twitter dataset of 10 million users, and run the experiments by varying different parameters (Figure 10(a) - (d)). Due to space constraints, we have only shown the number of nodes explored (which shows similar trends to the required time). We have observed similar performance improvement that we observed in other datasets. However, we have also observed few exceptions: e.g., in Figure 10(a), the results do not change much for varying $n'$, the number of nodes (also the time) even start to decrease after $n' = 6$. The reason is that, on average, each user has only a few connections in this dataset. Thus, if $n'$ is fixed to a large value, not many groups can satisfy that constraint. However, A, FA, and GA require about 5.5, 15, and 110 times less node exploration than the baseline.

### 8. CONCLUSIONS

We have proposed a novel Top k Flexible Socio Spatial Group Query (Top k-FSSGQ) to find the top $k$ groups of various sizes w.r.t. multiple POIs. To incorporate the trade-offs among different socio-economic factors, we have devised a ranking function by combining social closeness, spatial distance, and group size, which provides the flexibility of choosing groups of different sizes under different constraints. To effectively process Top k-FSSGQ, we have first developed an Exact approach that ensures early termination of the search based on the computed upper bound distance. We have proved that the problem is NP-hard, and thus we have designed a Fast Approximate approach based on the relaxed bound and strict social connectivity constraint, which is much faster than the exact solution by sacrificing the quality slightly. We have conducted detailed experimental studies with three popular real-world datasets and shown that the Exact approach runs up to one order magnitude faster than the baseline, and the Fast Approximate approach runs up to two orders of magnitude faster than the Exact approach and returns the same set of groups in most of the cases.

**Acknowledgement.** This work was conducted at BUET and partially supported by Australian Research Council DP160102114.
9. REFERENCES

[Online: accessed 27-03-2018].


