Recursive Fact-finding: A Streaming Approach to Truth Estimation in Crowdsourcing Applications

Dong Wang, Tarek Abdelzaher
Department of Computer Science
University of Illinois
Urbana, IL 61801
dwang24@illinois.edu, zaher@cs.illinois.edu

Lance Kaplan
Networked Sensing & Fusion Branch
US Army Research Laboratory
Adelphi, MD 20783
lance.m.kaplan@us.army.mil

Charu C. Aggarwal
IBM Research
Yorktown Heights, NY
charu@us.ibm.com

***Abstract*—This paper presents a *streaming* approach to solve the truth estimation problem in crowdsourcing applications. We consider a category of crowdsourcing applications where a group of individuals volunteer (or are recruited to) share certain observations or measurements about the physical world. Examples include reporting locations of gas stations that remain operational after a natural disaster or reporting locations of potholes on city streets. We call such applications *social sensing*. Ascertaining the correctness of reported observations is a key challenge in such applications, referred to as the *truth estimation* problem. This problem is made difficult by the fact that the reliability of individual sources is usually unknown *a priori*, since any concerned citizen may, in principle, participate. Moreover, the timescales of crowdsourcing campaigns of interest can be as small as a few hours or days, which does not offer enough history for a reputation system to converge. Instead, recent prior work, including our own, developed fact-finding algorithms to solve this problem by iteratively assessing the credibility of sources and their claims in the absence of reputation scores. Such algorithms, however, operate on the entire dataset of reported observations in a batch fashion, which makes them less suited to applications where new observations arrive continually. In this paper, we describe a *streaming* fact-finder that recursively updates previous estimates based on new data. The recursive algorithm solves an expectation maximization (EM) problem to determine the odds of correctness of different observations. We compare the performance of our recursive EM algorithm to a batch EM algorithm, as well as to several state-of-art fact-finders through extensive simulations. We also demonstrate convergence of the recursive algorithm to the results of the batch version through a real social sensing experiment. Our evaluation shows that the proposed approach can process data streams much more efficiently while keeping the truth estimation accuracy close to that of the (much slower) batch algorithm. Ours is therefore the first fact-finder developed with explicit consideration to the continuous update needs of crowd-sourcing applications.**

***Index Terms*—real-time, truth discovery, recursive expectation maximization, streaming data, social sensing**

I. Introduction

This paper presents a recursive fact-finding solution to the truth estimation problem in social sensing. We refer by social sensing to a broad set of crowdsourcing applications, where individuals volunteer or are recruited to collect data about the physical environment. For example, they may report events of mutual interest or download a cell-phone application to perform specific sensor data collection and sharing tasks. Due to the potentially unreliable nature of such unvetted human sources (and the potential problems with their sensors, if used), a key challenge in social sensing applications is to *assess the likelihood of correctness of reported data*. We call it the *truth estimation* problem.

Reputation systems [16] have been successful at assessing quality of *providers* (e.g., the reliability of data sources) when the same providers repeatedly execute transactions that can be scored by others. In contrast to such scenarios, we are specifically interested in short-lived crowdsourcing campaigns (e.g., to support post-disaster recovery and rescue missions, which may last for only a few days), where anyone can volunteer and where there is not enough history to accumulate meaningful reputations. For example, consider the recent severe gas shortage around New York City in the aftermath of hurricane Sandy. Social networks, such as Twitter carried tens of thousands of tweets on the availability of gas at different stations, but the reliability of the corresponding tweeters remained unknown.

Fact-finder algorithms [25], [27], [36] have been proposed that use unsupervised machine learning techniques to assess data reliability directly from multitudes of unreliable claims, whose sources may not have a known history in advance. The problem was also explored in data mining literature [12], [18], [37], with intuitions tracing back to Google's original PageRank [3], [23]. These solutions iteratively rank claims and sources to jointly assess the reliability of both, without requiring sources to explicitly comment on each other's performance. Unfortunately, they use batch algorithms, designed to run on a static dataset. As such, they are not well-suited to processing streaming data for applications such as crowdsourcing, where new observations continue to arrive over time. The batch algorithms will either need to operate on a growing data set as new data arrive (which does not scale), or ignore some previously computed results and run from scratch on a sliding recent data window (which does not exploit all available data).

In contrast, the main contribution of this paper is to develop a *recursive* fact-finder, based on expectation maximization (EM) that operates on new data only, as it arrives, updating previous truth estimates (i.e., estimates of correctness of reported data) in a manner that approximates running an optimal batch algorithm on the entire augmented dataset. To the best of our knowledge, the streaming EM scheme proposed in this pa-

per is the first *on-line* fact-finding approach designed to solve the truth estimation problem in social sensing applications, where there is no prior knowledge on source reliability and no immediate way to verify the correctness of the collected data. The streaming EM scheme is derived by formulating an optimization problem (in the sense of maximum-likelihood estimation) and approximating the optimal solution using results from estimation theory.

