Truth Discovery in Crowdsourced Detection of Spatial Events

Robin Wentao Ouyang, Mani Srivastava, Alice Toniolo, and Timothy J. Norman

Abstract—The ubiquity of smartphones has led to the emergence of mobile crowdsourcing tasks such as the detection of spatial events when smartphone users move around in their daily lives. However, the credibility of those detected events can be negatively impacted by unreliable participants with low-quality data. Consequently, a major challenge in mobile crowdsourcing is truth discovery, i.e., to discover true events from diverse and noisy participants' reports. This problem is uniquely distinct from its online counterpart in that it involves uncertainties in both participants' *mobility* and *reliability*. Decoupling these two types of uncertainties through location tracking will raise severe privacy and energy issues, whereas simply ignoring missing reports or treating them as negative reports will significantly degrade the accuracy of truth discovery. In this paper, we propose two new unsupervised models, i.e., Truth finder for Spatial Events (TSE) and Personalized Truth finder for Spatial Events (PTSE), to tackle this problem. In TSE, we model location popularity, location visit indicators, truths of events, and three-way participant reliability in a unified framework. In PTSE, we further model personal location visit tendencies. These proposed models are capable of effectively handling various types of uncertainties and automatically discovering truths without any supervision or location tracking. Experimental results on both real-world and synthetic datasets demonstrate that our proposed models outperform existing state-of-the-art truth discovery approaches in the mobile crowdsourcing environment.

Index Terms—Mobile crowdsourcing, truth discovery, probabilistic graphical models

1 Introduction

The growing smartphone user base has enabled mobile crowdsourcing applications on a large scale [24]. Several commercial markets such as Field Agent [2], Gigwalk [3], and TaskRabbit [5] have emerged, which represent the mobile equivalent of online crowdsourcing markets such as the Amazon Mechanical Turk [1]. Crowdsourced detection of spatial events is one such application where participants detect events while moving around in their daily lives. Example events include potholes on streets [23], graffiti on walls [26], and bike racks in public places [33]. The detection of these events can enable real-time monitoring of road and traffic conditions in a city [23], enable early detection of and response to social disorder [26], and enable up-to-date documentation of "green" resources for sustainable practices [33].

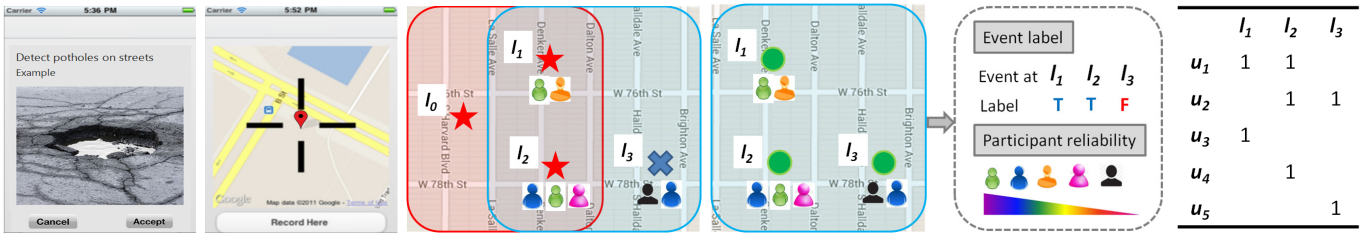
Consider the task of detecting the locations of potholes as an example, where Fig. 1(a) shows a user interface for task instruction. Since the number of possible event locations is huge and most locations normally do not have an event (e.g., no potholes), a participant uses her smartphone to make a report (tagged with time and location as shown in Fig. 1(b)) only when she detects an event. In other words, a participant either reports a detection (a positive report) or does not report at all (a missing report), but never reports a "lack of an event" (a negative report). As participants may erroneously report events due to misunderstanding, confusion, carelessness, incompetence, or even intent to deceive (Fig. 1(c)), there is a high demand for effective algorithms to handle diverse and noisy participants' reports and automatically discover the truths (Fig. 1(d)).

This truth discovery problem is uniquely distinct from its online counterpart in that it involves uncertainties not only in the participants' reliability but also in their mobility. As participants only sporadically reveal their locations when reporting events for the geotagging purpose, a missing report at a candidate event location is ambiguous (Fig. 1(e)). This is because a missing report can be due to either the mobility issue that a participant did not visit a location and could not assess the event there, or a negative event assessment when she visited that location. It is important to distinguish these two cases, as the former does not carry any information about the event truth, while the latter does.

Truth discovery in crowdsourced detection of spatial events with such ambiguous missing reports is challenging due to the following reasons. First, we cannot continuously track participants' locations (in order to disambiguate missing reports) due to privacy and energy issues [7], [19], [28], [43]. Nevertheless, if detailed location traces are known, we can then ignore missing reports corresponding to unvisited locations and treat those corresponding to visited locations as negative reports (Fig. 2(a)). After eliminating the uncertainty in mobility, we can then apply existing truth discovery methods for online crowdsourcing [29], [37], [38], [40],

- *R. W. Ouyang is with the Department of Computer Science, and M. Srivastava is with the Department of Electrical Engineering and the Department of Computer Science, University of California, Los Angeles, CA, USA.*
E-mail: {wouyang, mbs}@ucla.edu

- *A. Toniolo and T. J. Norman are with the Department of Computing Science, University of Aberdeen, Aberdeen, UK.*
E-mail: {a.toniolo, t.j.norman}@abdn.ac.uk



	l_1	l_2	l_3
u_1	1	1	
u_2		1	1
u_3	1		
u_4		1	
u_5			1

(a) (b) (c) (d) (e)

Fig. 1. (a) Example user interface for task instruction. (b) Example user interface for reporting a spatial event. (c) Illustration of the space of all true events (in the red circle, at locations l_0, l_1, l_2) and participant-reported events (in the blue circle, at locations l_1, l_2, l_3). (d) Input into the truth discovery algorithm and the expected output. (e) Participants (u) and reported events (represented by their locations l) shown in a matrix form, where a "1" and a blank space represent a positive and a missing report respectively.

	l_1	l_2	l_3
u_1	1	1	-1
u_2		1	1
u_3	1		-1
u_4	-1	1	-1
u_5		-1	1

(a) Location tracking

	l_1	l_2	l_3
u_1	1	1	
u_2		1	1
u_3	1		
u_4		1	
u_5			1

(b) Ignoring missing

	l_1	l_2	l_3
u_1	1	1	-1
u_2	-1	1	1
u_3	1	-1	-1
u_4	-1	1	-1
u_5	-1	-1	1

(c) Missing as neg.

Fig. 2. Different strategies of handling missing reports. An entry with a gray background means the corresponding report is taken into consideration for truth discovery. A "1" represents a positive report, a "-1" indicates that the missing report is treated as a negative report, and a blank space indicates that the missing report is ignored. (a) Tracking participants' locations. (b) Ignoring all missing reports. (c) Treating all missing reports as negative.

[42] to find true events. Unfortunately, continuous location tracking is impractical and thus we cannot use it to disambiguate missing reports.

Second, the attempts to reconstruct a participant's mobility paths (so as to disambiguate missing reports) would easily fail. Machine learning-based path reconstruction methods [25] require historical location traces which can only be obtained through location tracking. Moreover, such methods will easily fail when a participant deviates from her usual paths. Map matching-based path reconstruction methods [22] require road network information. Moreover, such methods will easily fail if the time interval between consecutively revealed locations is larger than several minutes.

Third, simply ignoring all the missing reports will lead to a trivial conclusion. For online crowdsourced binary image classification, Raykar et al. [32] simply ignore all the missing data. This is because online crowd workers are required to provide either a positive or a negative response for each given image, and the missing data simply imply that workers did not choose the images to work on. By applying this strategy to crowdsourced event detection, however, we will end up with only positive reports without any conflict (Fig. 2(b)). This will lead to a trivial conclusion that every reported event is true, which is obviously erroneous.

Fourth, simply treating all the missing reports as negative reports will significantly degrade the accuracy of truth discovery. In tackling conflicting Web information for data integration, Zhao et al. [42] treat missing reports as negative reports if a source did not make claims on some of the facts (e.g., did not claim that Emma Watson is a cast) but on others (e.g., claimed that Daniel Radcliffe is a cast) about an entity (e.g., the movie Harry Potter). By applying this strategy to crowdsourced event detection, we can regard the spatial area of interest as an entity and events inside it as multiple facts, where each can be either true or false. Each missing report will then become a negative report and will imply a lack of an event (Fig. 2(c)). If none of the events receives positive reports from more than half of the participants due to mobility issues, we will then conclude that all the events are false (by majority voting), which is again erroneous.

In this paper, we propose two new models, namely, Truth finder for Spatial Events (TSE) [27] and Personalized Truth finder for Spatial Events (PTSE), to tackle this challenging truth discovery problem. We observe that a participant's likelihood of reporting an event depends on three factors: 1) whether the participant has visited the event location, 2) whether the event at that location is true or false, and 3) how reliable the participant is. Based on these observations, in TSE, we model that each event location has certain popularity, which influences the possibility of a randomly selected participant to visit that location. This is motivated by the fact that some locations naturally attract more people (e.g., shopping malls) while others attract fewer (e.g., country roads). In PTSE, we model personal location visit tendencies instead of location popularity to address the mobility issues. In both of these two models, we treat the truths of events as latent variables and model three-way participant reliability, including: 1) true positive rate while present at a location, 2) false positive rate while present at a location, and 3) reporting rate while absent from a location. By doing so, positive and missing reports become random variables generated by conditioning on all the aforementioned factors. These two models thus directly address the mobility issues, can efficiently

handle missing reports, and can automatically infer the truths of events and different aspects of participant reliability. Moreover, they are unsupervised and avoid location tracking.

In contrast, existing truth discovery methods such as [11], [18], [29], [34], [37], [38], [40], [42] are mainly designed to tackle only the uncertainty in participants' reliability. None of them models location popularity, personal location visit tendencies, and three-way participant reliability. As a result, they are not effective for truth discovery in crowdsourced event detection, where both participants' mobility and reliability are uncertain.

