Truth Discovery in Crowdsourced Detection of Spatial Events

Robin Wentao Ouyang, Mani Srivastava
University of California, Los Angeles
Los Angeles, CA, USA
{wouyang, mbs}@ucla.edu

Alice Toniolo, Timothy J. Norman
University of Aberdeen
Aberdeen, UK
{a.toniolo, t.j.norman}@abdn.ac.uk

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 quality control is to discover true events from diverse and noisy participants' reports. This truth discovery 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 the discovered truth. In this paper, we propose a new method to tackle this truth discovery problem through principled probabilistic modeling. In particular, we integrate the modeling of location popularity, location visit indicators, truth of events and three-way participant reliability in a unified framework. The proposed model is thus capable of efficiently handling various types of uncertainties and automatically discovering truth without any supervision or the need of location tracking. Experimental results demonstrate that our proposed method outperforms existing state-of-the-art truth discovery approaches in the mobile crowdsourcing environment.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*; H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

Mobile crowdsourcing; quality control; graphical models

1. INTRODUCTION

The growing smartphone user base has enabled mobile crowdsourcing applications on a large scale [15]. Several commercial markets such as Field Agent [3], Gigwalk [4] 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. These events are arbitrary phenomena that the task requester is interested in, e.g., potholes on streets [14], graffiti on walls [17] and bike racks in public places [23].

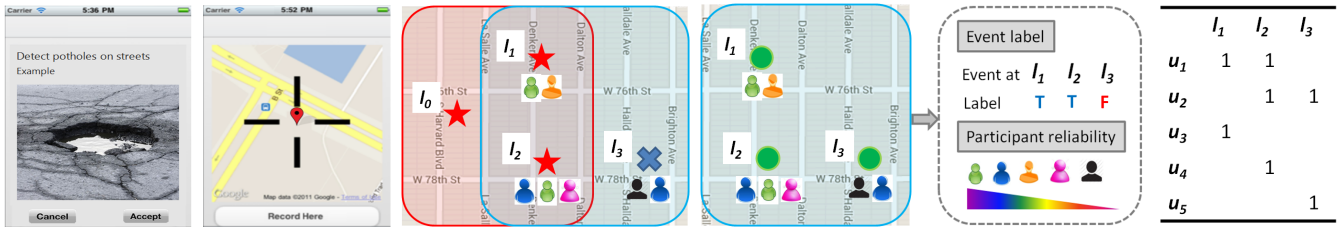
Consider the task of detecting the locations of potholes as an example, where Figure 1a shows a user interface for task instruction. Since the number of possible event locations is huge and most locations normally do not have an event (e.g., no potholes), a participant uses her smartphone to make a report (tagged with time and location as shown in Figure 1b) 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 (Figure 1c), there is a demand for efficient algorithms to handle these diverse and noisy participants' reports and automatically discover the truth (Figure 1d).

This truth discovery problem is uniquely distinct from its online counterpart in that it involves uncertainties in both participants' mobility and reliability. Since participants only sporadically reveal their locations when reporting events for the geotagging purpose, we cannot obtain their detailed trajectories. This imposes a significant challenge in interpreting missing reports at candidate event locations, which consequently impacts the quality of truth discovery (shown in Figure 1e where a positive and a missing report are annotated as a "1" and a blank space respectively). A missing report is ambiguous since it can be due to either the mobility issue that a participant did not visit a location and thus 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 truth of the event and the participant reliability while the latter does.

A possible solution to this problem is to continuously track participants' locations such that the missing reports corresponding to unvisited locations are ignored and those corresponding to visited locations are treated as negative. We illustrate this strategy in Figure 2a, where reports that are taken into consideration (as participants visited the corresponding locations) are marked with gray backgrounds and the inferred "negative" reports are annotated as "-1"s. After eliminating the uncertainty in mobility, we can apply existing truth discovery methods for online crowdsourcing. However, location tracking is impractical as it raises severe privacy and energy issues [6, 30]. Alternatively, one can try to reconstruct a participant's mobility path. Nevertheless, machine learning-based path reconstruction methods [16] require historical location traces

CIKM'14, November 3–7, 2014, Shanghai, China.

http://dx.doi.org/10.1145/2661829.2662003.



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

Figure 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.

Figure 2: Different strategies of handling missing reports. A entry with a gray (white) background means that the corresponding report is (not) taken into consideration for truth discovery. A "1" indicates a positive report, a "-1" indicates that the missing report is treated as negative 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 reports.

which can only be obtained through tracking and such methods will easily fail when a participant deviates from her usual paths. Map-matching-based path reconstruction methods [13] require road network information and they will easily fail if the time interval between consecutively revealed locations is larger than 5 minutes.

Strategies used for tackling missing data in related domains may be useful. For example, Raykar et al. [22] simply ignore the missing data for online crowdsourced binary image classification. This is because online crowd workers are required to provide either a positive or a negative response, 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 (Figure 2b). This will lead to a trivial conclusion that every reported event is true, which is obviously erroneous. In tackling conflicting Web information for data integration, Zhao et al. [29] 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, each can be either true or false. Each missing report will then become a negative report and imply a lack of an event (Figure 2c). 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. A work on social sensing [24] similarly treats missing reports as negative reports.