In our evaluation, we study the performance of the new recursive EM scheme by comparing it to a previously-proposed batch EM-based fact-finder [36] and several other state-of-art fact-finders [18], [25], [37] through extensive simulations. The recursive algorithm is shown to have a better performance trade-off between estimation accuracy and algorithm execution time than all baselines. We also evaluate the performance of the recursive EM scheme through a real social sensing application. The results demonstrate convergence of the recursive algorithm in quality to results of the corresponding (optimal but much slower) batch EM algorithm if run on the entire data set. The results of this paper are important because they allow social sensing applications to estimate data quality and participant reliability from *streaming data* on the fly, even in short-lived crowd-sourcing campaigns with no prior information on participants.

Finally, it is pertinent to note components that fall outside the scope of this work. First, we restrict this work to improving the data processing algorithm on the back-end. The mechanisms used on the front-end for data collection from participants are outside scope. For example, a cell-phone application might be used to report participants' observations. We also do not address security as part of this work and do not claim the system to be attack-proof. Instead, we simply contend ourselves with assuming that mechanisms are in place to increase the cost of identity, sybil and other attacks, and that the volunteer participants in our applications (e.g., post-disaster rescue) are generally well-meaning and have no incentive to disrupt operation. For example, phone companies already keep track of identities of individual phones (e.g., for billing purposes), which we can leverage to identify unique sources. Finally, we assume that campaign participants operate individually. Hence, to a first degree of approximation, reports from different sources may be considered conditionally independent.

With these caveats, the rest of the paper is organized as follows: We briefly go over the model of the truth estimation problem in Section II and propose the recursive EM algorithm in Section III. Evaluation results are presented in Section IV. We then review the related work in Section V and conclude the paper in Section VII.

II. Truth Estimation in Social Sensing

Social sensing addresses the challenge of estimating some pertinent "state of the world" from reports by human sources. In this paper, we model the state of the world by a set of true/false statements (e.g., "The Golden Gate bridge is on fire", "The 435 Main Street gas station is out of power", or "The 5th Avenue and 34th Street intersection is flooded"). Such a binary approach, while simple, is a powerful tool to articulate arbitrarily complex conditions. It is also well-suited to geotagging campaigns that mark locations of some conditions of interest (e.g., locations of street flooding after a thunderstrom). For example, each location may be associated with a number of Booleans indicating the presence or absence of different types of damage. A report from a source conveys one or more claims, each presenting the value of one of these Booleans. The "ground truth" state is unknown and needs to be reconstructed as accurately as possible from claims by different sources, whose reliability is unknown.

More formally, consider a social sensing application model, where a group of M participants (sources), $S_1, ..., S_M$, collectively make observations about N measured Boolean variables, $C_1, ..., C_N$, which are of interest to the application. We assume, without loss of generality, that the "normal" state of each (Boolean) variable is negative (e.g., a place is not damaged). Hence, participants only report when the positive state of the measured variable (repair is needed) is encountered. Each source generally reports only a subset of the variables (e.g., those at the places they have been to). The goal of truth estimation in social sensing is to jointly calculate the reliability of participants (i.e., the probability that a participant reports correct observations) and the correctness of observations, given only who reported what.

Importantly, in crowdsourcing applications, the observations from participants don't come all at once. Instead, updates are reported over the course of the campaign, lending themselves better to the abstraction of a *data stream* arriving from the community of sources. In our previous work, we developed a batch EM (expectation maximization) algorithm to solve the truth estimation problem based on a maximum likelihood estimation hypothesis [36]. As its name suggests, the batch EM scheme is designed to run in a batch mode, which is not suitable for continuously arriving data. This is because, every time a new report arrives, the batch EM algorithm needs to be re-run on the whole data set from scratch. Considering such inefficiency, this paper designs a new fact-finding approach based on a recursive EM algorithm to update estimation results on the fly in view of newly arriving data.

Following the terminology of previous work [33]–[36], let us define a few notations we will use in the following sections. Let S_i denote the i^{th} source and C_j denote the j^{th} measured variable. Let $X_{i,j}$ denote whether source S_i reports measured variable C_j. The matrix representing who reported what is called the observation matrix X, where $X_{i,j} = 1$ when source S_i reports that C_j is true, and $X_{i,j} = 0$ otherwise. Let T_j represent the ground truth value of C_j (i.e., T_j is 1 if C_j is true and 0 otherwise). Participant reliability t_i is defined as the probability that the participant is right in a randomly chosen measured variable he/she reported. Formally, t_i is defined as :

$$t_i = P(T_j = 1 | X_{i,j} = 1) \tag{1}$$

Let us also define two more important conditional probabilities: a_i is the (unknown) probability that source S_i reports

a variable to be true when it is indeed true, and b_i is the (unknown) probability that source S_i reports a variable to be true when it is in reality false. Formally, a_i and b_i are defined as follows:

$$a_i = P(X_{i,j} = 1|T_j = 1) \qquad b_i = P(X_{i,j} = 1|T_j = 0) \quad (2)$$

The relationship between t_i, a_i and b_i can be derived by the Bayes' theorem:

$$a_i = \frac{t_i \times s_i}{d} \qquad b_i = \frac{(1 - t_i) \times s_i}{1 - d} \quad (3)$$

where d is the overall background prior that a randomly chosen measured variable is true. Note that, this value does not indicate, however, whether any particular report about a specific measured variable is true or not. d can be either chosen from the prior knowledge or jointly estimated in the EM scheme [36]. Finally, s_i denotes the probability that participant S_i reports an observation.