In summary, our contributions are as follows:

1) We address the truth discovery problem in crowdsourced detection of spatial events, which is seldom studied in existing literature.
2) We propose two unsupervised probabilistic models to tackle this problem. In TSE, we model location popularity, location visit indicators, truths of events, and three-way participant reliability in an integrated framework. In PTSE, we further model personal location visit tendencies. These two models do not require location tracking.
3) We develop both batch and online model inference algorithms for TSE and PTSE.
4) We conduct extensive experiments to evaluate the performance of our proposed models and other state-of-the-art truth discovery approaches.

The remainder of this paper is organized as follows. We review related work in Section 2 and formalize the truth discovery problem in Section 3. We present the design of TSE and the model inference algorithms in Sections 4 and 5 respectively. We then present the design of PTSE and the corresponding model inference algorithms in Section 6. Experimental results are presented in Section 7. Possible future work is discussed in Section 8. Finally, we conclude the paper in Section 9.

2 Related Work

A number of unsupervised approaches have been proposed for discovering the truth from conflicting information sources. In the domain of truth discovery from conflicting Web information, Yin et al. [40] proposed Truth Finder, which is a transitive voting algorithm with rules specifying how votes iteratively flow from sources to claims and then back to sources. It has been shown to be superior than Majority Voting and the Hubs and Authorities algorithm [18] which was initially designed to find popular web pages. Pasternack and Roth [29] proposed Investment and PooledInvestment algorithms, where sources invest their credibility in the claims they make, and claim belief is then non-linearly grown and apportioned back to the sources. Zhao et al. [42] proposed a principled probabilistic approach which can automatically infer true claims and two-sided source quality. Yin et al. [41] proposed semi-supervised truth discovery.

In the domain of aggregating conflicting responses in crowdsourcing tasks, Dawid and Skene [11] modeled the generative process of the responses by introducing worker ability parameters. Whitehill et al. [38] further included the difficulty of the task in the model. Welinder et al. [37] proposed a model consisting of worker compatibility for each task. Wang et al. [34], [35] proposed an approach that models both the truth of tasks and the reliability of workers for social sensing.

Nevertheless, these methods are mainly designed to tackle only the uncertainty in participants' reliability. They are thus not effective for truth discovery in crowdsourced event detection where both participants' mobility and reliability are uncertain. Moreover, none of them models location popularity, personal location visit tendencies, location visit indicators, and three-way participant reliability. Alternative solutions, such as first tracking participants' locations and then applying existing truth discovery methods, will raise severe privacy and energy issues. In contrast, our proposed models integrate mobility, reliability, and latent truths in a unified framework, and they can jointly infer these aspects without location tracking.

3 Problem Statement

We formally define the truth discovery problem in this section. For ease of illustration, we list the notations used in this paper in Table 1.

Consider a scenario where a group of participants joins a task to detect spatial events. A participant uses her smartphone to make a report r upon detection. Each report $r = (u, l, t, \eta)$ contains the participant ID u, the location l (e.g., by GPS), the time t, and the type η of the report. Generally, truth discovery is performed on the same type of reports (e.g., those for pothole detection).

The set of related reports within a time window $\mathcal{T}$ and a spatial region $\mathcal{S}$ of interest is given by

$$\mathcal{R} = \{r | r.t \in \mathcal{T}, r.l \in \mathcal{S}\}.$$

The proper sizes of $\mathcal{T}$ and $\mathcal{S}$ are application-dependent and can be specified via domain knowledge or through data-driven spatio-temporal clustering [17]. From these reports, we can extract the set of all participants $\mathcal{U}$ and the set of all reported event locations $\mathcal{L}$ as

$$\mathcal{U} = \{u | u = r.u, r \in \mathcal{R}\},\ \mathcal{L} = \{l | l = r.l, r \in \mathcal{R}\}.$$

We use u_i and l_j to denote the ith participant and the jth event location respectively. We assume that all events last for the duration of the time window and thus can be distinguished by their locations. We denote $M = |\mathcal{U}|$ and $N = |\mathcal{L}|$.

We construct a report matrix $\mathbf{X} = \{x_{i,j}\}$ from $\mathcal{U}$ and $\mathcal{L}$ as: $x_{i,j} = 1$ if $\exists r | r.u = u_i, r.l = l_j$, which indicates that participant u_i made a report claiming that an event was detected at location l_j; $x_{i,j} = 0$ otherwise. We term $x_{i,j} = 1$ as a *positive* report and $x_{i,j} = 0$ as a *missing* report. A missing report is ambiguous since it can be

TABLE 1
Notations.

Notation	Meaning
u_i, l_j	ith participant and jth event location
M, N	number of participants and event locations
z_j; $\mathbf{z}$; $\mathbf{z}^{-j}$	label of the event at location l_j ($z_j \in \{0,1\}$); all the event labels ($\mathbf{z} = \{z_j\}$); $\mathbf{z}$ except z_j
$h_{i,j}$; $\mathbf{H}$; $\mathbf{H}^{-i,j}$	indicator of whether participant u_i visited location l_j ($h_{i,j} \in \{0,1\}$); $\mathbf{H} = \{h_{i,j}\}$; $\mathbf{H}$ except $h_{i,j}$
$x_{i,j}$; $\mathbf{x}_j$; $\mathbf{X}$	report made by participant u_i on the event at l_j ($x_{i,j} \in \{0,1\}$); all the reports for the event at l_j; $\mathbf{X} = \{x_{i,j}\}$
n_1^{-j}; n_0^{-j}	number of $z = 1$ and $z = 0$ in $\mathbf{z}^{-j}$
$n_{j,1}^{-i}$; $n_{j,0}^{-i}$	number of $h = 1$ and $h = 0$ in the jth column of $\mathbf{H}$ except the ith element
$n_{i,k,q,v}^{-j}$	number of tuples ($h = k, z = q, x = v$) associated with participant u_i except that for the jth event
s	probability that an event is true
g_j; $\mathbf{g}$	probability that l_j is visited by any participant; $\mathbf{g} = \{g_j\}$
$f_{i,j}$	u_i's tendency to visit l_j
a_i; $\mathbf{a}$	u_i's true positive rate while present; $\mathbf{a} = \{a_i\}$
b_i; $\mathbf{b}$	u_i's false positive rate while present; $\mathbf{b} = \{b_i\}$
c_i; $\mathbf{c}$	u_i's reporting rate while absent; $\mathbf{c} = \{c_i\}$
$\lambda_{v,1}; \lambda_{v,0}$	hyperparameters of the (prior) Beta distribution for variable v, $v \in \{s, g_j, a_i, b_i, c_i\}$

due to either the mobility issue that u_i did not visit l_j and could not assess the event there, or a negative event assessment made by u_i when she visited l_j. The former case does not relate to the event truth, while the latter does. However, participants' detailed mobility traces are not available due to privacy and energy issues.

Our problem of truth discovery in mobile crowdsourced detection of spatial events is to infer the true labels of events with locations in $\mathcal{L}$, based on the report matrix $\mathbf{X}$ with only positive and missing reports. We visually illustrated this problem in Fig. 1(d).

4 Truth Finder for Spatial Events

In this section, we present the design of Truth finder for Spatial Events (TSE), which is a probabilistic model for truth discovery in crowdsourced event detection. We first present the model components, and then illustrate some properties of the proposed model.

4.1 Model Design

We consider the process of how a crowdsourced report is generated. In order to make a report, a participant first needs to be physically present at a location, observes whether there is any target event, and then decides to make a report or not based on her judgment.

This process motivates us to model the following aspects: 1) location popularity, 2) a participant's location visit indicators, 3) event labels, 4) participant reliability, and 5) crowdsourced reports.

Fig. 3 shows the graphical structure of TSE, where each node represents a random variable. Dark shaded nodes indicate observed variables, and light nodes represent latent variables and model parameters. Hyperparameters that correspond to prior distributions are omitted for simplicity. We summarize the generative process of TSE in Algorithm 1 and detail its components below.

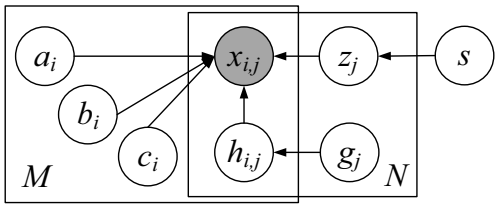


Fig. 3. Graphical model (TSE).

Algorithm 1 Generative process (TSE)

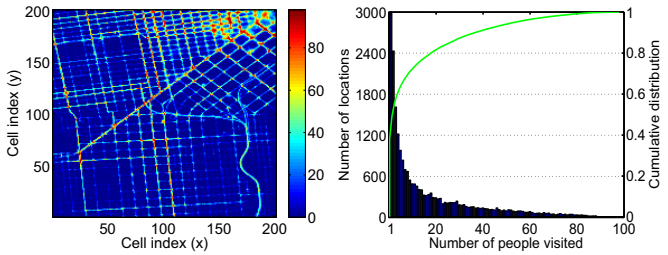
1. For each event at location l_j
   1.1 Draw the location's popularity $g_j \sim \mathsf{Beta}(\lambda_{g_{j,1}}, \lambda_{g_{j,0}})$
   1.2 Draw the event's prior truth probability $s \sim \mathsf{Beta}(\lambda_{s,1}, \lambda_{s,0})$
   1.3 Draw the event's true label $z_j \sim \mathsf{Bernoulli}(s)$
2. For each participant u_i
   2.1 Draw her true positive rate while present $a_i \sim \mathsf{Beta}(\lambda_{a_{i,1}}, \lambda_{a_{i,0}})$
   2.2 Draw her false positive rate while present $b_i \sim \mathsf{Beta}(\lambda_{b_{i,1}}, \lambda_{b_{i,0}})$
   2.3 Draw her reporting rate while absent $c_i \sim \mathsf{Beta}(\lambda_{c_{i,1}}, \lambda_{c_{i,0}})$
3. For each participant u_i and event at l_j
   3.1 Draw a location visit indicator $h_{i,j} \sim \mathsf{Bernoulli}(g_j)$
   3.2 Draw a report $x_{i,j}$
      3.2.1 If $h_{i,j} = 1$ and $z_j = 1$, draw $x_{i,j} \sim \mathsf{Bernoulli}(a_i)$
      3.2.2 If $h_{i,j} = 1$ and $z_j = 0$, draw $x_{i,j} \sim \mathsf{Bernoulli}(b_i)$
      3.2.3 If $h_{i,j} = 0$, draw $x_{i,j} \sim \mathsf{Bernoulli}(c_i)$

4.1.1 Location Popularity

In the physical world, different locations usually have different attractiveness. For example, shopping malls are generally visited by a large number of people, but a residence area will only be visited by a few people who live there. This motivates us to model location popularity, which is the probability that a randomly chosen participant will visit a location.