In this paper, we propose a new method to tackle the truth discovery problem in crowdsourced event detection through principled probabilistic modeling. We observe that a participant's likelihood of reporting an event depends on three factors: 1) whether the participant 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, 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 (e.g., shopping malls) naturally attract more people while others (e.g., country roads) attract fewer. Moreover, we treat the truth of events as latent variables and model three-way participant reliability, including true positive rate and false positive rate while present at a location and reporting rate while absent from a location. By doing so, positive and missing reports become random variables generated by conditioning on all these factors. Our approach thus directly incorporates the mobility issues in the model, can efficiently handle missing reports and can automatically infer the truth of events and different aspects of participant reliability. Moreover, it is unsupervised and avoids location tracking.

In summary, this paper makes the following contributions:

1. We propose to address the truth discovery problem in crowdsourced detection of spatial events.
2. We propose an unsupervised Bayesian probabilistic approach which models location popularity, location visit indicators, truth of events and three-way participant reliability in an integrated framework. Moreover, this approach does not require location tracking.
3. We develop an efficient algorithm for model inference via collapsed Gibbs sampling.

The remainder of this paper is organized as follows. We formalize the problem in Section 2. We then introduce our proposed model and develop the inference algorithm in Sections 3 and 4. Experimental setup and results are presented in Sections 5 and 6. Several possible improvements and related work are discussed in Sections 7 and 8. Finally, we conclude the paper in Section 9.

2. PROBLEM STATEMENT

We formally define the truth discovery problem in this section. Consider a scenario where a group of participants joins a task to report a specific type of spatial events (e.g., potholes). A participant uses her smartphone to make a report r upon detection. Each report $r = (u, l, t)$ contains the participant ID u, the location l of the event (e.g., by GPS) and the time t of the report.

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

$$x_{i,j} = 1 \text{ if } \exists r|r.u = u_i, r.l = l_j,$$

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 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 and the participant's reliability, while the latter does. However, participants' detailed mobility traces, which can be used to distinguish these two cases, are not available due to privacy and energy issues.

Our problem of truth discovery in mobile crowdsourced detection of spatial events is to infer 1) which reported events with locations in $\mathcal{L}$ are true and which are false and 2) which participants in $\mathcal{U}$ are reliable, based on the report matrix $\mathbf{X}$ with only positive and missing reports. This problem is visually illustrated in Figure 1d.

3. GRAPHICAL MODEL

In this section, we present our proposed probabilistic graphical model for the truth discovery problem in crowdsourced detection of spatial events. We first discuss our intuitions and the model components, and then illustrate some properties of the proposed model.

3.1 Model Details

We consider the process of how a 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. If a participant is not present at a location, she cannot make a report there since her location is recorded by her mobile device when reporting.

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

Figure 3 shows the graphical structure of our proposed model. Each node in the graph represents a random variable. Dark shaded nodes indicate observed variables, and light nodes represent latent variables and model parameters. Hyperparameters that correspond to the prior distributions are omitted for simplicity. A plate with a number such as M as its label means that there are M nodes of this kind. For ease of illustration, we list the notations used in this paper in Table 1. We summarize the generation process of our model in Algorithm 1 and detail its components below.

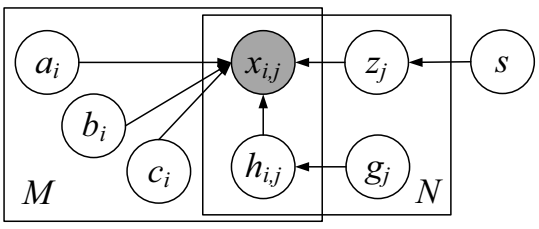
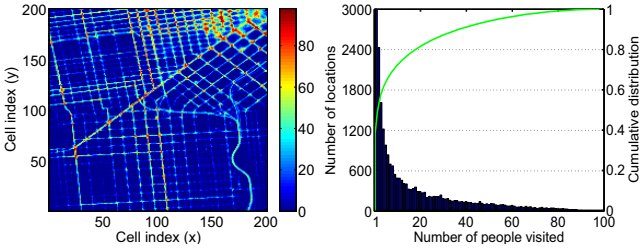


Figure 3: Graphical model.

Algorithm 1 Generation process

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 $x_{i,j} \sim \mathsf{Bernoulli}(c_i)$



(a) (b)

Figure 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) that visited that location. (b) Distribution (PDF and CDF) of the number of people that visited a specific location.

3.1.1 *Location Popularity*

It is clear that in the physical world, participants do not randomly visit different locations but with certain patterns. 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 which contains time-stamped GPS location traces for 536 taxicabs over a span of roughly one month in the city of San Francisco [20]. Figure 4a 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 more people while some by much fewer. We find that locations with high popularity mostly correspond to gas stations, crossroads and popular highways. Figure 4b plots the distribution of the number of people that 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 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 with hyperparameters $(\lambda_{g_{j,1}}, \lambda_{g_{j,0}})$, representing the prior counts of the number of distinct participants visited and did not visit location l_j respectively from a population.

3.1.2 *Participants' Location Visit Indicators*

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

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 $h_{i,j}$
$x_{i,j}$; $\mathbf{X}$	report made by participant u_i on the event at l_j ($x_{i,j} \in \{0,1\}$); $\mathbf{X} = \{x_{i,j}\}$
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; $\mathbf{g} = \{g_j\}$
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 for the Beta distribution (prior) for variable v, $v \in \{s, g_j, a_i, b_i, c_i\}$

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., $h_{i,j} \sim \mathsf{Bernoulli}(g_j)$. In this way, a participant has a higher chance to visit more popular locations.

3.1.3 Event Labels

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 $z_j \sim \mathsf{Bernoulli}(s)$.

