IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 19, 2018, accepted April 21, 2018, date of publication April 25, 2018, date of current version May 16, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2830300

Clustered Crowd GPS for Privacy Valuing Active Localization

FATIH YUCEL, (Member, IEEE), AND EYUPHAN BULUT, (Member, IEEE)
Department of Computer Science, Virginia Commonwealth University, Richmond, VA 23284, USA
Corresponding author: Eyuphan Bulut (ebulut@vcu.edu)

ABSTRACT With the proliferation of mobile devices having BLE capability and the introduction of Beacon technology, crowdsourcing-based approaches have recently emerged as a promising solution for localization of lost objects or individuals (e.g., children or elders). By attaching affordable Beacon tags to them, objects of care could be tracked and localized by user devices in the proximity. While crowd GPS service has gained popularity recently, it has not extended beyond passive mode in which localization is achieved in the background without intruding the mobility of users. In this paper, we study the localization of lost objects through the crowd GPS service in an active manner. We propose clustering users in a Beacon tag network based on the benefits they can receive from each other in terms of the localization of their lost items. A new metric is developed to quantify this benefit and the users that can provide most of the total possible benefits to each other are then grouped together so that they can provide active localization service for only the users most beneficial to them. The clustering of users is achieved based on both a greedy heuristic based algorithm and a genetic algorithm. Extensive simulation results are conducted utilizing both synthetic data and real location based social network datasets. The results show the effective partitioning of the users under different user counts and groups while valuing the privacy of users at its maximum by limiting the number of interactions between users.

INDEX TERMS BLE tags, crowd GPS, location tracking, lost item tracking, clustering.

I. INTRODUCTION

As human beings, we lose or misplace various of our belongings every day such as mobile phones, keys and sunglasses. However, searching for them could be time consuming (i.e., 15-20 minutes per day) as shown by several reports [1]. When we lose things outside rather than indoors, the localization task could be more challenging since the search area gets larger. Moreover, when what we lose is our loved ones such as pets [2] and vulnerable individuals like children [3] and elders [4], the process of finding torments us more due to involvement of emotions.

Recently, with the widespread adoption of smartphones, crowdsourcing based solutions have been provided for this challenging task. Harnessing the power of a mass of geographically dispersed user devices with Internet connectivity, an invaluable crowd GPS service is developed towards the goal of localizing lost items. With the release of Bluetooth Low Energy (BLE) that allows low-cost near-range communication [5], several vendors have created tiny, battery-powered, BLE tags that can easily be attached to the objects of care. Today, there are communities of over several millions of these devices managed by several vendors [6], [7].

These BLE-enabled (or beacon) tags connect to the users' mobile devices in the vicinity via BLE and send periodic updates (i.e., beacons) to the device to indicate its presence. However, such updates do not reach to the device when the distance between the device and tags is more than the limited range of BLE communication. With the crowd GPS service, collaboration of multiple user devices is targeted to achieve an enhanced coverage for the localization of lost items. That is, when an item with a tag attached is lost, any nearby user device in BLE range could detect it in a transparent way and notify the server and eventually the user who owns it with the current location information.

Clearly, the benefit of such a collaborative sensing system will be pronounced with increasing number of users participating in the system. Moreover, the success in areas with more user density will boost compared the other areas. Despite the popularity of crowd GPS, due to its design, its benefit to localization of lost items does not extend beyond passive localization by the participating devices. That is, the localization of lost tags is achieved in the background without intruding the mobility of users. If the user by chance passes by the lost tag, the observation is sent to the server transparent to the user. While such a design provides a service without disturbing users, the benefit stays limited as it is not controllable.

An interesting approach, which could potentially extend the benefit of such a system, is proposed by Locus Pocus [8], which aims to monetize the service of searching of the lost objects by charging a fee. Such an approach could potentially trigger active participation of users in this process. For example, users who are in the areas where the lost item has been recently seen, could change their mobility for sometime and look for it more actively depending on the price offered. The concept could be considered under the umbrella of a specific type of crowdsourcing called *spatial crowdsourcing*. Unlike traditional crowdsourcing such as image tagging, where tasks can be performed anywhere, in the context of spatial crowdsourcing, workers need to move to a specified location physically and perform a task (e.g., taking a photo of a point of interest (POI)) before the expiration time in order to successfully perform a task [9].

In the spatial crowdsourcing context, the main focus is the optimal matching of workers and tasks [10] while giving priorities to several different factors (e.g., privacy, minimization of the cost). Thus, in the context of crowd GPS which has been considered in non-intrusive or passive manner so far, if an active localization model will be employed, similar to other spatial crowdsourcing models, incentives should also be provided to users for their efforts. For some specific cases like finding missing children, users could be motivated to voluntarily participate in such a system. Otherwise, providing fee based incentives may contradict with the motivation of masses for adopting crowd GPS, as it is free of charge after the Beacon tags are purchased. In order to design such a complimentary service while utilizing active spatial crowdsourcing based localization of lost items, we propose to form groups of users depending on their historical visit patterns and let each member of the group benefit from other members freely. This can provide mutually similar benefit among group members and release the burden of providing incentives required in spatial crowdsourcing based task assignments. Note that while users initiate tasks (e.g., find my lost item in this area), they directly interact with other users in their group and notify them about their private location information. Thus, group sizes should be determined valuing the privacy of users at its maximum and inclusion of a user in a group should be allowed only if the improvement in network level benefit is worth the additional exposure of that user's privacy.

In this paper, we study the finding of lost items with the collaboration of users in an active manner. The very preliminary version of this study is published in [11], in which only the pairs of nodes are found to help each other towards finding their lost belongings. Such an approach provides very limited benefit to users and does not consider network level optimization. In this paper, our goal is to develop an intelligent crowd GPS solution for complimentary active localization service while keeping the number of user interactions as minimum as possible, thus valuing the user privacy at its maximum. Our contributions are (i) analyzing the users' visit patterns at a location and developing a metric that can quantify their potential benefits in terms of finding others' lost items, (ii) identifying the groups of users that can provide high mutual benefit to each other for a cost-free active localization service based on both a greedy heuristic based algorithm and an evolutionary genetic algorithm, and (iii) performing extensive simulations with both synthetic data and two different location based social network dataset to compare the effectiveness of grouping algorithms under different set of parameters.