Our intuition is supported by the findings from a public mobility dataset[1] which contains time-stamped GPS location traces for 536 taxicabs over a span of roughly one month in the city of San Francisco [30]. Fig. 4(a) plots the location heat map in the northeast part of the city generated from the dataset, where each point shows the number of distinct people (we randomly choose 100) who have visited that location. It is clear that some locations are visited by many people while some by much fewer. We find that locations with high popularity mostly correspond to crossroads, popular highways, and gas stations. Fig. 4(b) plots the distribution of the number of people who visited a specific location. It can be observed that around 80% of locations are visited by at most 20% of all the people. This suggests that

1. http://crawdad.org/epfl/mobility/.



(a) (b)

Fig. 4. (a) Location heat map in the northeast part of San Francisco (latitude: 37.75 to 37.79; longitude: -122.44 to -122.40; each point represents an approximately $20m$ by $20m$ grid cell). Each point shows the number of distinct people (totally 100) visited that location. (b) Distribution of the number of people visited a specific location.

mobility issues could result in a large proportion of missing reports in crowdsourced event detection.

Formally, we model that each event location l_j has a location popularity g_j, representing the probability that a randomly chosen participant will visit it. g_j is generated from a Beta distribution as $g_j \sim \text{Beta}(\lambda_{g_{j,1}}, \lambda_{g_{j,0}}) = \frac{1}{B(\lambda_{g_{j,1}}, \lambda_{g_{j,0}})} g_j^{\lambda_{g_{j,1}}-1}(1-g_j)^{\lambda_{g_{j,0}}-1}$, where the hyperparameters $(\lambda_{g_{j,1}}, \lambda_{g_{j,0}})$ represent the prior counts of the number of distinct participants who visited and did not visit location l_j respectively.

4.1.2 Participant's Location Visit Indicator

We use $h_{i,j} = 1$ and $h_{i,j} = 0$ to denote that participant u_i visited and did not visit location l_j respectively. We then model that a participant's location visit indicator $h_{i,j}$ is generated from a Bernoulli distribution parameterized by the location popularity g_j, i.e., $h_{i,j} \sim \text{Bernoulli}(g_j) = g_j^{h_{i,j}}(1-g_j)^{1-h_{i,j}}$. In this way, a participant has a higher chance to visit a more popular location.

4.1.3 Event Label

Since each reported event can be either true or false, we view them as binary random variables. We model that each event has a prior probability s of being true, and s is generated from a Beta distribution. We use $z_j = 1$ and $z_j = 0$ to denote that the ground truth label of the event at l_j is true and false respectively. The binary label z_j is then modeled as being generated from $z_j \sim \text{Bernoulli}(s)$.

4.1.4 Participant Reliability

In crowdsourced detection of spatial events, a participant's reliability depends on two factors: $h_{i,j}$ (a participant's location visit indicator) and z_j (an event's true label), where the former factor does not exist in an online setting. It is desirable to model different aspects of participant reliability due to the following reasons.

First, it is likely that different participants have different attitudes towards reporting true and false events. A reliable participant will mostly report detections for true events but will seldom make reports for false events, which results in a high true positive rate and a high true negative rate. On the other hand, a conservative participant is likely to make a report only when she is very confident that an event is true or when she is willing to report, which results in a low true positive rate but a high true negative rate. In other words, it is not appropriate to use a single correct rate (e.g., as that in [29], [38], [40]) to model participant reliability in crowdsourced event detection. Moreover, the true positive rate and the true negative rate in crowdsourced event detection make sense only with respect to those visited event locations (i.e., $h_{i,j} = 1$).

Second, as has been discussed, if a participant did not visit a location, she cannot make a report there. As a consequence, such a missing report is due to the participant's mobility issue rather than her bias or carelessness when judging an event's label. Therefore, it is desirable to use a parameter to characterize the participant's reporting rate without visiting a location.

Formally, we model three-way participant reliability as follows.

1) True Positive Rate while present (TPR): We use a_i to denote the probability that participant u_i reports that the event at l_j is true when she is present at l_j and the event there is indeed true; i.e., $a_i = p(x_{i,j} = 1 | h_{i,j} = 1, z_j = 1)$. The TPR a_i is modeled to be generated from a Beta distribution with hyperparameters $(\lambda_{a_{i,1}}, \lambda_{a_{i,0}})$, representing the prior counts of positive and missing reports when u_i is present at an event location and the event there is true. It is clear that TPR makes sense only when a participant really visited an event location ($h_{i,j} = 1$). Without such a consideration, missing reports resulted from mobility issues can easily bias a participant's TPR.

2) False Positive Rate while present (FPR): We use b_i to denote the probability that participant u_i reports that event at l_j is true when she is present at l_j and the event there is actually false; i.e., $b_i = p(x_{i,j} = 1 | h_{i,j} = 1, z_j = 0)$. The choice to model the false positive rate rather than the true negative rate is for notational convenience. The FPR b_i is modeled to be generated from a Beta distribution with hyperparameters $(\lambda_{b_{i,1}}, \lambda_{b_{i,0}})$, representing the prior counts of positive and missing reports when u_i is present at an event location and the event there is false. Similarly, FPR makes sense only when a participant really visited an event location. Otherwise, missing reports attributed to mobility issues can also easily bias a participant's FPR.

3) Reporting Rate while Absent (RRA): We use c_i to denote the probability that participant u_i reports that event at l_j is true when she is not physically at l_j; i.e., $c_i = p(x_{i,j} = 1 | h_{i,j} = 0)$. The RRA c_i is modeled to be generated from a Beta distribution with hyperparameters $(\lambda_{c_{i,1}}, \lambda_{c_{i,0}})$, representing the prior counts of positive and missing reports when u_i is absent from an event location. Since a participant's location is recorded when a report is made, the probability c_i that u_i made a report with a geotag l_j but was not physically at l_j (within the localization accuracy bound) should be close to zero (we discuss the issue of location obfuscation in Section 8). We specify a large $\lambda_{c_{i,0}}$ and a small

$\lambda_{c_{i,1}}$ to make c_i conform to such a real-world physical constraint. Moreover, as a participant cannot evaluate an event's label when she is not at the event location, we model c_i to be independent of the event label z_j. The introduction of the probability c_i also ensures that the model inference procedure only allows the presence of the case $(x_{i,j} = 0|h_{i,j} = 0)$ (missing reports due to mobility issues) but not $(x_{i,j} = 1|h_{i,j} = 0)$ (positive reports without visiting locations).

As can be seen, the modeling of TPR a_i, FPR b_i, and RRA c_i can fully specify the confusion matrix for crowdsourced reports under different combinations of $h_{i,j}$ and z_j.

4.1.5 Crowdsourced Report

We now consider how reports are generated. Take a missing report from a participant as an example. It can be resulted from several cases: i) the participant visited the event location which had a target event, but she did not make a report, ii) the participant visited the event location which did not have any target event, and she did not make a report, and iii) the participant did not visit the event location and thus could not make a report there. Therefore, we model each report $x_{i,j}$ as a Boolean random variable generated from a Bernoulli distribution that depends on the participant's location visit indicator $h_{i,j}$ and the event label z_j. Moreover, this distribution is parameterized by different aspects of participant reliability a_i, b_i, and c_i.

Formally, we model

$$x_{i,j} \sim \begin{cases} \text{Bernoulli}(a_i) \text{ if } h_{i,j} = 1, z_j = 1 \\ \text{Bernoulli}(b_i) \text{ if } h_{i,j} = 1, z_j = 0 \\ \text{Bernoulli}(c_i) \text{ if } h_{i,j} = 0. \end{cases}$$

4.2 Model Analysis

We discuss several properties of TSE below.

1) Missing reports are well explained. According to the model structure in Fig. 3, the probability of a missing report from participant u_i on event at l_j is given by

$$\begin{aligned} & p(x_{i,j} = 0) \\ &= \sum_{k=0}^{1}\sum_{q=0}^{1} p(h_{i,j} = k)p(z_j = q)p(x_{i,j} = 0|h_{i,j} = k, z_j = q) \\ &= (1 - g_j)(1 - c_i) + g_j[(1 - s)(1 - b_i) + s(1 - a_i)]. \end{aligned}$$

This expression clearly captures the composite effect of various factors that can result in a missing report. When the location popularity $g_j \to 1$, we have $p(x_{i,j} = 0) \to (1-s)(1-b_i)+s(1-a_i)$. It indicates that for a very popular location, the probability of observing a missing report is mainly due to the event's truth and a participant's TPR a_i and FPR b_i. On the other hand, when the location popularity $g_j \to 0$, we have $p(x_{i,j} = 0) \to 1 - c_i$. It indicates that for a very unpopular location, the probability of observing a missing report is then mainly due to a participant's limited mobility and RRA c_i. In more general scenarios, these two possibilities are combined through g_j. Positive reports can be explained similarly.

To demonstrate the importance of modeling the location popularity g_j, a participant's location visit indicator $h_{i,j}$, and her RRA c_i, we now examine the probability of observing a missing report without modeling these variables (equivalent to setting $g_j = 1$ and $h_{i,j} = 1$). We then have

$$\begin{aligned} & p(x_{i,j} = 0) = \sum_{q=0}^{1} p(z_j = q)p(x_{i,j} = 0|z_j = q) \\ &= (1 - s)(1 - b_i) + s(1 - a_i). \end{aligned}$$

This expression shows that any missing report is attributed to the event's truth and a participant's TPR and FPR. However, as has been discussed, a missing report can also be caused by the mobility issue. Consequently, such a model cannot tackle the uncertainty in participants' mobility and a large portion of missing reports caused by low location popularity can easily bias the event's truth and a participant's TPR and FPR.