3.1.4 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 will have different attitudes towards reporting true and false events upon observation. A reliable participant will mostly report detection for true events but 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 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 reasonable to use a single correct rate (e.g., as that in [19, 27, 28]) 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 the reports for events that a participant has an opportunity to observe (i.e., visited the event locations).

Second, as has been discussed previously, if a participant did not visit a location l_j, 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 the ease of illustration. 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 at l_j, i.e., $c_i = p(x_{i,j} = 1 | h_{i,j} = 0)$. Since a participant cannot evaluate the event label when she is not at the event location, we model this probability to be independent of the event label z_j. 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 the 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. Therefore, we specify a large $\lambda_{c_{i,0}}$ and a small $\lambda_{c_{i,1}}$ to make c_i conform to such real-world physical constraints. The introduction of this probability 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 reports under different combinations of $h_{i,j}$ and z_j. Note that each participant may have different reliability in different types of crowdsourcing tasks.

3.1.5 Reports

Finally, we 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 report, ii) the participant visited the event location which did not have any target event, and she did not report, and iii) the participant did not visit the event location and thus could not report. 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, and is parameterized by different participant reliability a_i, b_i and c_i.

Formally, we model

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

3.2 Model Analysis

We discuss several properties of our proposed model below.

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

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

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 and FPR. 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. In more general scenarios, these two possibilities for a missing report are combined through g_j. Positive reports can be explained similarly.

2) Location tracking is avoided. Our model does not require continuous location tracking for each participant to disambiguate the cause of missing reports, and thus it avoids 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, 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 other resources where location tracking is not a concern (e.g., experiments about taxis or for studying human mobility). Moreover, location popularity for each specific task can be jointly estimated with other model parameters from the corresponding data. In contrast, location tracking needs to be performed repeatedly for all the participants in each specific task (as new participants may join and existing participants may change mobility paths over time).

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, our model separates 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 in the truth discovery process. In the absence of such knowledge, we can simply use uniform priors. The Beta distribution is utilized as the prior distribution because it is the conjugate prior of the Bernoulli distribution. It can lead to posterior distributions having the same functional form as the prior, resulting in a greatly simplified and efficient Bayesian analysis [7].

4. INFERENCE ALGORITHM

In this section, we discuss how to perform inference to estimate 1) latent variables: event labels and participant's location visit indicators and 2) model parameters: participant reliability and location popularity from the model, given the report matrix $\mathbf{X}$. Algorithm 2 summarizes the model inference procedure.

Algorithm 2 Model Inference

Input: Reports $x_{i,j}$
Output: Latent variables $z_j, h_{i,j}$ and model parameters 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 $h_{i,j}=1$
Calculate all the counts n_q^{-j}, $n_{j,k}^{-i}$, $n_{i,k,q,v}^{-j}$
{*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$(p_j^1/(p_j^1+p_j^0))$ and 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$(p_{i,j}^1/(p_{i,j}^1+p_{i,j}^0))$ and update counts
end for
{*Estimate event labels and location visit indicators*}
Estimate $\hat{p}(z_j=1)$ and $\hat{p}(h_{i,j}=1)$ 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 $\hat{h}_{ij}$
{*Estimate model parameters*}
Estimate $\hat{a}_i$, $\hat{b}_i$ and $\hat{g}_j$ according to (3) and (4)
return $\hat{z}_j, \hat{h}_{i,j}, \hat{a}_i, \hat{b}_i, \hat{g}_j$

4.1 Estimating Event Labels and Location Visit Indicators

Given the data matrix $\mathbf{X}$ and the 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 [7]. For example, to infer the event labels, we need to solve

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

Given the complex form of the joint distribution (refer to Figure 3), direct optimization is difficult to perform, especially when z and h can take on only integers. Therefore, we resort to an efficient algorithm, which is the collapsed Gibbs Sampling [11], for model inference. In our implementation, we integrate out all the model parameters and only sample the latent variables z_j and $h_{i,j}$.

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.

$$p(z_j = 1|\mathbf{z}^{-j}, \mathbf{H}, \mathbf{X}) \propto (n_1^{-j} + \lambda_{s,1}) \prod_i f_{i,1,x_{i,j}}$$
$$p(z_j = 0|\mathbf{z}^{-j}, \mathbf{H}, \mathbf{X}) \propto (n_0^{-j} + \lambda_{s,0}) \prod_i f_{i,0,x_{i,j}} \quad (1)$$

where

$$f_{i,1,d} = \frac{n_{i,1,1,d}^{-j} + \lambda_{a_i,d}}{\sum_{d'}(n_{i,1,1,d'}^{-j} + \lambda_{a_i,d'})}, f_{i,0,d} = \frac{n_{i,1,0,d}^{-j} + \lambda_{b_i,d}}{\sum_{d'}(n_{i,1,0,d'}^{-j} + \lambda_{b_i,d'})}.$$

In the above expressions, the counts $n_{i,k,q,v}^{-j}$ reflect the reliability of u_i based on her reports on events other than that at l_j. The first part in (1) carries information from other event labels and the second part carries information from the reports made by all the participants on other events (except that at l_j). Note that, in $f_{i,1,d}$ and $f_{i,0,d}$, the counts only relates to $h=1$. 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.