Starting with a log-likelihood function that describes the likelihood of the observed data (i.e., who said what) given the estimation parameter defined in Equation (2), the batch EM algorithm converges to the maximum likelihood estimate of the variables in question (in this case, the truth values of measured variables and the reliability of sources). The likelihood function can be given by:

$$L = \prod_{j=1}^{N} \Bigg\{ \prod_{i=1}^{M} a_i^{X_{i,j}} (1 - a_i)^{(1-X_{i,j})} \times d \times z_j + \prod_{i=1}^{M} b_i^{X_{i,j}} (1 - b_i)^{(1-X_{i,j})} \times (1 - d) \times (1 - z_j) \Bigg\} \quad (4)$$

where, N and M are the numbers of measured variables and sources, respectively, z_j is 1 if measured variable C_j is true (and 0 otherwise). The optimal estimation of the parameters in the batch EM algorithm [36] are given by:

$$a_i^* = \frac{\sum_{j \in SJ_i} Z_j}{\sum_{j=1}^{N} Z_j}$$
$$b_i^* = \frac{K_i - \sum_{j \in SJ_i} Z_j}{N - \sum_{j=1}^{N} Z_j} \quad (5)$$

where SJ_i is the set of measured variables the participant S_i actually observes and K_i is its size. Z_j is the probability of C_j to be true given current estimation and observed data.

In this paper, we design a new streaming fact-finder based on a recursive EM algorithm to accurately estimate the above parameters from streaming data.

III. A Recursive Fact-finder

In the following subsections, we derive a recursive formula for our fact-finder (in Section III-A) then summarize the final algorithm (in Section III-B).

A. The Derivation

In estimation theory, a recursive formula of the EM scheme estimates parameters of the model in consecutive time intervals as follows [30]:

$$\hat{\theta}_{k+1} = \hat{\theta_k} + \{(k+1) I_c(\hat{\theta_k})\}^{-1} \psi(X_{k+1}, \hat{\theta_k}) \quad (6)$$

where $\hat{\theta_k}$ is the estimation parameter by observing the data up to the time interval k, $I_c^{-1}(\hat{\theta_k})$ represents the inverse of the Fisher information (i.e., Cramer Rao lower bound (CRLB)) of the estimation parameter at time k and $\psi(X_{k+1}, \hat{\theta_k})$ is the score vector of the observed data at time interval $k+1$ w.r.t the estimation parameter $\hat{\theta_k}$. This formula basically provides us a recursive way to compute the estimation parameter in the new time interval (i.e., $\hat{\theta}_{k+1}$) based on its estimation value in the previous time interval (i.e., $\hat{\theta_k}$), the CRLB of the estimation (i.e., $I_c^{-1}(\hat{\theta_k})$) and the score vector of the updated data observed in the new interval (i.e., $\psi(X_{k+1}, \hat{\theta_k})$). Based on our previous results of the EM scheme, $\hat{\theta_k}$ is the estimation vector defined as $\hat{\theta_k} = (\hat{a}_1^k, \hat{a}_2^k, ...\hat{a}_M^k; \hat{b}_1^k, \hat{b}_2^k, ...\hat{b}_M^k)$. $I_c^{-1}(\hat{\theta_k})$ and $\psi(X_{k+1}, \hat{\theta_k})$ are given by [35]:

$$I_c^{-1}(\hat{\theta_k})_{i,j} = \begin{cases} 0 & i \neq j \\ \frac{\hat{a}_i^k \times (1 - \hat{a}_i^k)}{N \times d} & i = j \in [1, M] \\ \frac{\hat{b}_i^k \times (1 - \hat{b}_i^k)}{N \times (1-d)} & i = j \in (M, 2M] \end{cases} \quad (7)$$

and

$$\psi(X_{k+1}, \hat{\theta_k})_{i,j} = \begin{cases} 0 & i \neq j \\ \sum_{j=1}^{N} \hat{Z_j}^{k+1} (\frac{X_{i,j}}{\hat{a}_i^k} - \frac{1 - X_{i,j}}{1 - \hat{a}_i^k}) & i = j \in [1, M] \\ \sum_{j=1}^{N} (1 - \hat{Z_j}^{k+1}) (\frac{X_{i,j}}{\hat{b}_i^k} - \frac{1 - X_{i,j}}{1 - \hat{b}_i^k}) & i = j \in (M, 2M] \end{cases} \quad (8)$$

where $\hat{Z_j}^{k+1}$ is the probability of the j^{th} measured variable to be true in the $k+1$ time interval. Plugging Equation (7) and (8) into (6), the recursive formula to update the estimation parameters is given by:

$$\hat{a_i}^{k+1} = \hat{a_i}^k + \frac{1}{Nd(k+1)} \times \Big[\sum_{j \in SJ_i^{k+1}} \hat{Z_j}^{k+1} (1 - \hat{a_i}^k) - \sum_{j \in S\bar{J}_i^{k+1}} \hat{Z_j}^{k+1} \hat{a_i}^k \Big]$$
$$\hat{b_i}^{k+1} = \hat{b_i}^k + \frac{1}{Nd(k+1)} \times \Big[\sum_{j \in SJ_i^{k+1}} (1 - \hat{Z_j}^{k+1}) (1 - \hat{b_i}^k) - \sum_{j \in S\bar{J}_i^{k+1}} (1 - \hat{Z_j}^{k+1}) \hat{b_i}^k \Big] \quad (9)$$