2) Location tracking is avoided. TSE does not require continuous location tracking for each participant to disambiguate the cause of missing reports, and thus it alleviates the privacy and energy issues. Instead, we model location popularity, which is the probability that a randomly chosen participant will visit a location. On the one hand, location popularity can be directly estimated through domain knowledge. For example, we can specify proper prior parameters $(\lambda_{g_{j,1}}, \lambda_{g_{j,0}})$ to impose a high location popularity for shopping malls, crossroads, gas stations, and popular highways. On the other hand, since location popularity is a collective rather than a personal measure, its prior counts can be estimated once from any resource where location tracking is not a concern (e.g., studies on human mobility [15], [16]). In contrast, location tracking needs to be performed repeatedly for each participant in each task.

3) Different aspects of participant reliability are handled. We model three-way participant reliability which covers all the cases conditioned on different combinations of $h_{i,j}$ and z_j. As a consequence, TSE jointly considers the effect of mobility and the effect of character on participants' reports. It can also efficiently handle different aspects of participants' attitudes towards reporting true and false events upon observation.

4) Prior belief can be easily incorporated. We take a Bayesian approach and specify prior distributions for model parameters. This allows us to easily incorporate domain knowledge for truth discovery. In the absence of such knowledge, we can simply use uniform priors.

5 Inference Algorithm

In this section, we discuss how to perform model inference to estimate 1) latent variables: event labels and location visit indicators, and 2) model parameters: prior truth probability, participant reliability, and location popularity based on TSE, given the report matrix $\mathbf{X}$.

We summarize the inference procedure in Algorithm 2. Moreover, we propose a fast online inference algorithm. As a result, the model parameters can be infrequently re-estimated (in an unsupervised manner) based on recent crowdsourced reports, and then be used for online truth discovery as new reports arrive.

5.1 Estimating Event Labels and Location Visit Indicators

Given the data matrix $\mathbf{X}$ and the TSE model, we need to find the optimal configuration of the random variables that maximize the posterior probability, i.e., using the maximum a posterior (MAP) estimator [8]. For example, to infer the event labels, we need to solve

$$\begin{aligned}\mathbf{z}^* &= \arg\max_{\mathbf{z}} p(\mathbf{z}|\mathbf{X}) \propto \sum_{\mathbf{H}} \int p(\mathbf{X},\mathbf{H},\mathbf{z},\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{g},s) d\mathbf{a} d\mathbf{b} d\mathbf{c} d\mathbf{g} ds \\ &= \sum_{\mathbf{H}} \int p(\mathbf{X}|\mathbf{H},\mathbf{z},\mathbf{a},\mathbf{b},\mathbf{c}) p(\mathbf{H}|\mathbf{g}) p(\mathbf{g}) p(\mathbf{z}|s) p(s) \\ &\quad \times p(\mathbf{a}) p(\mathbf{b}) p(\mathbf{c}) d\mathbf{a} d\mathbf{b} d\mathbf{c} d\mathbf{g} ds.\end{aligned}$$

Given the complex form of the joint distribution (refer to Fig. 3), direct optimization is difficult to perform, especially when z and h are discrete. We thus resort to the collapsed Gibbs sampling [20] for model inference. In our implementation, we integrate out all the model parameters and only sample the latent variables z_j and $h_{i,j}$ (more detailed derivation is provided in the Appendix).

1) Sampling z_j. We first iteratively sample the label for each event according to the following update rules. The meaning of the notations is listed in Table 1.

$$\begin{aligned}&p(z_j = q|\mathbf{z}^{-j},\mathbf{H},\mathbf{X}) \\ &\propto (n_q^{-j} + \lambda_{s,q}) \prod_{i:h_{i,j}=1} \phi_{i,j}(h_{i,j}=1, z_j = q, x_{i,j}),\end{aligned} \quad (1)$$

where $q \in \{0,1\}$ and $\phi_{i,j}(h_{i,j}, z_j, x_{i,j})$ is defined as

$$\phi_{i,j} = \begin{cases} \frac{n^{-j}_{i,1,1,1}+\lambda_{a_i,1}}{\sum_d (n^{-j}_{i,1,1,d}+\lambda_{a_i,d})} & \text{if } h_{i,j}=1, z_j=1, x_{i,j}=1 \\ \frac{n^{-j}_{i,1,1,0}+\lambda_{a_i,0}}{\sum_d (n^{-j}_{i,1,1,d}+\lambda_{a_i,d})} & \text{if } h_{i,j}=1, z_j=1, x_{i,j}=0 \\ \frac{n^{-j}_{i,1,0,1}+\lambda_{b_i,1}}{\sum_d (n^{-j}_{i,1,0,d}+\lambda_{b_i,d})} & \text{if } h_{i,j}=1, z_j=0, x_{i,j}=1 \\ \frac{n^{-j}_{i,1,0,0}+\lambda_{b_i,0}}{\sum_d (n^{-j}_{i,1,0,d}+\lambda_{b_i,d})} & \text{if } h_{i,j}=1, z_j=0, x_{i,j}=0. \end{cases}$$

The $\phi_{i,j}$ function appears naturally as a result of deriving the sampling rules. All the counts involved in the sampling rules can be updated incrementally [42].

The first part in (1) carries information from other event labels and the second part carries information from the reports made by participants on other events (except that at l_j). Note that, (1) only relates to participants whose $h_{i,j} = 1$. This is because only when a participant visited an event location and had an opportunity to assess the event label, that report (either positive or missing) carried information about the true event label. Otherwise, that report should not be taken into consideration for truth discovery.

Algorithm 2 Model Inference

Input: Reports $x_{i,j}$
Output: Latent variables $z_j, h_{i,j}$; model parameters s, a_i, b_i, g_j

{*Initialization*}
For all z_j, sample $z_j \sim$ Bernoulli(0.5)
For all $x_{i,j} = 0$, sample $h_{i,j} \sim$ Bernoulli(0.5)
For all $x_{i,j} = 1$, set $h_{i,j} = 1$
{*Sampling for K rounds*}
for $t = 1 : K$ **do**
  {*Update every z_j*}
  Calculate $p_j^q \triangleq p(z_j = q|\mathbf{z}^{-j},\mathbf{H},\mathbf{X})$ according to (1)
  Sample $z_j^{(t)} \sim$ Bernoulli$(p_j^1/(p_j^1+p_j^0))$ & update counts
  {*Update every $h_{i,j}$ for $x_{i,j} = 0$*}
  Calculate $p_{i,j}^k \triangleq p(h_{i,j} = k|\mathbf{z},\mathbf{H}^{-ij},\mathbf{X})$ according to (2)
  Sample $h_{i,j}^{(t)} \sim$ Bernoulli$(p_{i,j}^1/(p_{i,j}^1+p_{i,j}^0))$ & update counts
end for
{*Estimate event labels and location visit indicators*}
Estimate $\hat{p}(z_j = 1)$ based on every K_2 samples in the remaining $K - K_1$ rounds; $\hat{z}_j = 1$ if $\hat{p}(z_j = 1) \geq 0.5$ and $\hat{z}_j = 0$ otherwise; similarly for $\hat{h}_{i,j}$
{*Estimate model parameters*}
Estimate $\hat{s}$, $\hat{a}_i$, $\hat{b}_i$ and $\hat{g}_j$ according to (3) – (6)
return $\hat{z}_j, \hat{h}_{i,j}, \hat{s}, \hat{a}_i, \hat{b}_i, \hat{g}_j$

2) Sampling $h_{i,j}$. After sampling all z_j, we then iteratively sample each participant's location visit indicators $h_{i,j}$ according to the following update rules. Note that, we only need to sample $h_{i,j}$ when $x_{i,j} = 0$, i.e., for those missing reports. Since the location is recorded when $x_{i,j} = 1$, we can directly infer $h_{i,j} = 1$ if $x_{i,j} = 1$.

$$\begin{aligned}&p(h_{i,j} = 1|\mathbf{z},\mathbf{H}^{-i,j},\mathbf{X}) \propto (n_{j,1}^{-i} + \lambda_{g_j,1}) \phi_{i,j}(1, z_j, 0) \\ &p(h_{i,j} = 0|\mathbf{z},\mathbf{H}^{-i,j},\mathbf{X}) \propto (n_{j,0}^{-i} + \lambda_{g_j,0}) \frac{n^{-j}_{i,0,-,0} + \lambda_{c_i,0}}{\sum_d (n^{-j}_{i,0,-,d} + \lambda_{c_i,d})} \\ &\qquad \approx n_{j,0}^{-i} + \lambda_{g_j,0}.\end{aligned} \quad (2)$$

The first part in (2) carries information from other participants' (except u_i's) location visit indicators at l_j and the second part carries information from the reports made by u_i on other events (except that at l_j). As c_i should be close to 0, we assign $\lambda_{c_i,0} \gg \lambda_{c_i,1}$, which results in $\frac{n^{-j}_{i,0,-,0}+\lambda_{c_i,0}}{\sum_d (n^{-j}_{i,0,-,d}+\lambda_{c_i,d})} \approx 1$.

The sampling procedure is performed for K rounds. To obtain $\hat{p}(z_j = 1)$ and $\hat{p}(h_{i,j} = 1)$, we discard the first K_1 samples in the burn-in period, and then for every K_2 samples in the remainder we calculate their average (thinning), which is to prevent correlation in the samples. Finally, if $\hat{p}(z_j = 1) \geq 0.5$, we output $\hat{z}_j = 1$; otherwise, we have $\hat{z}_j = 0$. The estimation of $h_{i,j}$ is similar.

5.2 Estimating Prior Truth Probability

After we have obtained the estimates of event labels z_j and location visit indicators $h_{i,j}$, we can estimate the prior truth probability using the MAP estimator by treating these inferred values as observed data. This results in closed-form estimates as follows.

$$\hat{s} = \frac{\mathbb{E}(n_1) + \lambda_{s,1}}{\sum_d [\mathbb{E}(n_d) + \lambda_{s,d}]}, \quad (3)$$

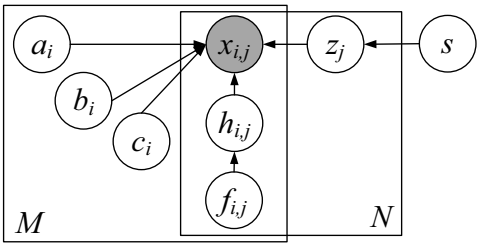


Fig. 5. Graphical model (PTSE).

where $\mathbb{E}(n_1) = \sum_j \hat{p}(z_j = 1)$ is the expected number of true events.