The rest of the paper is structured as follows. In Section II, the related work is discussed. In Section III, first, analyzing the relation between the visit patterns of different users at a location, a new metric is defined to quantify the benefit of users to each other. Then, the proposed clustering algorithms are discussed. In Section IV, evaluation of the proposed system using simulations based on synthetic data and real location based social network traces is presented. Finally, concluding remarks are discussed in Section V.

II. RELATED WORK

Localization of people's belongings through the sensors on mobile devices has recently been studied under different names such as people centric-sensing [12], participatory sensing and mobile crowd sensing [13], [14]. Especially with the proliferation of smartphones that are equipped with multiple sensors, the need for deploying and maintaining separate dedicated sensors for such kind of service is invalidated.

With the release of Bluetooth Low Energy (BLE) technology, the smartphones have received the capability of communication with nearby devices with lower consumption at lower data rates. As a result, high power savings (i.e., 60-80%) achieved compared to previous technologies [15]. After Apple devised the iBeacon standard protocol in 2013 [16], BLE has become more popular and several BLE-enabled products are released by multiple manufacturers.

Beacons tags are BLE devices that can periodically (e.g., every 100 ms in Apple's iBeacon standard) advertise themselves to their surroundings to be discovered by other BLE capable devices. Thanks to the flexibility in packet format, it is also possible to send some limited data during the broadcasting of these advertisement packets without making an actual connection to nearby devices. Compared to the other nearby communication technologies such as QR codes and NFC, Beacons are also more convenient since they require the least interactions with users. Moreover, compared to RFID based localization [17], [18], Beacons are also easy to deploy as most of the smartphones today support BLE technology.

The BLE functionalities and Beacons have recently been used in several applications in different domains such as indoor localization [19], [20] and navigation [21], [22]), ticketing [23], proximity marketing [24] and localization of missing and lost items. In indoor localization, utilizing the RSSI (Received Signal Strength Indication) value of BLE signals, the distance of the items are detected to be able to locate the items accurately. To the point localizations are shown to be possible in recent studies [25], with the integration of tags inside furnitures that makes them searchable. A simple prototype that use beacons for localization of personal items is also implemented and tested in [26]. In the commercial world, this concept has also attracted a lot of interest and several specific devices [7] for various purposes (e.g., tracking of pets and children) have been developed.

There are also some work that study the security aspects of crowd GPS applications and provide efficient and privacy preserving designs. In Techu [27], a privacy preserving system is introduced for Beacon based tracking systems. Rather than a vulnerable centralized design that could be exposed to single point failures, a unique bulletin board based observation posting system is introduced. Users report their observations of tags to a server that could be untrusted, however the actual tag location information is stored locally only in the observers. Once the owner of the tags claims the ownership of the lost tag to the observer directly (i.e. server is not involved in this communication), the location information is disclosed.

In this paper, different than previous work which consider detection of items in passive mode, we study the active localization of items in crowd GPS systems. To this end, we propose clustering of users in a way that only the users that can provide similar high mutual benefit to each other are grouped together so that no incentives are required for active mode localization and the benefit obtained in the network per exposed privacy (i.e., user interactions) is maximized.

III. PROPOSED SYSTEM AND SOLUTION

In this section, we elaborate on the design of the proposed system. We first develop a new metric to quantify the potential benefits of users to each other based on the relation of their visit patterns at a region (i.e., POI). Then, we study the clustering of users for building a complimentary active localization service for lost items.

A. SOCIAL TRACKING DISTANCE METRIC

Let $a = (t_s, t_e, loc_{id})$ denote a visit event by a user a with t_s and t_e denoting the start and end times of the visit and loc_{id} denoting the visited location id. All of the visits of a user at that location could be represented with a set $\mathcal{V}_a$, in which the end time of the previous event is always smaller than the start of the next event.

$$\mathcal{V}_a = \{a_1, a_2, a_3, \ldots, a_n\} \text{ where}$$
$$a_i.t_e < a_{(i+1)}.t_s, \quad \forall i \in \{1..n\}$$

Since a Beacon attached item could only be detected by a mobile device within certain proximity (i.e., preset BLE range), we use a probability to denote the likelihood for the detection of the lost item by the user's mobile device that is in the same location at the current time unit. Moreover, in order to include the increased likelihood of finding the items by owners compared to other users (as they might remember exact spots they lost the item within the location), we use p_s for probability of finding by *self*, and p_o, for probability of finding by *others*.

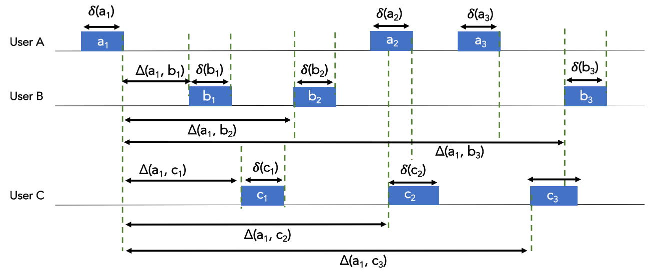

FIGURE 1. Sample visit patterns of three users.

To quantify the benefit of a user B to another user A in terms of finding his/her lost items, we propose a metric called *Social Tracking Distance (STD)*, inspired by the metrics [28], [29] used in analyzing contact patterns in DTNs. Consider the sample visit history of three nodes A, B and C in a location shown in Fig. 1. The visits of each user is shown in a different timeline row. The i^{th} visit of a user, A, is labeled as a_i. We assume that the time is divided into equal time units and the durations of visits are denoted with $\delta(.)$ time units and the time passed since the visit of user A's i^{th} visit to the user B's j^{th} visit is denoted as $\Delta(a_i, b_j)$ time units. Without loss of generality, assume that there are n visits of each user in a specific area. We define the $STD_{(A,B)}$ metric as the average delay that user B's device will sense the lost item (i.e., Beacon attached to the item) of user A. To calculate it, for each possible time unit during the visits of user A, we find the probability of finding in the upcoming visits of all users in the same location and corresponding delay. Then, we find the weighted average of these delays over all possible locations (i.e., POI).