From above equations, we observe that the estimation of the parameters related with reliability of each source in current time interval can be computed from their estimations in the past and the observed data in the new interval. Moreover, $\hat{Z_j}^{k+1}$ is unknown and can be estimated by its approximation $\tilde{Z_j}^{k+1}$, which can be computed as follows:

$$\tilde{Z_j}^{k+1} = f(\tilde{a_i}^{k+1}, \tilde{b_i}^{k+1}, X_{k+1}) = \frac{A_j^{k+1} \times d}{A_j^{k+1} \times d + B_j^{k+1} \times (1-d)}$$

where

$$A_j^{k+1} = \prod_{i=1}^{M} (\tilde{a_i}^{(k+1)})^{X_{i,j}{}^{k+1}} (1 - \tilde{a_i}^{(k+1)})^{(1-X_{i,j}^{k+1})}$$

$$B_j^{k+1} = \prod_{i=1}^{M} (\tilde{b_i}^{(k+1)})^{X_{i,j}{}^{k+1}} (1 - \tilde{b_i}^{(k+1)})^{(1-X_{i,j}^{k+1})}$$

$$\tilde{a_i}^{k+1} = \hat{a_i}^{k} \times \frac{s_i^{k+1}}{s_i^k} \quad \tilde{b_i}^{k+1} = \hat{b_i}^{k} \times \frac{s_i^{k+1}}{s_i^k} \tag{10}$$

where s_i^{k+1} and s_i^k are the probabilities of source S_i to report a measured variable at time interval $k+1$ and k. For the above equation to hold, we assume source reliability changes slowly over time and can be treated unchanged over two consecutive time intervals.

Based on the definition of $\tilde{Z_j}^{k+1}$, we can further represent it as a function of $\hat{a_i}^k, \hat{b_i}^k, X_k, X_{k+1}$, the values of which we know at time interval $k+1$:

$$\tilde{Z_j}^{k+1} = g(\hat{a_i}^k, \hat{b_i}^k, X_k, X_{k+1}) = \frac{C_j^{k+1} \times d}{C_j^{k+1} \times d + D_j^{k+1} \times (1-d)}$$

where

$$C_j^{k+1} = \prod_{i=1}^{M} (\hat{a_i}^k \times \frac{s_i^{k+1}}{s_i^k})^{X_{i,j}{}^{k+1}} (1 - \hat{a_i}^k \times \frac{s_i^{k+1}}{s_i^k})^{(1-X_{i,j}^{k+1})}$$

$$D_j^{k+1} = \prod_{i=1}^{M} (\hat{b_i}^k \times \frac{s_i^{k+1}}{s_i^k})^{X_{i,j}{}^{k+1}} (1 - \hat{b_i}^k \times \frac{s_i^{k+1}}{s_i^k})^{(1-X_{i,j}^{k+1})} \tag{11}$$

Plugging Equation (11) into Equation (9), we can get the following recursive computation of the estimation parameters:

$$\hat{a_i}^{k+1} = \hat{a_i}^k + \frac{1}{Nd(k+1)} \times \Big[\sum_{j \in SJ_i^{k+1}} g(\hat{a_i}^k, \hat{b_i}^k, X_k, X_{k+1})(1 - \hat{a_i}^k) - \sum_{j \in S\bar{J}_i{}^{k+1}} g(\hat{a_i}^k, \hat{b_i}^k, X_k, X_{k+1}) \hat{a_i}^k \Big]$$

$$\hat{b_i}^{k+1} = \hat{b_i}^k + \frac{1}{Nd(k+1)} \times \Big[\sum_{j \in SJ_i^{k+1}} (1 - g(\hat{a_i}^k, \hat{b_i}^k, X_k, X_{k+1}))(1 - \hat{b_i}^k) - \sum_{j \in S\bar{J}_i{}^{k+1}} (1 - g(\hat{a_i}^k, \hat{b_i}^k, X_k, X_{k+1})) \hat{b_i}^k \Big] \tag{12}$$

Additionally, we can also compute the updated correctness of measured variables (i.e, $\hat{Z_j}^{k+1}$) as follows:

$$\hat{Z_j}^{k+1} = f(\hat{a_i}^{k+1}, \hat{b_i}^{k+1}, X_{k+1}) \tag{13}$$

where function f is the same as the one in Equation (10).