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 $x_{i,j} = 1$.

$$p(h_{i,j} = 1|\mathbf{z}, \mathbf{H}^{-i,j}, \mathbf{X}) \propto (n_{j,1}^{-i} + \lambda_{g_j,1}) f_{i,z_j,0}$$

$$p(h_{i,j} = 0|\mathbf{z}, \mathbf{H}^{-i,j}, \mathbf{X}) \propto (n_{j,0}^{-i} + \lambda_{g_j,0}) \frac{n_{i,0,-,0}^{-j} + \lambda_{c_i,0}}{\sum_d (n_{i,0,-,d}^{-j} + \lambda_{c_i,d})}. \quad (2)$$

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).

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.

4.2 Estimating Participant Reliability

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

$$\hat{a}_i = \frac{\mathbb{E}(n_{i,1,1,1}) + \lambda_{a_i,1}}{\sum_d [\mathbb{E}(n_{i,1,1,d}) + \lambda_{a_i,d}]}, \; \hat{b}_i = \frac{\mathbb{E}(n_{i,1,0,1}) + \lambda_{b_i,1}}{\sum_d [\mathbb{E}(n_{i,1,0,d}) + \lambda_{b_i,d}]}, \quad (3)$$

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

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

c_i is not estimated since the setting of its prior counts will make it almost 0.

4.3 Estimating Location Popularity

Similarly, we can estimate the location popularity as

$$\hat{g}_j = \frac{\mathbb{E}(n_{j,1}) + \lambda_{g_j,1}}{\sum_d [\mathbb{E}(n_{j,d}) + \lambda_{g_j,d}]}, \quad (4)$$

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.

These estimated model parameters can be used to select participants and to compute $p(\mathbf{z}|\mathbf{X})$ in future tasks.

5. EXPERIMENTAL SETUP

In this section, we describe the truth discovery methods compared, the evaluation metrics used and the experiments conducted.

5.1 Methods in Comparison

We term our proposed method as Truth finder for Spatial Events (`TSE`) and compare it with the following state-of-the-art methods: 1) `MV`: the widely applied Majority Voting method, which predicts the event to be true if the proportion of positive reports exceeds 0.5; 2) `TF`: the Truth Finder proposed in [28], 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 [27] for online crowdsourced image classification (the authors' code is used); 4) `LTM`: the Latent Truth Model proposed in [29] for conflicting web information; 5) `EM`: the Expectation and Maximization method proposed in [24] for social sensing. Except our proposed `TSE`, other methods do not model location popularity, location visit indicators and three-way participant reliability. In dealing with missing data, they either ignore them or treat them as negative (for binary truth discovery). As has been discussed, the former treatment will result in consistent information and will lead to a trivial conclusion that every event is true, we thus use the latter treatment for all these compared methods.

Table 2: Statistics of reports in Area 1 (the left half) and Area 2 (the right half) in Figure 4a.

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

We also compare the performance of `TSE` in estimating participants' location visit indicators and location popularity with two baseline methods, where `Naive0` simply assumes $h_{i,j} = 0$ for $x_{i,j} = 0$ (all the missing reports are due to mobility issues) and `Naive1` simply assumes $h_{i,j} = 1$ for $x_{i,j} = 0$ (each participant visited all event locations). Actually, `Naive0` and `Naive1` act equivalently as ignoring missing reports and treating them as negative reports respectively.

We set the hyperparameters in `TSE` as follows. We set $\lambda_{c_i,1} = 2$ and $\lambda_{c_i,0} = 10^4$, since by domain knowledge we have $\lambda_{c_i,1} \ll \lambda_{c_i,0}$. For the FPR b_i, we set $\lambda_{b_i,1} = 5$ and $\lambda_{b_i,0} = 40$. This is to prevent the model from flipping the inference while still achieving a high likelihood. The setting of $\lambda_{g_j,1}$ and $\lambda_{g_j,0}$ will be explained in each experiment. We set all other hyperparameters to 5. These hyperparameters can be set much larger for large datasets or set to different values if prior domain knowledge is available.

5.2 Metrics

We use the following metrics to evaluate the performance of these methods.

1) Precision, recall and F1 score: $pre = \frac{TP}{TP+FP}$, $rec = \frac{TP}{TP+FN}$, $F1 = 2\frac{pre \times rec}{pre+rec}$, where TP represents the number of true positives (an algorithm infers an event is true when it is indeed true). We use them to evaluate the performance of the estimation $\hat{z}_j$ on event labels. The higher these metrics, the better a method performs.

2) Mean absolute error (MAE): $mae(a) = \frac{1}{N}\sum_{i=1}^{N} |\hat{a}_i - a_i|$. We use it to evaluate the performance of the estimations $\hat{h}_{i,j}$ on location visit indicators, $\hat{g}_j$ on location popularity, $\hat{a}_i$ on participants' TPRs and $\hat{b}_i$ on participants' FPRs. The lower this metric, the better a method performs. Ground truth of g_j, a_i and b_i are calculated based on their definitions (Table 1). $mae(h)$ is calculated only for missing reports $x_{i,j} = 0$ whose $h_{i,j}$ need to be estimated.

5.3 Experiments

We conduct three sets of experiments to evaluate the performance of the compared methods. The first set is to detect the locations of traffic lights, where reports were "made" by participants driving vehicles when they were waiting at a location for a certain time. The second set is to detect image-based events. It is created by combining a mobility dataset and three online image-based event detection datasets to examine the influence of mobility in addition to participants' bias. The third set is to further examine the methods' performance when participants with specific kinds of reliability (such as conservative) are present through simulations. We describe their details below.