It is assumed that when user A loses an item during a visit, she will notice that she lost the item after she left the area and start a search process in the network. Thus, we assume that the item will not be found during the same visit it is lost. It is possible that when the owner visits the same location later, she may or may not find the item during that visit. The findings during visits by other users will only be considered if the owner or other users in earlier visits do not find the item. User A's device can potentially lose the item at any time during her visit (in range (0, $\delta(a_i)$]) and user B's device can potentially sense the lost item at any time during her visit (in range (0, $\delta(b_i)$]). This results in a range of ($\Delta(a_i, b_j)$, $\Delta(a_i, b_j) + \delta(a_i) + \delta(b_j)$] for the delay of finding the lost item since it is lost. However, the probability of each of these delays is different, and can be calculated using the visit durations and their temporal relations at the same location.

Assume that user A lost an item around x time units before her current visit, a_i, ends in that region. If user B can detect the presence of that item in that region, the average delay of finding it will be:

$$\mathcal{D}(a_i, x) = \sum_{j=s}^{n} \sum_{y=1}^{\delta(b_j)} \left((\Delta(a_i, b_j) + x + y) p_{(b_j,y)}^{(a_i,x)} \right)$$
$$\text{where, } s = \arg\min_k \{a_i.t_e < b_k.t_s\} \quad (1)$$

Here, $p_y^x(a_i, b_j)$ denotes the probability of finding the item, that is lost x time units before the end of a_i visit, at y^{th} time unit of visit b_j. This probability should be calculated considering the visits of other users and the owner of the item in the same region before this time unit as well as the previous visits of user B and previous time units in the current visit of B. The item should not be found in previous time units and should be found in this exact time unit. That is,

$$p_{(b_j,y)}^{a_i} = p_o(1-p_o)^{y-1} \beta_y^{pre}\binom{a_i}{b_j}$$
$$\text{where, } \beta_y^{pre}\binom{a_i}{b_j} = (1-p_o)^{\beta'_y\binom{a_i}{b_j}} (1-p_s)^{\beta_y\binom{a_i}{b_j}}$$

Here, $\beta_y\binom{a_i}{b_j}$ denotes the sum of visit durations of the owner (i.e., user A) and $\beta'_y\binom{a_i}{b_j}$ denotes the sum of visit durations of the other users, between visits a_i and b_j. More formally:

$$\beta_y\binom{a_i}{b_j} = \left(\sum_{\forall k, i \; < k \leq m} \delta(a_k) \right) + \delta'_{b_j}(a_{k+1}) \text{ where,}$$
$$m = \arg\max_k \{a_k.t_e < b_j.t_s + y\}$$
$$\delta'_{b_j}(a_{k+1}) = \max\{0, b_j.t_s + y - a_{k+1}.t_s\}, \quad \text{if } \exists a_{k+1} \in \mathcal{V}_A$$
$$\beta'_y\binom{a_i}{b_j} = \sum_{\forall u \in \mathcal{U}, u \neq A} \left(\left(\sum_{\forall k, l_u \leq k \leq m_u} \delta(u_k) \right) + \delta'(u_{k+1}) \right)$$
$$\text{where, } l_u = \arg\min_k \{a_i.t_e < u_k.t_s\}$$
$$\text{where, } m_u = \arg\max_k \{u_k.t_e < b_j.t_s\}$$

Note that, $\delta'(.)$ is used to denote the part of the visits which did not end yet and can still contribute to the finding of visits.

Having the expected delay formula for finding an item lost at a specific time unit in a visit of user A, we then iterate through all possible losing times to calculate the STD metric:

$$STD_{(A,B)} = \frac{\sum_{i=1}^{n} \left(\sum_{x=1}^{\delta(a_k)} \mathcal{D}(a_c, x) \right)}{P_{(A,B)}}$$

Note that the numerator in the formula above is the sum of products of probabilities and delays, and the denominator is the sum of all probabilities used. This average delay is the expected value of the delay assuming that A's lost item is found by B. The probability that it will be found by B is:

$$P(A, B) = \frac{\sum_{i=1}^{n} \sum_{x=1}^{\delta(a_i)} \left(\sum_{j=s}^{n} \sum_{y=1}^{\delta(b_j)} p_{(b_j,y)}^{(a_i,x)} \right)}{\sum_{i=1}^{n} \delta(a_k)}$$

Simplifying the equations, the general formula for the $STD_{(A,B)}$ could be rewritten in a more structured way as:

$$STD_{(A,B)} = \frac{\sum_{\forall a_i} \left(\delta(a_i) \left(f_1(a_i) + f_2(a_i) + \sum_{j=s}^{n} f_v(b_j) \right) \right)}{\sum_{\forall a_i} \delta(a_i) p(a_i)}$$

where,

$$f_1(a_i) = \sum_{\forall b_j.t_s > a_i.t_e} \Delta(a_i, u_j) p_f^{\delta(u_j)} \beta_0^{pre}\binom{a_i}{b_j}$$
$$p_f^d = 1 - (1-p)^d$$
$$f_2(a_i) = (\delta(a_i) + 1) p(a_i)/2$$
$$p(a_i) = \sum_{j=s}^{n} p_f^{\delta(b_j)} \beta_0^{pre}\binom{a_i}{b_j}$$
$$f_v(b_j) = \left(p_f^{\delta(b_j)}/p + (1 - p_f^{\delta(b_j)}) \delta(b_j) \right) \beta_0^{pre}\binom{a_i}{b_j}$$

Similarly, $P_{(A,B)}$ could be simplified as:

$$P_{(A,B)} = \left(\sum_{\forall a_i} \delta(a_i) p(a_i) \right) / \sum_{\forall a_i} \delta(a_i)$$

STD metric defines the expected delay of finding and is derived from over all scenarios that ends up with item's finding. However, it is possible that the item may not be found during all visits, thus, that probability should be considered in defining the benefit of user B to A. Moreover, the *STD* value for the same pair of nodes can vary at different locations. To accommodate the impact of such differences in the average benefit of users to each other, we define a weighted satisfaction value for user B's efforts in finding the lost items of user A in any of the locations visited by A.

$$\gamma_{(A,B)} = \sum_{\forall r} \left(w_r^A \left(\frac{P_{(A,B)}^r}{STD_{(A,B)}^r} \right) \right) \quad (2)$$

where, w_r^A denotes the weight of the region r (i.e., total visit durations by A in region r within all visit durations in all regions).