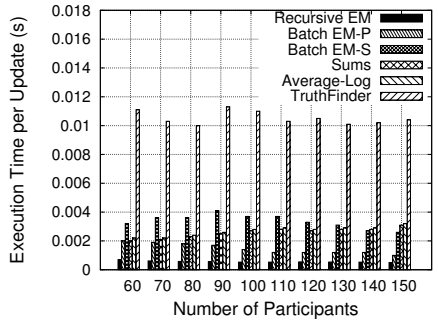
This gives us the recursive equations to compute the estimation parameters of our model in the current time interval based on the estimations from the previous time interval and the observed data up to now. Therefore, we can utilize (12) to keep track of the estimation parameter of the sources that report new observations consecutively over time. We also note that the estimation parameter change of the updated sources will affect the credibility of measured variables they report, which in turn will affect the credibility of other sources asserting the same measured variable. We call this credibility update propagation "ripple effect". To capture such an effect, we do a simple trick: only run one EM iteration after applying the recursive formula (as compared to running the full version of EM from scratch). This turns out to be an efficient heuristic based on the following observations: i) the recursive estimation already offers us a reasonably good initialization on the estimation parameter; ii) the credibility change of sources by a few updates in a short time interval is usually slight. This allows the recursive EM to converge much faster than the batch algorithm that starts from a random point.

B. The Final Algorithm

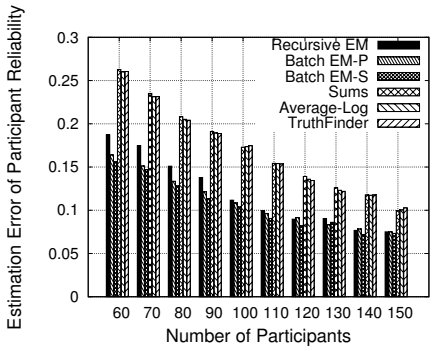
Algorithm 1 Recursive Expectation Maximization Algorithm

```
while new update X_{k+1} arrives do
    for i = 1 : M do
        compute â_i^{k+1}, b̂_i^{k+1} based on Equation (12)
        update â_i^k, b̂_i^k with â_i^{k+1}, b̂_i^{k+1}
    end for
    for j = 1 : N do
        compute Ẑ_j^{k+1} based on Equation (13)
    end for
    run one EM iteration to capture the "ripple effect"
    Let Z_j^r = the value of Ẑ_j^{k+1} after the iteration
    Let a_i^r = the value of â_i^{k+1} after the iteration
    Let b_i^r = the value of b̂_i^{k+1} after the iteration
    for j = 1 : N do
        if Z_j^r ≥ 0.5 then
            C_j is true
        else
            C_j is false
        end if
    end for
    for i = 1 : M do
        calculate t_i^r from a_i^r, b_i^r based on Equation (3)
    end for
    k = k + 1
end while
```

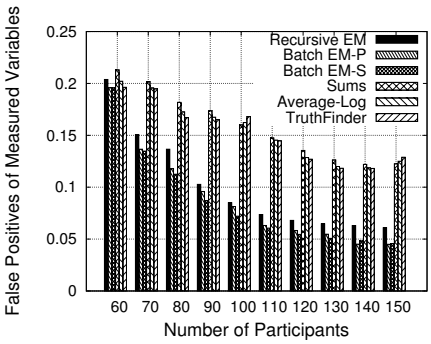
In summary of the recursive EM algorithm derived above, the psuedocode of the algorithm is given in Algorithm 1. The algorithm runs when a new update X_{k+1} arrives and it first computes the recursive update on the estimation parameter (i.e., $\hat{a_i}^{k+1}, \hat{b_i}^{k+1}$) based on Equation (12). The correctness of measured variables are consequently updated from the



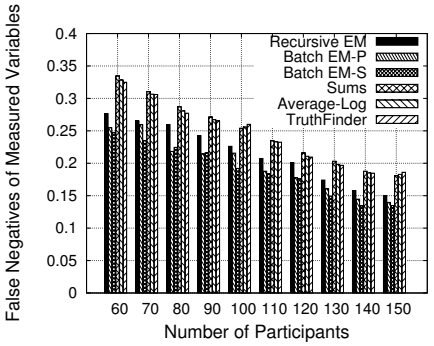
(a) Algorithm Execution Time



(b) Participant Reliability Estimation Accuracy



(c) Measured Variable Estimation: False Positives



(d) Measured Variable Estimation: False Negatives

Figure 1. Algorithm Performance versus Number of Participants

estimation parameters based on Equation (13). The recursive algorithm runs one EM iteration to capture the "ripple effect" of the credibility prorogation as we discussed in the previous subsection. After that, we decide the truthfulness of each measured variable C_j at current time slot based on the updated value of $\hat{Z_j}^k$ (i.e., Z_j^r). We can also compute the reliability of each source from the updated values of $\hat{a_i}^{k+1}, \hat{b_i}^{k+1}$ (i.e., a_i^r and b_i^r) based on Equation (3).

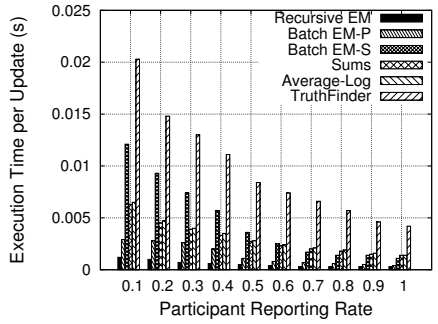
IV. Evaluation

In this section, we evaluate the performance of the proposed recursive EM algorithm compared to the batch EM algorithm and three state-of-art fact-finders; namely, Sums [18], Average-Log [25] and Truthfinder [37]. For the batch EM algorithm, there are two ways for parameter initialization: one way is to statically initialize the estimation parameters based on the observed data and run EM from scratch (denoted as batch EM-S) [36] and the other way is to use the values computed from the previous updates for the current initialization (denoted as EM-P). Below, We first evaluate estimation accuracy and algorithm execution time through an extensive simulation study. The recursive EM algorithm is shown to achieve a better performance tradeoff compared to the batch EM algorithm and other state-of-art baselines. Then, we empirically demonstrate convergence of the recursive EM algorithm to results of the (optimal but slower) batch EM algorithm through a real-world social sensing application.