5.3 Estimating Participant Reliability

Similarly, we estimate the participant reliability as

$$\hat{a}_i = \frac{\mathbb{E}(n_{i,1,1,1}) + \lambda_{a_i,1}}{\sum_d [\mathbb{E}(n_{i,1,1,d}) + \lambda_{a_i,d}]}, \hat{b}_i = \frac{\mathbb{E}(n_{i,1,0,1}) + \lambda_{b_i,1}}{\sum_d [\mathbb{E}(n_{i,1,0,d}) + \lambda_{b_i,d}]}, \tag{4}$$

where $\mathbb{E}(n_{i,k,q,v})$ is the expected count of tuples $(h = k, z = q, x = v)$ related to u_i. This count depends on the probability of the location visit indicators, the probability of the event labels, and the actual reports. Formally,

$$\mathbb{E}(n_{i,k,q,v}) = \sum_{x_{i,j}=v} \hat{p}(h_{i,j} = k)\hat{p}(z_j = q). \tag{5}$$

We have $\hat{c}_i = 0$ according to the real-world physical constraint discussed in Section 4.1.4.

5.4 Estimating Location Popularity

We estimate the location popularity as

$$\hat{g}_j = \frac{\mathbb{E}(n_{j,1}) + \lambda_{g_j,1}}{\sum_d [\mathbb{E}(n_{j,d}) + \lambda_{g_j,d}]}, \tag{6}$$

where $\mathbb{E}(n_{j,1}) = \sum_i \hat{p}(h_{i,j} = 1)$ is the expected number of participants that visited event location l_j.

5.5 Online Truth Discovery

Assume we have inferred $\hat{s}$, $\hat{a}_i$, $\hat{b}_i$, $\hat{c}_i$, and $\hat{g}_j$. We can then use these estimates to perform *fast* online truth discovery, i.e., inferring the event label for a new event z_o at l_o given new crowdsourced reports $\mathbf{x}_o$. We use the MAP estimator and compute $p(z_o = 1|\mathbf{x}_o)$ as

$$p(z_o = 1|\mathbf{x}_o) \propto p(z_o = 1) \prod_i p(x_{i,o}|z_o = 1)$$
$$= p(z_o = 1) \prod_i \sum_k p(h_{i,o} = k) p(x_{i,o}|h_{i,o} = k, z_o = 1)$$
$$= \hat{s} \prod_i \left[\hat{g}_o \hat{a}_i^{x_{i,o}} (1 - \hat{a}_i)^{1-x_{i,o}} + (1 - \hat{g}_o)\hat{c}_i^{x_{i,o}} (1 - \hat{c}_i)^{1-x_{i,o}}\right].$$

Similarly, we compute $p(z_o = 0|\mathbf{x}_o)$ as

$$p(z_o = 0|\mathbf{x}_o)$$
$$\propto (1-\hat{s}) \prod_i \left[\hat{g}_o \hat{b}_i^{x_{i,o}} (1-\hat{b}_i)^{1-x_{i,o}} + (1-\hat{g}_o)\hat{c}_i^{x_{i,o}} (1-\hat{c}_i)^{1-x_{i,o}}\right].$$

After normalization, we estimate $\hat{z}_o = 1$ if $p(z_o = 1|\mathbf{x}_o) \geq 0.5$ and $\hat{z}_o = 0$ otherwise. If the event location l_o is new, we estimate its location popularity $\hat{g}_o$ based on its prior counts.

6 Personalized Truth Finder for Spatial Events

In this section, we present the design of Personalized Truth finder for Spatial Events (PTSE), which is also a probabilistic model for truth discovery in crowdsourced event detection. PTSE differs from TSE (Section 4) in that it models personal location visit tendencies instead of location popularity to address the uncertainty in participants' mobility. As such tendencies are personalized while location popularity is not, PTSE is expected to better capture the probability that a participant will visit an event location than TSE. The properties discussed in Section 4.2 for TSE also hold for PTSE. We illustrate the model structure of PTSE in Fig. 5. In the following, we detail its new components and present the corresponding model inference algorithms.

6.1 Model Design

6.1.1 Personal Location Visit Tendency

We denote $f_{i,j}$ as the personal tendency that u_i will visit l_j. It is personalized as for the same event location l_j, $f_{i,j}$ may differ from one participant to another (in contrast to location popularity g_j, which is independent of u_i). It is inferred based on the assumption that u_i is also likely to visit l_j, if u_i and $u_{i'}$ have similar location visit behaviors and $u_{i'}$ has visited l_j.

In particular, we represent each participant's *revealed* locations as an N-vector L_i, whose jth element $L_{i,j} = 1$ if u_i makes a report at event location l_j, and $L_{i,j} = 0$ otherwise. We further assume that we have a set $\mathcal{D}$ of template users whose complete location histories are known (e.g., users who participate in a mobility study). We also represent each template user $u_d^{\mathcal{D}}$'s *visited* locations as an N-vector $L_d^{\mathcal{D}}$, whose jth element $L_{d,j}^{\mathcal{D}} = 1$ if $u_d^{\mathcal{D}}$ visited location l_j, and $L_{d,j}^{\mathcal{D}} = 0$ otherwise.

We then compute the similarity $w_{i,d}$ between a participant u_i and a template user $u_d^{\mathcal{D}}$ based on their location visit behaviors as

$$w_{i,d} = \frac{\#1(L_i \cap L_d^{\mathcal{D}})}{\#1(L_i \cup L_d^{\mathcal{D}})}, \tag{7}$$

where $\#1(L)$ denotes the number of ones in L. $w_{i,d}$ is then essentially the fraction of locations that both u_i and $u_d^{\mathcal{D}}$ visited, over the total number of locations that either u_i or $u_d^{\mathcal{D}}$ visited.

Finally, we compute the personal location visit tendency $f_{i,j}$ for u_i and l_j as

$$f_{i,j} = \frac{\sum_{d \in \mathcal{D}} w_{i,d} L_{d,j}^{\mathcal{D}}}{\sum_{d' \in \mathcal{D}} w_{i,d'}}. \tag{8}$$

$f_{i,j}$ is essentially a weighted combination of whether these template users visited l_j. It is computed based on the assumption that u_i is also likely to visit l_j, if u_i and $u_d^{\mathcal{D}}$ have similar location visit behaviors and $u_d^{\mathcal{D}}$ has visited l_j. As $L_{d,j}^{\mathcal{D}}$ can be either 0 or 1, we have $f_{i,j} \in [0, 1]$. As a result, $f_{i,j}$ can serve as a valid probability.

6.1.2 Participant's Location Visit Indicator

Given a participant u_i's personal location visit tendency $f_{i,j}$, we model that her location visit indicator $h_{i,j}$ for location l_j is generated from a Bernoulli distribution parameterized by $f_{i,j}$ as $h_{i,j} \sim \text{Bernoulli}(f_{i,j})$. In this way, if $f_{i,j}$ is large, u_i has a high chance to visit l_j.

6.2 Model Inference

In PTSE, we first infer $f_{i,j}$ based on (8). We then fix $f_{i,j}$ and use the collapsed Gibbs sampling to sample event labels z_j and location visit indicators $h_{i,j}$ as that for TSE (Section 5). The rules for sampling z_j are exactly the same as those for TSE. Similarly, we only need to sample $h_{i,j}$ when $x_{i,j} = 0$, as we can directly infer $h_{i,j} = 1$ if $x_{i,j} = 1$. The rules for sampling $h_{i,j}$ for $x_{i,j} = 0$ need to be revised as follows

$$\begin{aligned} p(h_{i,j} = 1|\mathbf{z}, \mathbf{H}^{-i,j}, \mathbf{X}) &\propto f_{i,j}\phi_{i,j}(1, z_j, 0) \\ p(h_{i,j} = 0|\mathbf{z}, \mathbf{H}^{-i,j}, \mathbf{X}) &\propto 1 - f_{i,j}. \end{aligned} \quad (9)$$

The procedures for estimating the prior truth probability s, and participant reliability a_i, b_i, and c_i are the same as those for TSE.

6.3 Online Truth Discovery

For the PTSE model, we can also perform fast online truth discovery for a new event z_o at l_o given new crowdsourced reports $\mathbf{x}_o$. In particular, we compute

$$\begin{aligned} &p(z_o = 1|\mathbf{x}_o) \\ &\propto \hat{s}\prod_i \left[f_{i,o}\hat{a}_i^{x_{i,o}}(1-\hat{a}_i)^{1-x_{i,o}} + (1-f_{i,o})\hat{c}_i^{x_{i,o}}(1-\hat{c}_i)^{1-x_{i,o}}\right], \\ &p(z_o = 0|\mathbf{x}_o) \\ &\propto (1-\hat{s})\prod_i \left[f_{i,o}\hat{b}_i^{x_{i,o}}(1-\hat{b}_i)^{1-x_{i,o}} + (1-f_{i,o})\hat{c}_i^{x_{i,o}}(1-\hat{c}_i)^{1-x_{i,o}}\right]. \end{aligned}$$

After normalization, we estimate $\hat{z}_o = 1$ if $p(z_o = 1|\mathbf{x}_o) \geq 0.5$ and $\hat{z}_o = 0$ otherwise.

7 Experiments

In this section, we evaluate the effectiveness and the efficiency of our proposed TSE and PTSE compared with several state-of-the-art approaches for truth discovery in crowdsourced detection of spatial events, on both real-world and synthetic datasets.

7.1 Methods in Comparison

We compare the following methods for truth discovery in crowdsourced event detection.