We also define the average delay of finding a user's item by any user in the network as follows:

$$STD_A = \frac{\sum_{\forall A} \left(STD_{(A,A')} P_{(A,A')} \sum_{i=1}^{n} \delta(a_i) \right)}{\sum_{\forall A} \left(P(A, A') \sum_{i=1}^{n} \delta(a) \right)} \quad (3)$$

Here, A' notes all other nodes except A in the network. The satisfaction of a user from all other nodes in the network can also be computed using a similar formulation to (2).

B. CLUSTERING OF USERS

Once the satisfaction values of each user from every other user in the network is found, we want to group them such that the users that can mutually benefit from each other similarly are in the same group. This is to ensure that the users in the same group will share the same eagerness for active localization of the other's lost items.

Let the set of users in the network be $X = \{u_1, u_2, \ldots u_N\}$. A group of users, G_i, is a subset of X and the set of all groups is denoted by:

$$\mathcal{G} = \{G_1, G_2, \ldots, G_r\},$$

where,

$$\bigcup_{i=1}^{r} G_i = X \text{ and } G_i \cap G_j = \emptyset \ (i \neq j)$$

Assume that there are R possible locations that these N users visit. These locations could be considered as all potential locations that users visit with some boundaries. Moreover, they can also be considered as the locations of the POIs that users visit more frequently or they have a high likelihood to lose their items. Each node visits all or some of these locations with different durations and frequencies.

Assume that the number of groups is $|\mathcal{G}| = r$. The goal is to find the partitioning of N users into r groups such that the sum of average satisfaction values of each group will be maximized. More formally, the objective function is:

$$\tau(\mathcal{G}) = \max \sum_{\forall G_i \in \mathcal{G}} \left(\frac{\sum_{\forall i,j \in G_i} \gamma_{(i,j)}}{|G_i|(|G_i| - 1)} \right) \quad (4)$$

This objective function can also be interpreted as maximization of total benefit per all user interactions in the network. As only the users in the same group interact with each other for active localization queries, there exists $|G_i|(|G_i| - 1)$ interactions in group i. Taking the average group benefit by dividing with this value and iterating it through all groups yields us the sum of all average group benefits.

Note that dividing a set of n labeled objects into r different non-empty unlabeled subsets is defined by Stirling numbers of the second kind and can be explicitly calculated as:

$$\left\{ {n \atop r} \right\} = \frac{1}{r!} \sum_{j=0}^{r} (-1)^{r-j} \binom{r}{j} j^n$$

It is possible to generate different number of groups between 1 and n, thus, we actually need to try $\sum_{r=1}^{r=n} \left\{ {n \atop r} \right\}$ possible cases (which is defined as n^{th} Bell number, B_n) to find the best group that maximizes the objective function. As this will take too long, in the next subsections, we propose two different clustering algorithms to find the best group.

1) GREEDY ALGORITHM

In order to achieve an algorithm that runs fast, we design a clustering algorithm with a greedy heuristic. The steps of this algorithm is summarized in Algorithm 1. It is assumed that initially there are N different clusters and each cluster consists of a single user. $\mathcal{G} = \{G_1, G_2 \ldots G_{|N|}\}$,

Algorithm 1 Greedy Clustering Algorithm

```
continue = true
while continue do
    maxIncrease = 0
    maxGroupIndices = {0, 0}
    G_max = ∅
    for ∀i ∈ G do
        for ∀j ∈ G and j > i do
            Create a new group G_new = G_i ∪ G_j
            G_temp = G - {G_i, G_j} ∪ G_new
            if (τ(G_temp) - τ(G)) > maxIncrease then
                maxIncrease = τ(G_temp) - τ(G)
                maxGroupIndices = {i, j}
                G_max = G_temp
            end
        end
    end
    if maxIncrease > 0 then
        G ← G_max
    else
        continue = false
    end
end
```

with $G_i = \{u_i\} \subset X$. Then, we try every possible merging of two different groups in current set of groups, $\mathcal{G}$, and find the one that will provide with the maximum possible increase in the objection function, $\tau(\mathcal{G})$. The new set of groups is then obtained by merging these individual groups and next iteration of the same operation is run. If there is no merging possible that will give a positive increase in the objective function, the algorithm stops.

2) EVOLUTIONARY GENETIC ALGORITHM

For the clustering of the users, we also design an evolutionary genetic algorithm, which can provide near-optimal grouping results with comparably fast running times [30], [31]. Each chromosome consists of N numbers where each number indicates the group of the user up to N. For example, a sample chromosome <1, 1, 2, 3, 3> indicates that users (indexes) 1 and 2 will be in the group 1, user 3 will be in group 2 by itself and users 4 and 5 will be in group 3. The crossover function is achieved through standard single point crossover at random locations of two random chromosomes. For mutation operation, we update the group number of a random node with another random group number between 1 and N. The fitness function is set to $\tau(\mathcal{G})$ defined in (4). The list of the parameters used in this algorithm and their corresponding values are shown in Table 1. We have tested several parameter values suggested in the literature [32] and used the best ones.

IV. EVALUATIONS

In this section, we evaluate the performance of the proposed system. To this end, we first elaborate on the simulation environment and then present extensive simulation results.

TABLE 1. Genetic algorithm parameters.

Parameter	Value
Population size	15
Number of generations	100k and 200k
Probability of crossover at each generation	0.9
Probability of mutation at each generation	0.15

TABLE 2. Parameter values used in synthetic data generation.