5.3.1 *Traffic Light Detection*

Dataset: We use the mobility dataset (denoted as $\mathcal{M}$) provided in [20] as our experiment dataset. It contains time-stamped GPS location traces for 536 taxicabs in San Francisco with successive location updates recorded 1-60 seconds apart. We choose a region shown in Figure 4a 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 area into two subareas of equal size, where participants more densely visited the right half (Area 2) than the left half (Area 1).

Task: The crowdsourcing task that we consider is to detect the locations of traffic lights. We observe that vehicles usually wait at traffic light locations for a few seconds to a few minutes. Therefore, 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 such as a careless driver may pass stop signs without stopping and 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 respectively, collectively identifying 2,051 and 2,683 distinct 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 [7] with a cutoff distance of $80m$. This procedure ensures that the traffic light reports are attributed to the correct intersections and it results in 486 and 537 distinct cluster centers in Area 1 and 2 respectively. We then randomly pick out 100 of them in each area to annotate the ground truth using the Street View in Google Maps.

Data analysis: We define a participant's location coverage as $|l(u_i)|/|l|$, where $|l(u_i)|$ and $|l|$ denote the number of event locations visited by u_i and the total number of event locations. A location's popularity can be expressed as $|u(l_j)|/|u|$, where $|u(l_j)|$ and $|u|$ denote the number of participants that visited l_j and the total number of participants. Figure 5 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 respectively. Only around 20% and around 40% of event locations have a popularity of over 0.8 in Area 1 and 2 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. Figure 6 plots example positive reports for true and false events. We can observe that a majority of true events and false events receive a similar number of reports. Majority voting can easily fail in such an environment since the number of reports for true events can seldom exceed half of the number of participants.

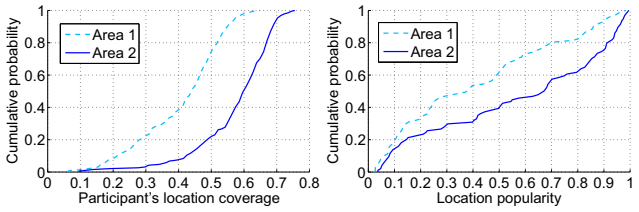


Figure 5: Participants' location coverage ($|l(u_i)|/|l|$) and locations' popularity ($|u(l_j)|/|u|$).

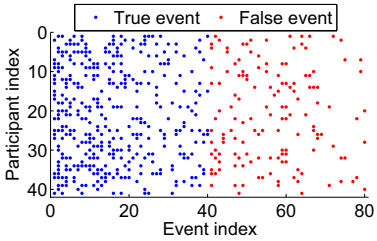


Figure 6: Positive reports for true and false events.

5.3.2 *Image-based Event Detection*

We combine the mobility dataset $\mathcal{M}$ and three online crowdsourced event detection datasets to evaluate the performance of the methods in a broader range of applications. This set of experiments is to mimic the scenario that participants move outdoors, observe events and make reports based on their judgments. The combination procedure described below is essentially to capture the physical constraint that a participant could make a report at a location only when she visited that location. It assumes nothing about mobility patterns, event labels and participant reliability.

We created three image-based event detection tasks on CrowdFlower [2] with clear instructions. They were to detect 1) bike racks, 2) Chinese restaurants and 3) flowering cherries respectively. For each task, $M = 20$ crowd workers were recruited and each of them was asked to report the images with target events among $N = 40$ images, where half of them contain the target event and half do not. We chose images with different viewing angles and distances, and also carefully selected a portion of them which may cause false alarms or missed detections. To introduce mobility, we randomly assigned each participant u_i a GPS trace from $\mathcal{M}$. We then randomly sampled N locations l_j from all these assigned GPS traces to represent the event locations for the N images. Only if u_i's GPS trace showed that she visited l_j at some time and u_i reported that the jth image contained the target event, we generated a positive report $x_{i,j} = 1$. We term these combined dataset corresponding to the three tasks as $\mathcal{B}_\mathcal{M}$, $\mathcal{C}_\mathcal{M}$ and $\mathcal{P}_\mathcal{M}$ respectively.

5.3.3 *Simulation Study*

In the simulation study, we put our focus on evaluating the performance of different methods in the presence of specific types of participants (defined in Table 3). We are particularly interested in the following representative ones: 1) reliable participants with a large TPR a and a small FPR b, 2) unskilled participants with a and b close to 0.5, reporting almost randomly, 3) conservative participants with a small a and a small b, reporting occasionally only when they are very confident or willing to report, 4) aggressive participants with a large a and a large b, reporting over actively, and 5) malicious participants with a small a and a large b, flipping the labels most of the time. Of course, this categorization is rough. However, we can use it to gain some insights on the impact of participant reliability on the algorithm performance. We assume there are $M = 40$ participants and $N = 200$ event locations where half of them contain true events and half do not. We again randomly

Table 3: Categorization of participants according to their reliability parameters. a is the TPR and b is the FPR.

Category	a	b	Category	a	b
Reliable	[0.8, 1]	[0, 0.2]	Aggressive	[0.8, 1]	[0.6, 1]
Unskilled	[0.4, 0.6]	[0.4, 0.6]	Malicious	[0, 0.2]	[0.8, 1]
Conservative	[0, 0.2]	[0, 0.2]	Other	Other ranges	

Table 4: Precision, recall and F1 score on inferring event labels for traffic light detection.

	Area 1			Area 2		
	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**

assign GPS traces and sample event locations, but generate reports $x_{i,j}$ according to the process in Section 3.1.5 with $c_i = 0$.