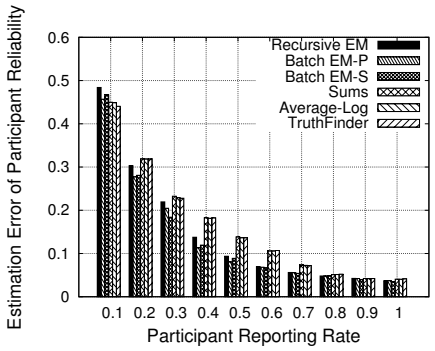
A. Simulation Study

We begin by evaluating the performance of the proposed recursive EM algorithm in simulation by measuring (i) the accuracy of participant reliability estimation, (ii) the false positive and false negative rates (i.e., claims misclassified as true or false), and (iii) the average time the algorithm takes to process an update in different conditions.

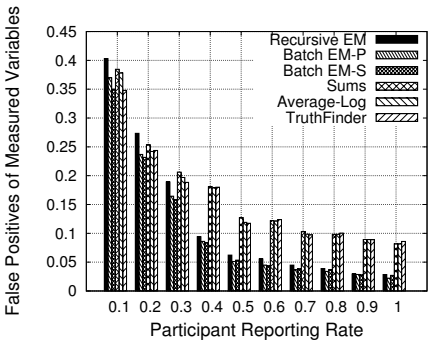
We built a simulator in Matlab 7.10.0 that generates a random number of participants and measured (Boolean) variables. A random probability P_i is assigned to each participant S_i representing his/her reliability (i.e., the ground truth probability that they report correct observations). A "reporting rate" of a source is defined as the probability that the source reports an observation at a given time slot, reflecting the source's willingness to report. At a given time slot, for each participant S_i, the simulator decides whether or not the participant reports an observation based on its reporting rate. Each reported observation from S_i has a probability t_i of being true (i.e., reporting the value of a variable correctly) and a probability $1-t_i$ of being false. We let t_i be uniformly distributed between



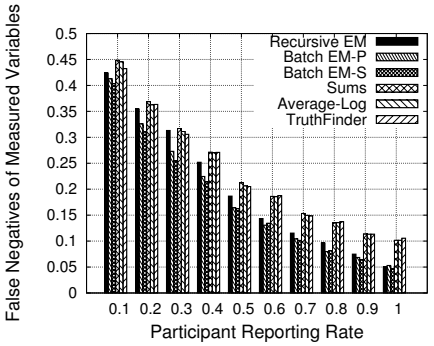
(a) Algorithm Execution Time



(b) Participant Reliability Estimation Accuracy



(c) Measured Variable Estimation: False Positives



(d) Measured Variable Estimation: False Negatives

Figure 2. Algorithm Performance versus Participant Chat Rate

0.5 and 1 in our experiments[1]. The fact-finder is executed as reports arrive to update estimates of participant reliability and truth values of reported data. Each point on the following curves is an average of 50 experiments.

In the first experiment, we evaluated the performance of recursive EM, the batch EM, and other baselines while varying the number of participants in the system. The total number of reported variables was set to 2000, half of which were reported correctly. The reporting rate of participants was fixed at 0.5. The number of participants was varied from 60 to 150. We simulated 100 time slots for the data stream generation. The observation updates of the last 20 slots were used to evaluate the algorithm performance. Reported results were averaged 50 experiments that differ in participant reliability distributions. Results are shown in Figure 1. Observe that the recursive EM algorithm takes the shortest time to process an update while keeping the estimation accuracy (in terms of both participant reliability estimation and measured variable classification) very close to the batch EM algorithm.

The second experiment compares the recursive EM to baseline algorithms when the source reporting rate changes from 0.1 to 1. Reported results are averaged over 50 experiments. The results are shown in Figure 2. We observe that the recursive EM algorithm continues to achieve a better trade-off between estimation accuracy and execution time: it runs fastest while offering comparable quality to the batch algorithm. Note also that both estimation accuracy and execution time of the studied algorithms improve as the source reporting rate increases. The reason is that a higher reporting rate leads to more data, which eventually allows faster convergence of the algorithm to a more accurate point.

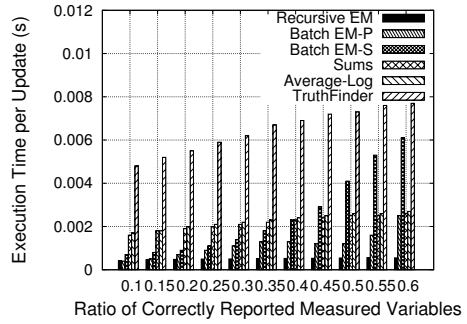
In the third and last experiment, we examine the effect of changing the measured variable mix on the performance of all algorithms. We fixed the total number of measured variables to be 2000 and vary the ratio of the number of correctly reported measured variables to the total number of reported variables from 0.1 to 0.6. The number of participants is set to 120 and source reporting rate is fixed at 0.5. Reported results are averaged over 50 experiments. The results are shown in Figure 3. As before, we observe that the recursive EM algorithm has the shortest execution time and does almost as well as the batch EM algorithm.