Parameter	Value
Time frame	7 days
Number of users	100, 250, 500
Number of POIs	10
Visit duration	5-15 min
Average number of visits per user per region	5
Probability of self finding (p_s)	0.2
Probability of finding by other users (p_o)	0.05
Probability of visits in home region	0.7
Probability of visits in other regions	0.3

A. DATA SETS

In order to understand the performance differences of the proposed system, we have built our custom simulator using different data sets. First, we create a synthetic data set of user visits, then we use two different real user traces from location based social network platforms. Next, we provide the details of these datasets.

1) SYNTHETIC DATA

We generate visits of the each node independently from other nodes. In every 10 minutes, a visit is created for a node with a probability of 0.05 (i.e., one visit per 200 minutes in average). The duration of the visits is assigned uniformly and randomly from the range of 5-15 minutes. Then, we decide the region of the visit. Each node has a home region, which is the region it most frequently visits. The probability of the visit being in the home region is 0.7, while for all other regions, the probability is (0.3/(R-1)). This visit generation process continues for a week time (appr. 10,000 minutes, meaning 50 visits for each node in average). Table 2 summarizes the list of parameters used in synthetic data generation together with their values.

2) REAL USER DATA

We use two location-based social network datasets, namely Gowalla and Brightkite datasets [33], to capture the real user visits at different locations. We specifically focus on the check-ins that are reported in San Francisco area in these datasets. The user check-ins in the dataset are considered as the visit start times of users.

However, user check-outs were not available for the locations, thus, we decide visit durations uniformly from a range of [5-15] minutes. We find the top 20 locations in the area using density of check-ins over a grid as the POIs in our context. We first calculate the relations and satisfaction values between all pairs of users. Then, we find the clustering of users using the proposed greedy heuristic using algorithm shown in Algorithm 1 and aforementioned evolutionary greedy algorithm.

TABLE 3. Comparison of two location based social network datasets and values of parameters used.

	Gowalla	Brightkite
Total user count	6187	3331
Number of (top) users used	263	59
Number of (top) regions used	20	
Visit duration	5-15 min	

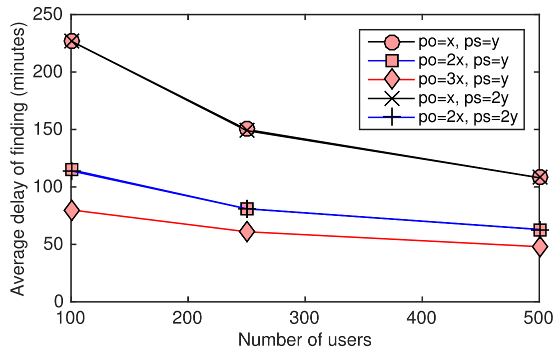

FIGURE 2. Average delay of finding lost items in synthetic data for different probabilities of p_o and p_s, with $x = 0.05$ and $y = 0.2$.

Table 3 shows the comparison of these two datasets. There are many users with smaller visit counts and they are distributed to only a few number of areas. Thus, we select only the users with more than 100 visits, yielding 263 and 59 (top) users, respectively.

B. RESULTS

We first run simulations to see the impact of the average delay of finding lost items of a user by other nodes. To this end, we let each user lose an item at a random time in randomly selected one of his/her visits at a randomly selected POI he/she visits. Using the probabilities, we then find the user who can detect this item (which could be either other users or him/herself). Fig. 2 shows the results for different p_o and p_s values. The results show the average for all nodes in the network generated by synthetic data. As the number of users increase, the average delay decreases as expected. Moreover, due to the dominance of other users, changes in p_s can only slightly affect the results. To verify these findings, we also calculated the average delay of finding using (3). The comparison of simulation and analysis results for different p_o and p_s values are shown in Fig. 3. As the plots clearly show, the results almost perfectly match. We also obtain results using real datasets, and confirmed the match between simulation and analytical results. The corresponding figures are not shown here for brevity.

Before looking at the results with clustering, first we show the percentage of total benefits each user can obtain from its top benefit providers in each dataset. Fig. 4 shows the CDF of average total benefit obtained with different number of top benefit providers for each user. For this, we sorted out all benefit providers to each user in descending order of benefits

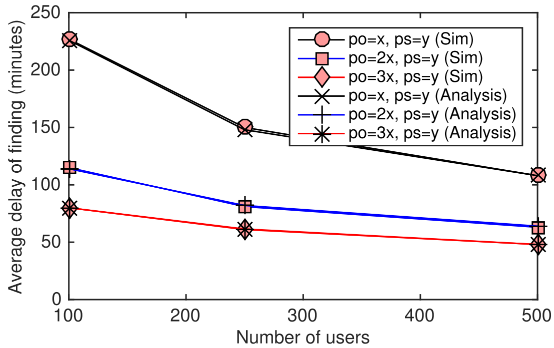

FIGURE 3. Comparison of simulation and analytical results for average delay of finding lost items in synthetic data.

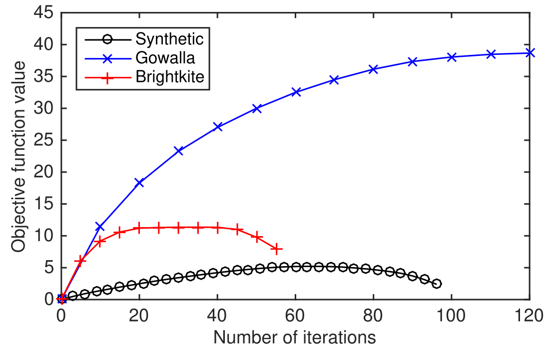

FIGURE 5. Change in objective function value as the greedy algorithm iterates in three datasets (algorithm was not stopped after maximum point intentionally to observe the curve).