6. EXPERIMENTAL RESULTS

In this section, we demonstrate the effectiveness of our proposed method compared with several top-performing truth discovery approaches on both real-world and synthetic datasets.

6.1 Traffic Light Detection

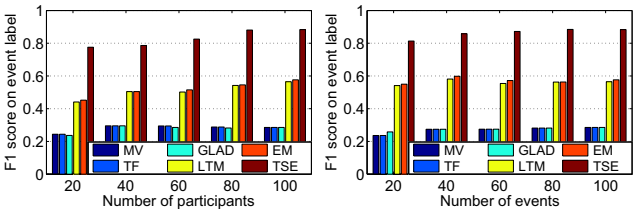
We report results based on 20 runs of tests for each experiment by randomly sampling the corresponding number of participants and events unless all of them are used. Only participants and events with at least two reports are kept for evaluation. We utilize a disjoint set of participants to estimate the prior counts of location popularity and set these counts fixed for all the experiments. To reduce noise, we consider a participant visited an event location if she had traversed there at least twice.

6.1.1 Estimation of Event Labels

Table 4 lists the precisions, recalls and F1 scores of all the methods on the two datasets from Area 1 and Area 2 when $M = 100$ and $N = 100$. We can observe that TSE achieves the highest recall and F1 score as well as very high precision on both datasets, showing that it can better handle missing reports and more accurately infer the truth of events in crowdsourced detection of spatial events. 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.

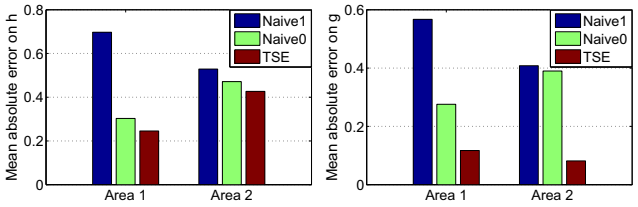
MV performs badly, since it does not take into account the participant reliability in its prediction. TF contains a mechanism to assign implication scores between similar observations. However, there is no similar but only contradictory observations for binary events. The power of TF is thus lost and it also performs badly. GLAD models only a single correct rate for participant reliability and thus it is not suitable for crowdsourced event detection. The overfitting problem makes it perform similar to MV. LTM and EM perform comparably and much better than MV, TF and GLAD, since they model two-sided participant reliability. However, they are not designed to tackle the mobility issues and thus fail to detect lots of positive events, resulting in low recalls.

Figure 7 plots the F1 scores on estimating the event labels versus the number of participants and the number of events respectively in Area 2 (the trends in Area 1 are similar). We can observe that TSE



(a) F1 versus M (b) F1 versus N

Figure 7: F1 scores 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.



(a) $mae(h)$ (b) $mae(g)$

Figure 8: MAEs on h and g for traffic light detection.

performs much better than all the other methods in all the cases, and it can benefit from more available information to improve its performance.

6.1.2 Estimation of Location Visit Indicators and Location Popularity

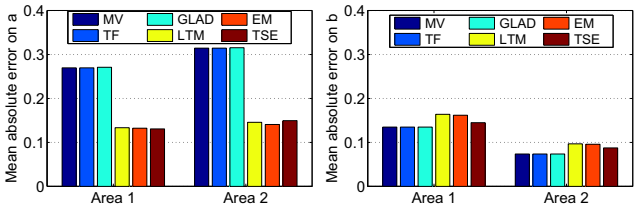
Figure 8 plots $mae(h)$ and $mae(g)$ when $M = 100$ and $N = 100$ via TSE and two baseline methods. Naive1 results in the largest $mae(h)$ in both areas. This shows that mobility issues lead to lots of missing reports and simply treating them as implicit negative reports is incorrect. TSE performs better than Naive0, showing that it indeed correctly infers some location visit indicators. However, TSE is not significantly better than Naive0 (e.g., $mae(h)$ in Area 1 are 0.246 and 0.303 for TSE and Naive0 respectively), meaning that TSE still faces difficulty in reliably inferring a large proportion of location visit indicators. As to $mae(g)$, Naive1 still results in the largest error but TSE results in a significantly lower error than Naive0 (e.g., $mae(g)$ in Area 1 are 0.118 and 0.276 for TSE and Naive0 respectively), showing that TSE can more reliably infer location popularity g_j. This is because we only statistically model $h_{i,j} \sim \text{Bernoulli}(g_j)$, so that $h_{i,j}$ can have different realizations given the same g_j. We may more accurately discover true events when $h_{i,j}$ can be more accurately inferred. However, if privacy is an important concern, accurately inferring location popularity g_j in a collective sense but less accurately inferring individual location visit indicators $h_{i,j}$ may be more desirable.

6.1.3 Estimation of Participant Reliability

Figure 9 plots the MAEs on estimating participants' TPRs and FPRs. We can observe that MV, TF and GLAD are biased towards more accurately inferring FPRs b. This is because they are prone to predict most events to be false and participants mostly do not report, leading to small FPRs. Since in reality, the FPRs are usually small, these methods accidentally achieve good performance. However, they perform poorly in estimating TPRs a. On the other hand, LTM, EM and TSE achieve much better and more balanced performance in estimating a and b.

6.2 Image-based Event Detection

We use uniform prior counts for location popularity. Table 5 lists the precision, recall and F1 scores of all the methods on the three



(a) $mae(a)$ (b) $mae(b)$

Figure 9: MAEs on a and b for traffic light detection.