1) `MV`: the widely used Majority Voting method.
2) `TF`: the Truth Finder method proposed in [40], which utilizes the interdependency between source trustworthiness and claim confidence to find truth.
3) `GLAD`: the Generative model of Labels, Abilities, and Difficulties proposed in [38] for truth discovery in online crowdsourced image classification (the authors' code is used). It models the label of each image, the one-sided reliability of each labeler, and the difficulty of each image.
4) `LTM`: the Latent Truth Model proposed in [42] for truth discovery from conflicting web information. It models the latent truth and two-sided source reliability.
5) `EM`: the Expectation and Maximization method proposed in [34] for social sensing. It also models the latent truth and two-sided source reliability.
6) `TSE`: the proposed Truth finder for Spatial Events (Section 4). It models the latent truth, location popularity, location visit indicators, and three-way participant reliability.
7) `PTSE`: the proposed Personalized Truth finder for Spatial Events (Section 6). It models the latent truth, personal location visit tendencies, location visit indicators, and three-way participant reliability.
8) `TSEon`: the online version of `TSE` (Section 5.5).
9) `PTSEon`: the online version of `PTSE` (Section 6.3).

Except our proposed `TSE`, `PTSE`, and their online variants, other methods do not model location popularity, personal location visit tendencies, location visit indicators, and three-way participant reliability. In dealing with missing data, these other methods either ignore them or treat them as negative data. For events with binary truth, the former treatment will lead to a trivial conclusion that every event is true. We thus use the latter treatment for all these other compared methods.

We set the hyperparameters of `TSE` and `PTSE` as follows. We estimate $\lambda_{g_j,1}$ and $\lambda_{g_j,0}$ from a disjoint set of participants. We set $\lambda_{b_i,1} = 5$, $\lambda_{b_i,0} = 20$, and the hyperparameters for s and a_i to 5. We evaluate `TSEon` (`PTSEon`) through 5-fold cross validation, where the unsupervised training (i.e., inferring model parameters) is performed by `TSE` (`PTSE`).

7.2 Metrics

1) Precision (pre), recall (rec), and F1 score ($F1$). We use them to evaluate the effectiveness of these methods in estimating the event labels. They are defined as $pre = \frac{TP}{TP+FP}$, $rec = \frac{TP}{TP+FN}$, $F1 = 2\frac{pre \times rec}{pre+rec}$, where TP represents the number of true positives (a method infers an event is true when it is indeed true). The higher these metrics, the better a method performs.

2) CPU time. We use the CPU time to evaluate the efficiency of these methods. A shorter CPU time implies a faster method. All these methods are implemented in Matlab and evaluated using a Mac Desktop with Intel Core 2 Duo CPU and 4GB RAM.

We conduct three sets of experiments to evaluate the performance of the compared methods. The first two sets focus on truth discovery in crowdsourced detection of traffic lights on real-world datasets. The last set examines the impact of the average location popularity on the effectiveness of these methods through simulations. We report results based on 20 runs of tests for each

TABLE 2
Statistics of reports in Area 1 (SF; left half) and Area 2 (SF; right half) in Fig. 4(a).

Area	# pts	# total reports	# unique reported locs	# locs after clustering	# locs with ground truth	# reports used
Area 1	100	26,054	2,051	486	54 T + 46 F	2,996
Area 2	100	95,856	2,683	537	43 T + 57 F	3,627

experiment by randomly sampling the desired number of participants and events unless all of them are used. Only participants and events with at least two positive reports are kept.

7.3 Traffic Light Detection I

Dataset: We use the mobility dataset provided in [30] as our first experiment dataset. It contains time-stamped GPS location traces from 536 taxicabs in San Francisco, USA, with successive location updates recorded 1–60 seconds apart. This dataset was collected over 25 days in 2008. For notational convenience, we term this dataset as the SF dataset. We choose a region shown in Fig. 4(a) as our spatial area of interest, which spans roughly $3.5km \times 4.4km$ (an area with a reasonable size such that it is possible for participants to visit all the event locations inside it). We partition this area into approximately $40m \times 40m$ grid cells and then project the large number of distinct GPS locations into a much smaller set of cells. We further vertically divide this region into two areas of equal size, and denote the left and the right half as Area 1 (SF) and Area 2 (SF) respectively.

Task: The crowdsourcing task we consider is to detect the locations of traffic lights, following [35]. As vehicles usually wait at traffic light locations for a few seconds to a few minutes, by processing the waiting behaviors of vehicles driven by various participants, we will be able to crowdsource the locations of traffic lights. However, the waiting behavior is a noisy indicator of traffic lights since a car can also stop at stop signs or anywhere else on the road due to traffic jam or crossing pedestrians (false positive). Moreover, a car does not stop at traffic lights that are green (false negative). Furthermore, different drivers have different driving behaviors. For example, a careless driver may pass stop signs without stopping, while a careful driver may stop at stop signs for a relatively long time. These factors make the waiting behavior diverse, noisy, and participant-dependent.

Reports: In our experiment, we assume there is an application on each vehicle and if the vehicle waits at a location for 15–120 seconds, such a behavior triggers the application to issue a detection report (of a traffic light). Since the application uses the same criterion for data processing, when and where to issue a report is actually controlled by the participants, except that their reliability comes from their behaviors rather than mind. By randomly picking out 100 participants, we obtain 25,054 and 95,856 reports in Area 1 and 2 (SF) respectively, collectively identifying 2,051 and 2,683 distinct

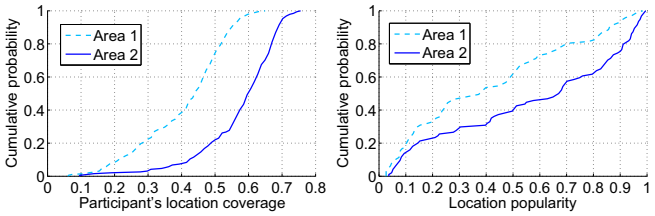


Fig. 6. Cumulative distributions of each participant's location coverage ($|l(u_i)|/|l|$) and each location's popularity ($|u(l_j)|/|u|$) in Area 1 and 2 (SF).

TABLE 3
Precisions, recalls, and F1 scores on inferring event labels for traffic light detection (SF).

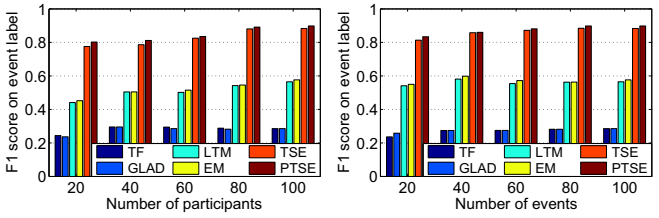
	Area 1 (SF)			Area 2 (SF)		
	pre	*rec*	$F1$	*pre*	*rec*	$F1$
MV	**1.000**	0.098	0.179	**1.000**	0.167	0.286
TF	**1.000**	0.098	0.179	**1.000**	0.167	0.286
GLAD	**1.000**	0.098	0.179	**1.000**	0.167	0.286
LTM	0.960	0.423	0.587	**1.000**	0.398	0.569
EM	0.956	0.431	0.594	**1.000**	0.404	0.576
TSE	0.970	0.895	0.931	0.995	0.794	0.883
PTSE	0.972	**0.923**	**0.947**	0.992	0.820	**0.898**
TSEon	0.942	0.907	0.924	0.976	0.798	0.878
PTSEon	0.938	0.918	0.928	0.959	**0.825**	0.887

cells respectively (listed in Table 2). To account for the location granularity of GPS devices and the fact that a vehicle may also wait at a certain distance from the traffic light due to the traffic queue, we further cluster these cells using a hierarchical clustering approach [8]. This procedure results in 486 and 537 cluster centers in Area 1 and 2 (SF). We then randomly pick out 100 of them in each area to annotate the ground truth using the Street View in Google Maps (some locations are rectified to the closest road intersections, as cluster centers may be on the top of buildings).

Data analysis: We define a participant's location coverage as $|l(u_i)|/|l|$, where $|l(u_i)|$ denotes the number of event locations visited by u_i and $|l|$ denotes the total number of event locations. A location's popularity can be expressed as $|u(l_j)|/|u|$, where $|u(l_j)|$ denotes the number of participants that visited l_j and $|u|$ denotes the total number of participants. Fig. 6 plots the cumulative distributions of these two metrics. As can be seen, a participant can cover at most 62% and 75% of all the event locations in Area 1 and 2 (SF) respectively. Only around 20% and around 40% of event locations have popularity of over 0.8 in Area 1 and 2 (SF) respectively. Some event locations are visited by almost all the participants while some are visited by less than 5%. These results suggest that mobility is an important factor that causes missing reports in mobile crowdsourcing.

Results: We utilize a disjoint set of 200 participants to estimate the prior counts of location popularity for TSE, and to infer personal location visit tendencies for PTSE.

Table 3 lists the precisions, recalls, and F1 scores of all the methods in the two areas when $M = 100$ and $N = 100$. It is observed that TSE, PTSE, and their



(a) F1 versus M ($N = 100$) (b) F1 versus N ($M = 100$)

Fig. 7. F1 score on estimating event labels versus (a) the number of participants M when $N = 100$ and (b) the number of events N when $M = 100$ in Area 2 (SF).

online variants outperform other methods in the F1 scores, showing that they can better handle missing reports and more accurately infer the truths of events in crowdsourced detection of spatial events. PTSE performs slightly better than TSE. It is because in PTSE, each location visit indicator is impacted by the personal location visit tendency $f_{i,j}$, while in TSE, each location visit indicator is impacted by the collective location popularity g_j. As a result, PTSE can better tailor to each participant's specific location visit behaviors. All the other methods cannot tackle mobility issues and perform much worse. They are prone to infer that most events are false due to the large number of missing reports and thus fail to detect lots of true events, resulting in high precisions but low recalls.

MV, TF, and GLAD perform worst. As TF and GLAD cannot handle unknown mobility, these advanced methods are not effective than the simple MV for crowdsourced event detection. GLAD models only a single correct rate for participant reliability. However, as discussed in Section 4.1.4, we need to model different aspects of participant reliability for truth discovery in crowdsourced event detection. LTM and EM perform comparably and much better than MV, TF, and GLAD, since they model two-sided participant reliability. However, they still cannot tackle the mobility issues and thus fail to detect lots of positive events (i.e., biased by missing reports).