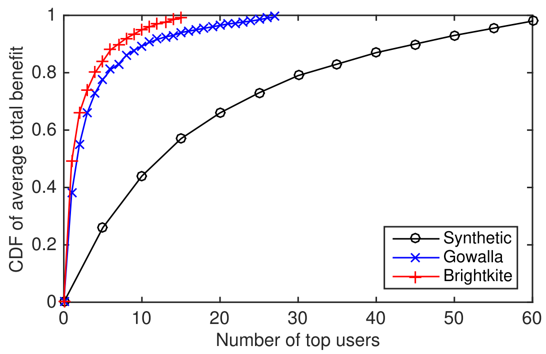

FIGURE 4. Percentage of total benefits from top users in all locations in three datasets.

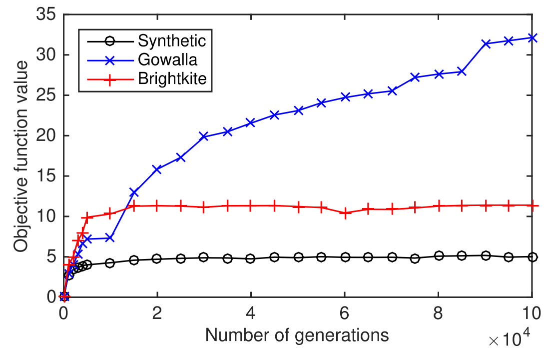

FIGURE 6. Change in objective function value as the genetic algorithm (with 100k generations) iterates in three datasets.

provided and selected the first X of them as the top X users. In real datasets, we use 20 POIs, thus average user per location is around 13 in Gowalla and 3 in Brighkite. We can see that when each user selects only its that many top users (equal to average user count per location), on average, Gowalla users can obtain 92% of all possible benefits already and Brightkite users obtain around 75% of all possible benefits. These show that users actually do not need to be part of the same network with many others as they can get sufficiently high benefit with certain number of other users (e.g., users visiting the same regions with themselves). In synthetic data design, we intentionally relaxed this high benefit providing ability of top users a bit, thus, the users can only get around 47% of all possible benefits with average group count of 10.

Next, we present results regarding the performance of proposed clustering algorithms. In Fig. 5, we show the change in the objective function during the iteration in the greedy algorithm. The greedy algorithm is designed to stop when the new group merge operation do not provide an increase in the objective function, thus it stops when it reaches a maximum value. However, to observe the decrease after this maximum point, we let the algorithm run for some time. In all three datasets, we observe that at the iteration which is around the half number of users in the dataset, the maximum value for the objective function is obtained. While this could be considered expected due to the nature of the greedy algorithm, how the grouping of users is done will matter and may not have provided such smooth curves. Using the genetic algorithm proposed, we also calculate the objective (i.e., fitness) function value through the generations. Fig. 6 shows these results for the three datasets. In synthetic and Brightkite datasets, due to the smaller number of users used compared to Gowalla, convergence of the objective value is achieved earlier than it is in Gowalla. But in all datasets, genetic algorithm can provide similar maximum objective function value as in greedy algorithm.

However, to understand the performance differences between these clustering algorithms as the user count increases in the network, we compare their running time and the maximum values they achieve for objective function. Fig. 7 shows the maximum benefits obtained with these algorithms for different number of users up to 1000. While both of these algorithms provide similar value with smaller number of users, we observe that greedy algorithm can achieve better objective function value with larger number of users. On the other hand, we compare the running time of these algorithms in Fig. 8. As the figure shows clearly, the advantage of greedy algorithm in terms of providing better objective function value at larger user counts is mitigated due to its longer running time.

Finally, using the greedy algorithm, we compare the benefits obtained and reductions in user interactions with the

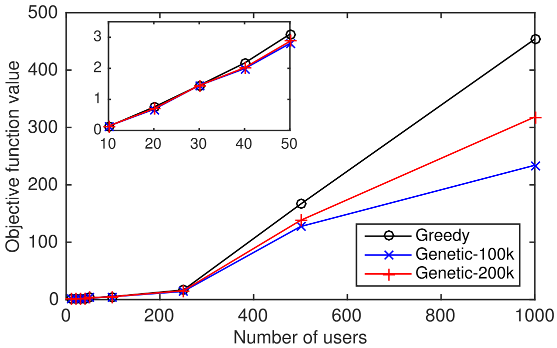

FIGURE 7. Comparison of maximum benefits obtained with different clustering algorithms.

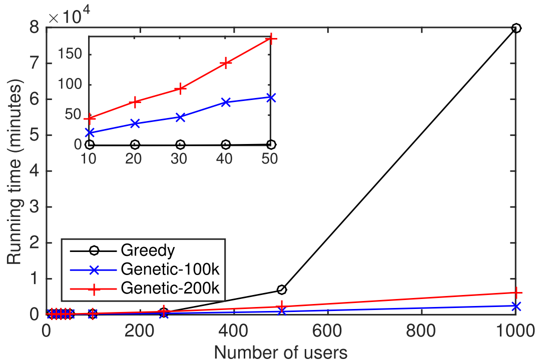

FIGURE 8. Comparison of running time (minutes) of different clustering algorithms.

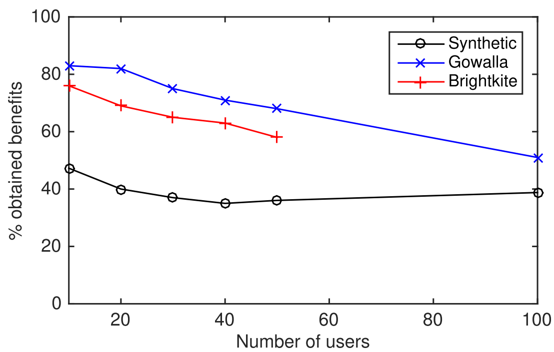

FIGURE 9. Percentage of benefits obtained with proposed greedy clustering of users to all possible benefits with single grouping (last data point for Brightkite is missing as there are not that many users).

proposed grouping model. In Fig. 9, we plot the percentage of benefits obtained with proposed grouping idea compared to max benefits that could be obtained when all users are in a single group. The results show that we can obtain around 40-80% of maximum possible benefit with only a few (top) users in all three datasets. These results are obtained by only selecting a certain random number of users in these datasets. While the grouping strategy decreases the overall benefit for the users a bit, it decreases the interactions between users a lot. In Fig. 10, we show the percentage of the reduction of these interactions in proposed grouping model compared to the single group model. As the figure shows, there is a