Table 5: Precision, recall and F1 score on inferring event labels for image-based event detection.

	Bike rack $\mathcal{B}_\mathcal{M}$			Restaurant $\mathcal{C}_\mathcal{M}$			Plant $\mathcal{P}_\mathcal{M}$		
	pre	rec	$F1$	pre	rec	$F1$	pre	rec	$F1$
MV	**1.000**	0.468	0.638	0.836	0.468	0.600	0.908	0.488	0.602
TF	**1.000**	0.493	0.661	0.836	0.468	0.600	0.908	0.488	0.602
GLAD	**1.000**	0.545	0.706	**0.954**	0.463	0.623	**1.000**	0.510	0.675
LTM	0.954	0.630	0.758	0.921	0.569	0.704	0.983	0.550	0.677
EM	0.955	0.636	0.763	0.917	0.568	0.702	0.973	0.583	0.685
TSE	0.963	**0.874**	**0.916**	0.916	**0.708**	**0.799**	0.865	**0.671**	**0.756**

combined datasets when $M = 20$ and $N = 40$. We can observe similar characteristics as those in Table 4. TSE achieves the highest recalls and F1 scores on all the datasets, followed by LTM and EM. GLAD easily leads to overfitting, and MV and TF performs worst. However, their performance is better than that in Table 4, possibly due to different sizes of the datasets and different natures of tasks.

For the MAEs on estimating h and g, we also observe that TSE performs much better than the two naive methods and it is more capable of accurately estimating location popularity than participants' location visit indicators (figures are not shown due to space limitations). Since each participant only reports a few events, it is unreliable to directly calculate TPRs and FPRs from these datasets as ground truth and we thus do not consider the estimation of them here.

6.3 Simulation Study

We report results based on 20 independent runs for each experiment where uniform prior counts for location popularity are used. Figure 10 shows the F1 scores on event labels and the MAEs on participant reliability under different combinations of reliable and other types of participants. We can observe that TSE achieves the highest F1 scores across all different scenarios, and it also results in smallest MAEs on a and b in most scenarios. LTM and EM are prone to generate large errors on estimating the TPR a, since treating missing reports as negative will reduce the TPR. MV and TF perform poorly, easily resulting in low F1 scores on estimating event labels or large errors on estimating participant reliability. Conservative participants have the highest impact on the performance of truth discovery among all the compared participant categories.

7. DISCUSSION

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 [12]. One possible solution is to apply copy detection methods between sources [9]. Alternatively, we can directly incorporate source dependency in the modeling [21].

Sequential mobility modeling. Our current method models location popularity in a collective sense. Alternatively, we can also model the most likely trajectory for each participant. This may improve the accuracy of the estimated location visit indicators and subsequently improve the accuracy of other estimates. However, it will increase the model complexity and impose higher privacy risks. Moreover, this approach may not work well if the time interval between consecutive reports is large (e.g., hours) [13].

Cross-domain truth discovery. Our experiments show that event detection tasks in some domains are intrinsically more difficult than those in other domains. Difficult tasks and limited number of reports can deteriorate the performance of truth discovery. Leveraging tasks in different domains may help since it can balance the varied difficulties and increase the number of reports. To achieve this goal, we may need to leverage transfer learning techniques [18].

8. RELATED WORK

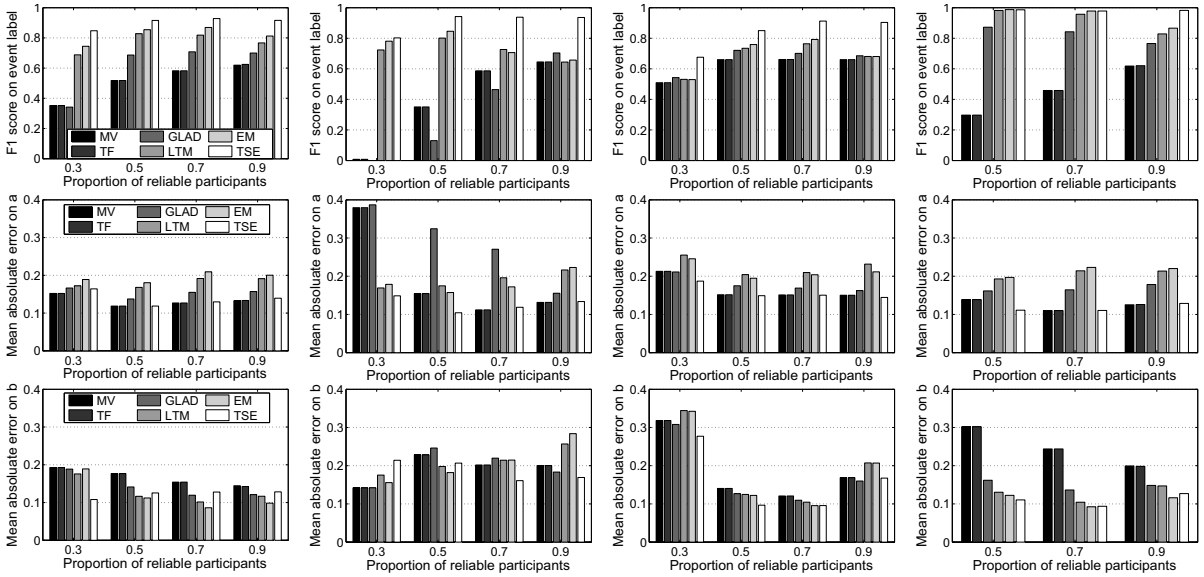
A number of unsupervised approaches have been proposed for discovering the truth from conflicting information sources. In the domain of truth discovery from conflicting Web information, Yin et al. [28] proposed truth finder, which is a transitive voting algorithm with rules specifying how votes iteratively 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 [10] which was initially designed to find popular web pages. Pasternack and Roth [19] 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. Unlike these heuristics, Zhao et al. [29] proposed a more principled probabilistic approach which can automatically infer true claims and two-sided source quality.