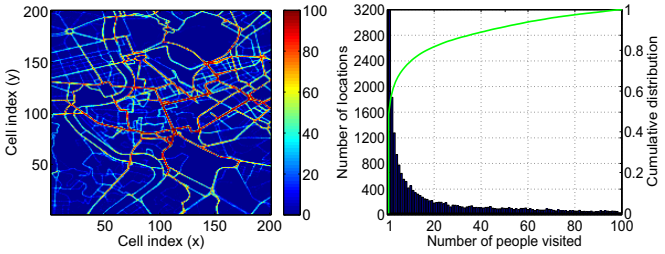
Fig. 7 plots the F1 scores on estimating the event labels versus the number of participants and the number of events in Area 2 (SF). It is observed that TSE and PTSE perform much better than all the other methods in all the cases, and they can benefit from more available information to improve their performance.

7.4 Traffic Light Detection II

In this section, we further evaluate the performance of different methods utilizing another real-world dataset.

Dataset: We use the mobility dataset[2] provided in [6] as our second experiment dataset. It contains time-stamped GPS location traces from approximately 320 taxicabs in Rome, Italy. It was collected over 30 days in 2014. We term this dataset as the RM dataset. We choose a region shown in Fig. 8(a) as our spatial area of interest. Its size is the same as the area in SF shown in Fig. 4(a).

2. http://crawdad.org/roma/taxi/.



(a) (b)

Fig. 8. (a) Location heat map in the center of Rome (latitude: 41.88 to 41.92; longitude: 12.46 to 12.50; each point represents an approximately $20m$ by $20m$ grid cell). Each point shows the number of distinct people (totally 100) visited that location. (b) Distribution of the number of people visited a specific location.

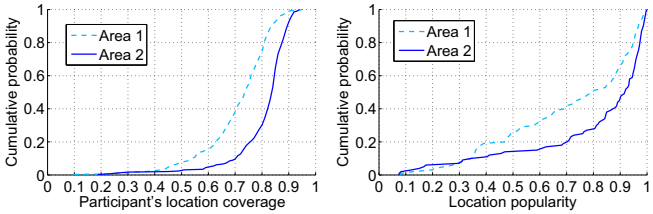


Fig. 9. Cumulative distributions of each participant's location coverage ($|l(u_i)|/|l|$) and each location's popularity ($|u(l_j)|/|u|$) in Area 1 and 2 (RM).

We also vertically divide this region into two areas of equal size and denote the left and the right half as Area 1 (RM) and Area 2 (RM) respectively.

Task and reports: We also consider the crowdsourcing task as to detect the locations of traffic lights. Following the same procedure as that for the SF dataset, we randomly pick out 100 participants, and obtain 105,193 and 189,255 reports in Area 1 and 2 (RM) respectively, collectively identifying 2,790 and 2,607 cells respectively (listed in Table 4). After clustering, we obtain 426 and 394 cluster centers in Area 1 and 2 (RM) respectively. We then randomly pick out 100 of them in each area to annotate the ground truth.

Data analysis: Fig. 9 plots the cumulative distributions of each participant's location coverage and each location's popularity. It is observed that each participant can cover at most 93% and 93% of all the event locations in Area 1 and 2 (RM) respectively. Moreover, around 50% and around 70% of event locations have popularity of over 0.8 in Area 1 and 2 (RM) respectively. Comparing with the analysis on the SF dataset (Fig. 6 and Section 7.3), participants in the RM dataset more actively visit the event locations, and most event locations in the RM dataset have relatively high popularity.

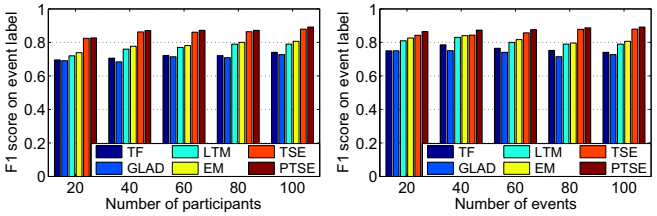
Results: We utilize a disjoint set of 100 participants to estimate the prior counts of location popularity for TSE, and to infer personal location visit tendencies for PTSE. Table 5 lists the results of all the methods when $M = 100$ and $N = 100$. We again observe that TSE, PTSE, and their online variants outperform other methods in the F1 scores, showing their effectiveness for truth discovery in

TABLE 4
Statistics of reports in Area 1 (RM; left half) and Area 2 (RM; right half) in Fig. 8(a).

Area	# pts	# total reports	# unique reported locs	# locs after clustering	# locs with ground truth	# reports used
Area 1	100	105,193	2,790	426	49 T + 51 F	3,755
Area 2	100	189,255	2,607	394	52 T + 48 F	4,616

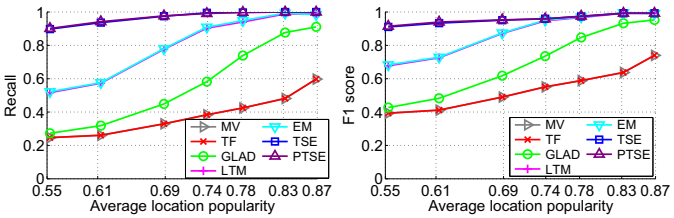
TABLE 5
Precisions, recalls, and F1 scores on inferring event labels for traffic light detection (RM).

	Area 1 (RM)			Area 2 (RM)		
	pre	*rec*	*F*1	*pre*	*rec*	*F*1
MV	**1.000**	0.605	0.754	**0.973**	0.769	0.859
TF	0.980	0.596	0.741	**0.973**	0.774	0.862
GLAD	**1.000**	0.572	0.728	0.966	0.786	0.867
LTM	0.944	0.685	0.794	0.963	0.838	0.896
EM	0.948	0.703	0.807	0.961	0.852	0.903
TSE	0.961	0.810	0.879	0.952	**0.886**	0.918
PTSE	0.967	0.828	**0.892**	0.959	**0.886**	**0.921**
TSEon	0.940	0.816	0.874	0.942	**0.886**	0.913
PTSEon	0.952	**0.830**	0.887	0.948	**0.886**	0.916



(a) F1 versus M ($N = 100$) (b) F1 versus N ($M = 100$)

Fig. 10. F1 score on estimating event labels versus (a) the number of participants M when $N = 100$ and (b) the number of events N when $M = 100$ in Area 1 (RM).



(a) Recall (b) F1 score

Fig. 11. Performance vs. average location popularity. (a) Recall. (b) F1 score. $M = 40$ and the maximum $N = 600$.

crowdsourced event detection. However, MV, TF, GLAD, LTM, and EM perform much better on the RM dataset than on the SF dataset. We investigate the possible reason in the next section.

Fig. 10 plots the F1 scores on estimating the event labels versus the number of participants and the number of events in Area 1 (RM). We again observe that TSE and PTSE perform better than all the other methods in all the cases, and they can benefit from more available information to improve their performance.

7.5 Impact of Average Location Popularity

After observing the cumulative distributions of location popularity (Figs. 6 and 9) and the experimental results on the two real-world datasets (Sections 7.3 and 7.4), we hypothesize that the average location popularity over the event locations plays an important role in impacting the performance of different methods. We expect that MV, TF, GLAD, LTM, and EM will perform poorly when the average location popularity is low but they will perform better as the average location popularity increases. This is because high average location popularity indicates that most event locations are indeed visited by most participants and treating missing reports as negative reports becomes more accurate. In contrast, TSE and PTSE are expected to perform well under varying average location popularity as they explicitly handle unknown mobility issues, in addition to unknown reliability issues.

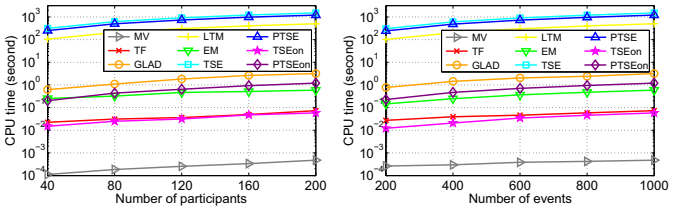
In order to systematically verify this hypothesis, we conduct a simulation study in this section to examine how the average location popularity over the event locations impacts the performance of different methods.

Dataset: We randomly sample $N = 600$ event locations from the SF dataset (as the location popularity in the SF dataset is more diverse than that in the RM dataset). Next, we select event locations whose popularity is above a given threshold, ranging from 0.1 to 0.7 with an increment of 0.1. In this way, the average location popularity of these selected event locations will increase as the threshold increases. In particular, the resulting average location popularity is 0.55, 0.61, 0.69, 0.74, 0.78, 0.83, and 0.87 respectively. After determining event locations, we randomly sample $M = 40$ participants with GPS location traces, and generate reports $x_{i,j}$ according to the process in Section 4.1.5 with $c_i = 0$, where we assume 30% of participants are reliable (with $a_i \in [0.8, 1]$ and $b_i \in [0, 0.2]$) and the remaining are unskilled (with $a_i, b_i \in [0.4, 0.6]$). In order not to favor any methods, we set half of the events as true and half as false.

Results: Fig. 11 depicts the performance of various methods versus the average location popularity (precision curves are not shown as all the methods have very high precisions which are above 0.9). It is observed that TSE and PTSE perform well regardless of the average location popularity, resulting in a F1 score of above 0.9 in all the cases. When the average location popularity is 0.55, MV, TF, and GLAD can only result in a F1 score of around 0.4. LTM and EM can result in a F1 score of around 0.7. However, these methods perform much better when the average location popularity increases. As in reality, event locations are random and we cannot expect a high average location popularity, TSE and PTSE are more appropriate solutions for truth discovery in crowdsourced detection of spatial events.

7.6 Efficiency

Finally, we plot in Fig. 12 the CPU time of all the methods, where we vary the number of events from 200 to 1000 and vary the number of participants from 40 to 200. It is observed that TSE is the most time-consuming,



(a) CPU time vs. M ($N = 1000$) (b) CPU time vs. N ($M = 200$)

Fig. 12. CPU time versus (a) M when $N = 1000$ and (b) N when $M = 200$.

followed by PTSE and LTM. This is because these three methods use the collapsed Gibbs sampling (we use 400 runs) for model inference, and such sampling is known to be slow [8]. LTM only needs to sample z_j, while TSE and PTSE need to sample not only z_j but also $h_{i,j}$. However, PTSE does not need to update the counts $n_{j,1}^{-i}$ while TSE does. As a result, TSE is the most time-consuming, followed by PTSE and LTM.