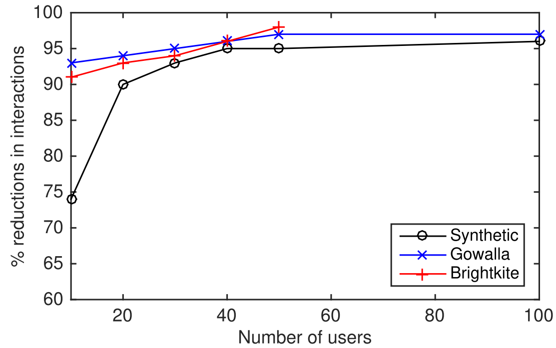

FIGURE 10. Percentage of reductions obtained with proposed greedy clustering of users compared to the interactions between all users in single grouping (last data point for Brightkite is missing as there are not that many users).

huge benefit with introduced grouping model in preserving the privacy of users.

V. CONCLUSION

In this paper, we study the problem of lost object tracking with a clustered crowd GPS service. Approaching the problem through recently trending spatial crowdsourcing context, we propose clustering of users of the Beacon tag network that can provide most of the total possible benefits to each other in terms of the localization of their lost items. To this end, we analyze the visit patterns of users at the same location and present a new metric called Social Tracking Distance (STD) that quantifies the benefit of users to each other in terms of the capability of finding each other's lost objects. Once the potential benefit of each user to every other user is calculated, we then divide all users into clusters such that the users in the same group provide high benefit to each other and the privacy of users is valued at its maximum. For clustering we have used both a greedy algorithm and a genetic algorithm. In simulations, using both a synthetic data and two real location based social network datasets, the performance of the proposed clustered crowd GPS system for active localization is evaluated. The results show that with the grouping strategy proposed, the user interactions drop drastically without sacrificing from the maximum possible benefits. We also compare the performance of the clustering algorithms used and show that genetic algorithm can provide faster calculation time compared to greedy approach when the number of nodes in the network increases. However, the greedy algorithm provides better results in terms of achieving maximum total benefit per shared location privacy.

In our future work, we will enhance the proposed metric with the integration of mobility prediction algorithms [34]–[36] and network community structure [37], [38] for a better accuracy in understanding the future benefits of users to each other. Moreover, for the indoor areas where the GPS information is not available or accurate (e.g., tunnel or underground subway station), we will enhance the proposed system with complementary solutions based on the proximity analysis between the mobile devices in the vicinity (e.g., Bluetooth aided localization [39]). To this end, we will also integrate neural network based approaches (e.g., RNN, nonlinear neural circuits [39]) to increase the performance of the proposed system.

REFERENCES

[1] Pixie. (Mar. 2017). *The Nation's Biggest Lost and Found Survey*. [Online]. Available: https://getpixie.com/blogs/news/lostfoundsurvey

[2] N. Guy. (Oct. 2017). *The Best GPS Trackers for Cats and Dogs*. [Online]. Available: https://thewirecutter.com/reviews/best-gps-pet-trackers/

[3] R. Edwards. *Best Wearable GPS Trackers for Kids Buyers Guide*. Accessed: Apr. 2018. [Online]. Available: https://www.safewise.com/resources/wearable-gps-tracking-devices-for-kids-guide

[4] K. Bishop. (2017). *Growth Business: GPS TrackingĚ The Elderly*. [Online]. Available: https://www.cnbc.com/2014/03/11/growth-business-gps-tracking-the-elderly.html

[5] C. Gomez, J. Oller, and J. Paradells, "Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology," *Sensors*, vol. 12, no. 9, pp. 11734–11753, 2012.

[6] thetileapp.com. (2016). *Tile Bluetooth Tracker Device*. [Online]. Available: https://www.thetileapp.com

[7] TraxFamily.com. (2016). *A Really Small and Light GPS Tracker for Kids and Pets*. [Online]. Available: https://traxfamily.com

[8] S. Cohen. (2015). *Locus Pocus Can Track all Your Bluetooth Devices—And Other People's, Too*, VentureBeat. [Online]. Available: http://venturebeat.com/2015/06/06/locus-pocus-can-trackall-your-bluetooth-devices-and-other-peoples-too

[9] J. Zhang, F. Tang, L. Barolli, Y. Yang, and W. Xu, "Information gain based maximum task matching in spatial crowdsourcing," in *Proc. IEEE 31st Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, Mar. 2017, pp. 886–893.

[10] L. Kazemi and C. Shahabi, "GeoCrowd: Enabling query answering with spatial crowdsourcing," in *Proc. 20th Int. Conf. Adv. Geographic Inf. Syst.*, 2012, pp. 189–198.

[11] E. Bulut and B. K. Szymanski, "Identifying the space buddies to track lost items," in *Proc. 2nd Int. Workshop Social Sens.*, 2017, pp. 69–74.

[12] A. Campbell, "The rise of people-centric sensing," in *Distributed Computing and Networking. ICDCN 2009* (Lecture Notes in Computer Science), vol. 5408, V. Garg, R. Wattenhofer, and K. Kothapalli, Eds. Berlin, Germany: Springer, 2008. [Online]. Available: https://link.springer.com/chapter/10.1007%2F978-3-540-92295-7_3

[13] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: Current state and future challenges," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 32–39, Nov. 2011.

[14] B. O. Holzbauer, B. K. Szymanski, and E. Bulut, "Socially-aware market mechanism for participatory sensing," in *Proc. 1st ACM Int. Workshop Mission-Oriented Wireless Sensor Netw.*, 2012, pp. 9–14.

[15] (2015). *Bluetooth Beyond*. [Online]. Available: http://www.telecomreview.com/index.php?option=com_k2&view=item&id=1545:bluetooth-and-beyond&Itemid=427

[16] Apple. (2016). *iBeacon for Developers*. [Online]. Available: https://developer.apple.com/ibeacon

[17] J. Hallberg, C. Nugent, R. Davies, and M. Donnelly, "Localisation of forgotten items using rfid technology," in *Proc. 9th Int. Conf. Inf. Technol. Appl. Biomed. (ITAB)*, Nov. 2009, pp. 1–4.