In the domain of aggregating conflicting responses in crowdsourcing tasks, several statistical techniques have been proposed. To name a few, Dawid and Skene [8] modeled the generative process of the responses by introducing worker ability parameters. Whitehill et al. [27] also included the difficulty of the task in the model, and Welinder et al. [26] proposed a model consisting of worker compatibility for each task. Wang et al. [24, 25] proposed an EM algorithm that models both the truth of tasks and the reliability of workers for social sensing.

Nevertheless, these methods are not designed to tackle truth discovery in mobile crowdsourced event detection where both participants' mobility and reliability are uncertain. Moreover, none of them models location popularity, location visit indicators and three-way participant reliability. Alternative solutions, such as first continuously tracking participants' locations and then applying existing truth discovery methods, will raise severe privacy and energy issues. In contrast, our proposed model integrates mobility, reliability and latent truth in a unified framework and can jointly optimize all the model parameters without the need of location tracking.

9. CONCLUSION

In this paper, we have proposed a probabilistic graphical model to the problem of truth discovery in crowdsourced detection of spatial events. The proposed method integrates the modeling of location popularity, participants' location visit indicators, truth of events and three-way participant reliability in a unified framework. We demonstrate the accuracy and efficiency with which this method can handle ambiguous missing reports caused by either mobility or reliability issues, and automatically infer the truth of events and different aspects of participant reliability without any supervision or location tracking. Experimental results on real-world and synthetic datasets demonstrate that our proposed method outperforms existing state-of-the-art approaches for mobile crowdsourcing.



(a) Unskilled (b) Conservative (c) Aggressive (d) Malicious

Figure 10: Performance with the combination of reliable and (a) unskilled, (b) conservative, (c) aggressive and (d) malicious participants. The first row: F1 scores on estimating event labels z. The second row: MAEs on estimating TPRs a. The third row: MAEs on estimating FPRs b. $M = 40$ and $N = 200$.

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.

10. REFERENCES

[1] Amazon mechanical turk. `https://www.mturk.com/mturk/welcome`.

[2] Crowdflower. `https://crowdflower.com/`.

[3] Field agent. `http://www.fieldagent.net`.

[4] Gigwalk. `http://gigwalk.com`.

[5] Taskrabbit. `http://www.taskrabbit.com`.

[6] A. R. Beresford and F. Stajano. Location privacy in pervasive computing. *Pervasive Computing, IEEE*, 2(1):46–55, 2003.

[7] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*. Springer New York, 2006.

[8] A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied Statistics*, pages 20–28, 1979.

[9] X. L. Dong et al. Truth discovery and copying detection in a dynamic world. *VLDB Endowment*, 2(1):562–573, 2009.

[10] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[11] 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.

[12] 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.

[13] Y. Lou et al. Map-matching for low-sampling-rate gps trajectories. In *GIS*, pages 352–361. ACM, 2009.

[14] P. Mohan et al. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In *SenSys*, pages 323–336. ACM, 2008.

[15] M. Musthag and D. Ganesan. Labor dynamics in a mobile micro-task market. In *CHI*, pages 641–650. ACM, 2013.

[16] R. W. Ouyang et al. Energy efficient assisted gps measurement and path reconstruction for people tracking. In *GLOBECOM*, pages 1–5. IEEE, 2010.

[17] R. W. Ouyang et al. If you see something, swipe towards it: crowdsourced event localization using smartphones. In *UbiComp*, pages 23–32. ACM, 2013.

[18] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE TKDE*, 22(10):1345–1359, 2010.

[19] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). In *COLING*, pages 877–885. Association for Computational Linguistics, 2010.

[20] M. Piórkowski et al. A parsimonious model of mobile partitioned networks with clustering. In *COMSNETS*, pages 1–10. IEEE, 2009.

[21] G.-J. Qi et al. Mining collective intelligence in diverse groups. In *WWW*, pages 1041–1052. ACM, 2013.

[22] V. C. Raykar et al. Learning from crowds. *The Journal of Machine Learning Research*, 99:1297–1322, 2010.

[23] S. Reddy et al. Recruitment framework for participatory sensing data collections. In *Pervasive Computing*, pages 138–155. Springer, 2010.

[24] D. Wang et al. On truth discovery in social sensing: a maximum likelihood estimation approach. In *IPSN*, pages 233–244. ACM, 2012.

[25] D. Wang et al. On credibility estimation tradeoffs in assured social sensing. *IEEE JSAC*, 31(6):1026–1037, 2013.

[26] P. Welinder et al. The multidimensional wisdom of crowds. In *NIPS*, pages 2424–2432, 2010.

[27] J. Whitehill et al. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *NIPS*, pages 2035–2043, 2009.

[28] 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.

[29] B. Zhao et al. A bayesian approach to discovering truth from conflicting sources for data integration. *VLDB Endowment*, 5(6):550–561, 2012.

[30] Z. Zhuang et al. Improving energy efficiency of location sensing on smartphones. In *MobiSys*, pages 315–330. ACM, 2010.