The simulation results show that the proposed recursive EM algorithm succeeds at offering similar estimation accuracy to its best batch counterpart while running significantly faster.

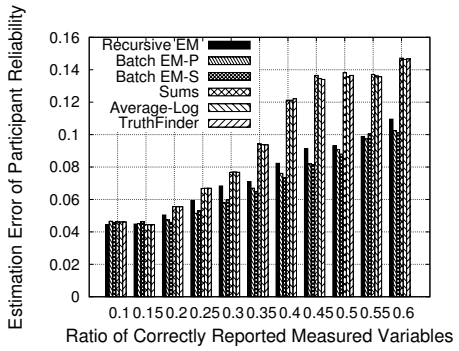
B. A Real World Case Study

In this section, we evaluate the performance of the proposed recursive EM algorithm compared to the batch EM algorithm

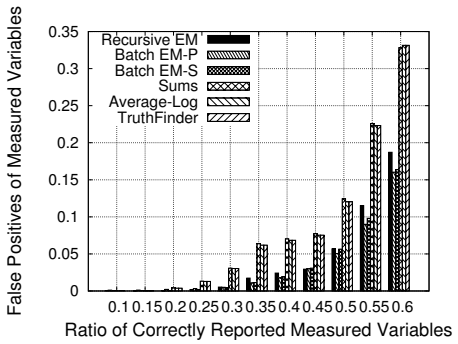
[1]In principle, there is no incentive for a participant to lie more than 50% of the time, since negating their statements would then give a more accurate truth



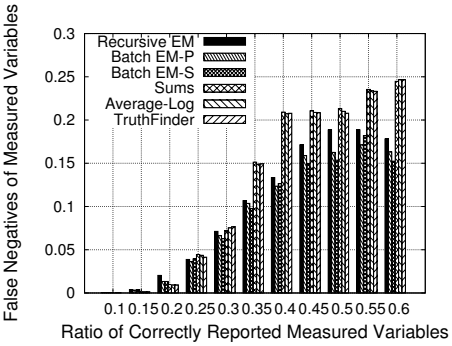
(a) Algorithm Execution Time



(b) Participant Reliability Estimation Accuracy



(c) Measured Variable Estimation: False Positives



(d) Measured Variable Estimation: False Negatives

Figure 3. Algorithm Performance versus Ratio of Correctly Reported Measured Variables

through a real world social sensing application. The application targets at finding the free parking lots on the campus of University of Illinois at Urbana-Champaign (UIUC). The "free parking lots" are defined as the parking lots that are free of charge after 5pm daily in this application. The goal here was to see if our recursive EM algorithm can track the performance of the batch EM algorithm and correctly find the real locations of free parking lot on campus. Specially, we selected 106 parking lots on campus and asked volunteers to mark the ones they believe as "Free". Participants marked those parking lots they have been to or are familiar with. We observe that various types of parking lots exist on campus: enforced parking lots with time limits, parking meters, permit parking, street parking, etc. Different parking lots have different regulations for free parking. Moreover, instructions and permit signs often read similar and easy to miss. Hence, participants are prone to make mistakes in their marks. For the purpose of evaluation, we went to those selected parking lots and manually collected the ground truth.

In the experiment, 30 participants were invited to provide their "free parking lot" marks on the 106 parking lots (46 of which are indeed free). There were 340 marks collected from participants in total. We then ran both the recursive and batch EM algorithms on the collected marks and compared their performance on identifying the correct free parking lots. Results are shown in Figure 4. We observe that the recursive EM algorithm is able to track the performance of the batch EM algorithm and converge to the number of free parking lots found by the batch algorithm as the amount of marks used by the algorithm increases. This result verified the nice convergence property of the developed recursive EM algorithm using real world data.

It should be emphasized that our choice of application is intended to be a proxy for other more pertinent uses of our fact-finding tool that are harder to experiment with in a paper (due to absence of ground truth). For example, "free parking lots" may stand for "operational gas stations" in a post-disaster scenario (such as the New York gas crisis in the aftermath of recent hurricane Sandy).

We should also highlight that we chose an application where ground truth *does not change*. This is intentional, in order to favor our competition (the batch algorithms) that operate on the entire data set at once and hence have difficulty handling dynamic changes. We expect the advantages of our recursive algorithm to be more pronounced if ground truth did change during the experiment (e.g., a gas station runs out of gas), since it is easy to adapt them to give more weight to more recent measurements. Due to space limitations, we do not include an evaluation of such more favorable scenarios to the recursive scheme.