The online algorithms TSEon and PTSEon are much faster than TSE and PTSE. TSEon is faster than PTSEon, as computing $\hat{g}_o$ for all new event locations is fast, while computing $f_{i,o}$ for all u_i and l_o according to (8) is more time-consuming. TSEon is able to perform truth discovery when $M = 200$ and $N = 1000$ in around 0.06 second and PTSEon is able to finish such a task in around 1 second, showing their efficiency. GLAD is more time consuming than EM, followed by TF. MV is the most time efficient as it is a model-less method. However, it does not perform well in terms of the effectiveness as shown in the previous experiments.

8 Future Work

Location obfuscation. Location obfuscation is a technique used in location-based services or information systems to protect the location of a user by slightly altering, substituting, or generalizing her location in order to avoid reflecting her real position [4]. In case participants obfuscate their locations when making reports, our assumptions on the RRA c_i will be violated. However, as obfuscated locations are different from true event locations, the reported locations from a participant who uses location obfuscation will naturally have less overlap with the reported locations by other normal participants. We can thus add a filtering step before employing our models to detect and remove reports from participants who obfuscate their locations.

Dependent reports. We currently assume that participants independently make reports. However, sources can sometimes be dependent and such dependency can undermine the wisdom of crowd [21]. One possible solution is to apply copy detection methods between sources [14]. Alternatively, we can directly incorporate source dependency in the modeling [9], [13], [31], [36].

Exploiting geographical information. In TSE, we model that the probability that a participant visits an event location depends on individual location's popularity. Alternatively, we can also exploit the distance between pairwise candidate event locations to infer such a probability, as [39] shows that people are more likely to visit nearby locations than distant locations.

Scalable model inference algorithms. We currently use the collapsed Gibbs sampling to infer model parameters for TSE and PTSE. As it is a batch algorithm, it is not scalable. Although we also propose scalable online variants of TSE and PTSE, the training phase is still performed by sampling. To handle large datasets, we need to design more scalable model inference algorithms, possibly exploring the MapReduce framework [10], [12] which is designed to process large datasets with a parallel, distributed algorithm on a cluster.

9 Conclusion

In this paper, we have proposed two probabilistic models to the problem of truth discovery in crowdsourced detection of spatial events. In TSE, we model location popularity, participants' location visit indicators, truths of events, and three-way participant reliability in a unified framework. In PTSE, we further model personal location visit tendencies instead of location popularity to address the mobility issue. We demonstrate that these models can effectively handle ambiguous missing reports, and automatically infer the truths of events and different aspects of participant reliability without any supervision or location tracking. These models perform well regardless of the average location popularity while most existing methods do not. Our proposed models are thus more appropriate solutions for truth discovery in crowdsourced detection of spatial events.

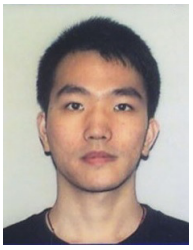
Acknowledgements

This research is based upon work supported in part by the U.S. ARL and U.K. Ministry of Defense under Agreement Number W911NF-06-3-0001, and by the NSF under award CNS-1213140. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views or represent the official policies of the NSF, the U.S. ARL, the U.S. Government, the U.K. Ministry of Defense or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

[1] Amazon mechanical turk. https://www.mturk.com/mturk/welcome.
[2] Field agent. http://www.fieldagent.net.
[3] Gigwalk. http://gigwalk.com.
[4] Location obfuscation. http://en.wikipedia.org/wiki/Location_obfuscation.
[5] Taskrabbit. http://www.taskrabbit.com.
[6] R. Amici et al. Performance assessment of an epidemic protocol in vanet using real traces. *Procedia Computer Science*, 40:92–99, 2014.
[7] A. R. Beresford and F. Stajano. Location privacy in pervasive computing. *Pervasive Computing, IEEE*, 2(1):46–55, 2003.

[8] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*. Springer New York, 2006.
[9] A. Das et al. Debiasing social wisdom. In *KDD*, pages 500–508. ACM, 2013.
[10] A. S. Das et al. Google news personalization: scalable online collaborative filtering. In *WWW*, pages 271–280. ACM, 2007.
[11] A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied Statistics*, pages 20–28, 1979.
[12] J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.
[13] X. L. Dong et al. Integrating conflicting data: the role of source dependence. *VLDB Endowment*, 2(1):550–561, 2009.
[14] X. L. Dong et al. Truth discovery and copying detection in a dynamic world. *VLDB Endowment*, 2(1):562–573, 2009.
[15] M. C. Gonzalez et al. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.
[16] S. Isaacman et al. Human mobility modeling at metropolitan scales. In *MobiSys*, pages 239–252. ACM, 2012.
[17] S. Kisilevich et al. *Spatio-temporal clustering*. Springer, 2010.
[18] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
[19] K. Lin et al. Energy-accuracy trade-off for continuous mobile device location. In *MobiSys*, pages 285–298. ACM, 2010.
[20] J. S. Liu. The collapsed gibbs sampler in bayesian computations with applications to a gene regulation problem. *Journal of the American Statistical Association*, 89(427):958–966, 1994.
[21] J. Lorenz et al. How social influence can undermine the wisdom of crowd effect. *Proceedings of the National Academy of Sciences*, 108(22):9020–9025, 2011.
[22] Y. Lou et al. Map-matching for low-sampling-rate gps trajectories. In *GIS*, pages 352–361. ACM, 2009.
[23] P. Mohan et al. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In *SenSys*, pages 323–336. ACM, 2008.
[24] M. Musthag and D. Ganesan. Labor dynamics in a mobile micro-task market. In *CHI*, pages 641–650. ACM, 2013.
[25] R. W. Ouyang et al. Energy efficient assisted gps measurement and path reconstruction for people tracking. In *GLOBECOM*, pages 1–5. IEEE, 2010.
[26] R. W. Ouyang et al. If you see something, swipe towards it: crowdsourced event localization using smartphones. In *UbiComp*, pages 23–32. ACM, 2013.
[27] R. W. Ouyang et al. Truth discovery in crowdsourced detection of spatial events. In *CIKM*, pages 461–470. ACM, 2014.
[28] J. Paek et al. Energy-efficient rate-adaptive gps-based positioning for smartphones. In *MobiSys*, pages 299–314. ACM, 2010.
[29] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). In *COLING*, pages 877–885. Association for Computational Linguistics, 2010.
[30] M. Piórkowski et al. A parsimonious model of mobile partitioned networks with clustering. In *COMSNETS*, pages 1–10. IEEE, 2009.
[31] G.-J. Qi et al. Mining collective intelligence in diverse groups. In *WWW*, pages 1041–1052. ACM, 2013.
[32] V. C. Raykar et al. Learning from crowds. *The Journal of Machine Learning Research*, 99:1297–1322, 2010.
[33] S. Reddy et al. Recruitment framework for participatory sensing data collections. In *Pervasive Computing*, pages 138–155. Springer, 2010.
[34] D. Wang et al. On truth discovery in social sensing: a maximum likelihood estimation approach. In *IPSN*, pages 233–244. ACM, 2012.
[35] D. Wang et al. On credibility estimation tradeoffs in assured social sensing. *IEEE JSAC*, 31(6):1026–1037, 2013.
[36] T. Wang et al. Quantifying herding effects in crowd wisdom. In *KDD*, pages 1087–1096. ACM, 2014.
[37] P. Welinder et al. The multidimensional wisdom of crowds. In *NIPS*, pages 2424–2432, 2010.
[38] J. Whitehill et al. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *NIPS*, pages 2035–2043, 2009.
[39] M. Ye et al. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, pages 325–334. ACM, 2011.
[40] X. Yin, J. Han, and P. S. Yu. Truth discovery with multiple conflicting information providers on the web. *IEEE TKDE*, 20(6):796–808, 2008.
[41] X. Yin and W. Tan. Semi-supervised truth discovery. In *WWW*, pages 217–226. ACM, 2011.
[42] B. Zhao et al. A bayesian approach to discovering truth from conflicting sources for data integration. *VLDB Endowment*, 5(6):550–561, 2012.
[43] Z. Zhuang et al. Improving energy efficiency of location sensing on smartphones. In *MobiSys*, pages 315–330. ACM, 2010.



Robin Wentao Ouyang received the Ph.D. degree in electronic and computer engineering from Hong Kong University of Science and Technology (HKUST), Hong Kong, China, in 2011, and the B.E. degree in electronic engineering from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2007. He is currently a postdoctoral associate with the Department of Computer Science, University of California, Los Angeles (UCLA), USA. His current research interests include human computation, data mining and mobile computing.

Mani Srivastava received the B.Tech. degree from the Indian Institute of Technology Kanpur, Kanpur, India, in 1985, and the M.S. and Ph.D. degrees from the University of California Berkeley, Berkeley, in 1987 and 1992, respectively.

He was with Bell Laboratory Research, Murray Hill, NJ. He joined the University of California, Los Angeles, as a Faculty Member, in 1997, where he is currently a Professor of the Electrical Engineering and Computer Science Department. His current research interests include embedded systems, low-power design, wireless networking, and pervasive sensing.

Dr. Srivastava is affiliated with the National Science Foundation Science and Technology Center on Embedded Networked Sensing, where he co-leads the System Research Area. He currently serves as the Steering Committee Chair of the IEEE TRANSACTIONS ON MOBILE COMPUTING.

Alice Toniolo is a postdoctoral researcher in the Computing Science Department at the University of Aberdeen (UK). Her interest is in computational models of argumentation for reasoning and dialogue. She was awarded her PhD in Computing Science by the University of Aberdeen (UK) in 2013.

Timothy J. Norman is a professor of computing science at the University of Aberdeen, Scotland, UK. His research interests are in multi-agent systems, computational models of trust, policies and argumentation, and how these methods are applied to problems of information interpretation and management. Dr. Norman has a PhD in Computer Science from University College London, UK.