[18] M. Sasagawa, K. Ikematsu, and I. Siio, "Simply tag and find: Finding indoor items by using detection history of RFID tags," in *Proc. 2016 ACM Int. Joint Conf. Pervasive Ubiquitous Comput., Adjunct*, 2016, pp. 365–368.

[19] D. Schwarz, M. Schwarz, J. Stückler, and S. Behnke, "Cosero, find my keys! Object localization and retrieval using bluetooth low energy tags," in *RoboCup 2014: Robot World Cup XVIII. RoboCup* (Lecture Notes in Computer Science), vol. 8992, R. Bianchi, H. Akin, S. Ramamoorthy, and K. Sugiura, Eds. Cham, Switzerland: Springer, 2015. [Online]. Available: https://link.springer.com/chapter/10.1007%2F978-3-319-18615-3_16

[20] P. Martin, B.-J. Ho, N. Grupen, S. Muñoz, and M. B. Srivastava, "An iBeacon primer for indoor localization: Demo abstract," in *Proc. 1st ACM Conf. Embedded Syst. Energy-Efficient Buildings*, 2014, pp. 190–191.

[21] H.-E. Chen, Y.-Y. Lin, C.-H. Chen, and I.-F. Wang, "BlindNavi: A navigation app for the visually impaired smartphone user," in *Proc. 33rd Annu. ACM Conf. Extended Abstracts Human Factors Comput. Syst.*, 2015, pp. 19–24.

[22] J. Yang, Z. Wang, and X. Zhang, "An iBeacon-based indoor positioning systems for hospitals," *Int. J. Smart Home*, vol. 9, no. 7, pp. 161–168, 2015.

[23] R. Couto, J. Leal, P. M. Costa, and T. Galvão, "Exploring ticketing approaches using mobile technologies: QR codes, NFC and BLE," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, Sep. 2015, pp. 7–12.

[24] N. Allurwar, B. Nawale, and S. Patel, "Beacon for proximity target marketing," *Int. J. Eng. Comput. Sci*, vol. 5, no. 5, p. 16359–16364, 2016.

[25] J. Nickels *et al.*, "Find my stuff: Supporting physical objects search with relative positioning," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, 2013, pp. 325–334.

[26] S. Mirri, C. Prandi, P. Salomoni, and L. Monti, "Social location awareness: A prototype of altruistic IoT," in *Proc. 8th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Nov. 2016, pp. 1–5.

[27] I. Agadakos, J. Polakis, and G. Portokalidis, "Techu: Open and privacy-preserving crowdsourced GPS for the masses," in *Proc. 15th Annu. Int. Conf. Mobile Syst., Appl., Services*, 2017, pp. 475–487.

[28] E. Bulut and B. K. Szymanski, "Exploiting friendship relations for efficient routing in mobile social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 12, pp. 2254–2265, Dec. 2012.

[29] E. Bulut, S. C. Geyik, and B. K. Szymanski, "Utilizing correlated node mobility for efficient DTN routing," *Pervasive Mobile Comput.*, vol. 13, pp. 150–163, Aug. 2014.

[30] E. Falkenauer, *Genetic Algorithms and Grouping Problems*. Hoboken, NJ, USA: Wiley, 1998.

[31] A. H. Kashan, M. Jenabi, and M. H. Kashan, "A new solution approach for grouping problems based on evolution strategies," in *Proc. Int. Conf. Soft Comput. Pattern Recognit. (SOCPAR)*, Dec. 2009, pp. 88–93.

[32] R. L. Haupt, "Optimum population size and mutation rate for a simple real genetic algorithm that optimizes array factors," in *Proc. Antennas Propag. Soc. Int. Symp.*, vol. 2. Jul. 2000, pp. 1034–1037.

[33] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 1082–1090.

[34] S. C. Geyik, E. Bulut, and B. K. Szymanski, "Grammatical inference for modeling mobility patterns in networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 11, pp. 2119–2131, Nov. 2013.

[35] S. C. Geyik, E. Bulut, and B. K. Szymanski, "PCFG based synthetic mobility trace generation," in *Proc. Global Telecommun. Conf. (GLOBECOM)*, Dec. 2010, pp. 1–5.

[36] E. Bulut, S. C. Geyik, and B. K. Szymanski, "Efficient routing in delay tolerant networks with correlated node mobility," in *Proc. IEEE 7th Int. Conf. Mobile Adhoc Sensor Syst. (MASS)*, Nov. 2010, pp. 79–88.

[37] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Comput. Surv.*, vol. 45, no. 4, p. 43, 2013.

[38] E. Bulut, Z. Wang, and B. K. Szymanski, "Impact of social networks on delay tolerant routing," in *Proc. Global Telecommun. Conf. (GLOBECOM)*, Nov./Dec. 2009, pp. 1–6.

[39] S. Li, Y. Lou, and B. Liu, "Bluetooth aided mobile phone localization: A nonlinear neural circuit approach," *ACM Trans. Embedded Comput. Syst.*, vol. 13, no. 4, p. 78, 2014.

FATIH YUCEL (M'17) received the B.S. degree from Gazi University, Turkey, in 2017. He is currently pursuing the Ph.D. degree with the Computer Science Department, Virginia Commonwealth University, Richmond, VA, USA, under the supervision of Dr. E. Bulut. He joined the MoWiNG Lab, Virginia Commonwealth University, in 2017. He is involved in the development of efficient algorithms for the Internet of Things and spatial crowdsourcing.

EYUPHAN BULUT (M'08) received the Ph.D. degree from the Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY, USA, in 2011. He was a Senior Engineer with the Mobile Internet Technology Group, Cisco Systems, Richardson, TX, USA, for four and half years. He is currently an Assistant Professor with the Department of Computer Science, Virginia Commonwealth University, Richmond, VA, USA. His research interests include mobile and wireless computing, network security and privacy, mobile social networks, and crowd-sensing.