Finally, we should note that we kept our data sets small enough such that running the batch algorithm upon every

update remained feasible (for evaluation purposes, where each point needs 50 runs). The real advantage of the recursive scheme, however, becomes clear when the input volume is scaled up. For example, hundreds of thousands of tweets may be received in the aftermath of real disaster events. Interpreting individual tweets as claims, a recursive fact-finder can rank the claims by credibility in real-time as events unfold, which would be much less time consuming than if a batch fact-finder is re-run continuously as new tweets arrive. Our prior work presents the results of applying batch fact-finders to Twitter data [36]; a painfully slow experience which motivated this work.

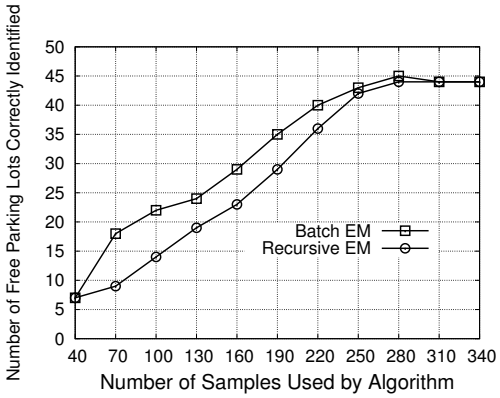


Figure 4. Recursive EM Algorithm Convergence

V. Related Work

Social sensing which is also referred to as human-centric sensing [4], [20], is generally achieved by various kinds of sensors which are closely attached to humans, either in their wearable form or in their mobile devices (e.g., cell phones). A broad overview of social sensing applications is presented in [1]. Some early applications include CenWits [14], CarTel [15], CabSense [29] and BikeNet [8]. More recent work has focused on addressing new challenges emerging in social sensing applications such as preserving privacy of participants [2], improving energy efficiency of sensing devices [24] and measuring the sociability of participants and strengthening their interactions [22], [28]. Examples include privacy-aware regression modeling, a data fusion technique that produce the same model as that computed from raw data by properly computing non-invertible aggregates of samples [2]. E-Gesture is an energy efficient gesture recognition architecture that significantly reduces the energy consumption of mobile sensing device while keeping the recognition accuracy acceptable [24]. SociableSense is a smart phones based platform used to measure the sociability of users and foster interactions among participants by studying their behavior in the office environment [28]. Nawaz et al. [22] adapted a similar social sensing system to understand group dynamics and information flow at building construction sites. Our work complements the past work by addressing the truth estimation in social sensing on the fly.

A relevant body of work in the machine learning and data mining communities performs trust analysis based on the source and claim information network. Fact-finders are a class of iterative trust analysis algorithms that estimate both the credibility of claims and the trustworthiness of the sources. Examples include Sums [18], TruthFinder [37], the Investment, PooledInvestment, Average·Log algorithms [25] and Bayesian Interpretation [32]. Many fact-finders also enhance the basic trust analysis models. 3-Estimates [10] rewards sources that correctly assert highly disputed claims, while AccuVote [7] considers "source dependence" by effectively boosting the trustworthiness of independent sources. More recent works came up with some new fact-finding algorithms designed to handle domain expertise of information sources, multi-valued facts of an entity and a subset of known ground truth of variables. Kasneci et al. [17] proposed a CoBayes scheme to learn the affinity between users' expertise and their statements by mapping them into a common latent knowledge space. Zhao et al. [40] presented a Bayesian scheme to model different types of errors made by sources and merge multi-valued attributes in data integration systems. Yin et al. [38] provided a semi-supervised approach to find the true values with the help of (a small amount of) ground truth data. In contrast, this paper proposed the first *on-line* fact-finder to solve the truth discovery problem in social sensing applications with explicit consideration to continuous data update.

Since people are an indispensable element in social sensing, some popular attacks originated from human (or source) interactions are interesting to investigate. Collusion attack is carried out by a group of colluded attackers who collectively perform some malicious actions to defraud honest sources or obtain objective forbidden by the system. This attack could be mitigated by monitoring the interactions or relationships among colluded attackers or identifying the abnormal behavior from the group [19]. Sybil attack is another related attack carried out by a single attacker who intentionally create a large number of pseudonymous entities and use them to gain a disproportionately large influence on the system. This attack could be mitigated by certifying trust of identity assignment, increasing the cost of creating identities, limiting the resource the attacker can use to create new identities and etc. [39]. By handling reports from colluded or duplicated sources in a way that takes care of the source dependency, we will be able to address the above attacks to some extent. For example, by identifying duplicate sources, we can remove them along with their reports from the observed dataset, which is expected to improve the estimation performance. Problems become more interesting when sources are not just duplicates but actually linked through the social network [21].

Our work also bears resemblance to reputation systems. Different types of reputation systems are being used successfully in commercial online applications. For example, eBay is a type of reputation system based on homogeneous peer-to-peer systems, which allows peers to rate each other after transactions [13]. Our developed scheme may not be able to be directly applied to those systems. The reason is: the

structure of a homogeneous peer-to-peer system is commonly represented by a *mesh* network graph while the structure of our scheme is represented by a *bipartite* network graph (i.e., sources and measures are in disjoint sets). Amazon on-line review system represents another type of reputation systems, where different sources offer reviews on products (or brands, companies) they have experienced [16]. Customers are affected by those reviews (or reputation scores) in making purchase decisions. It turns out that our work fits better into this type of reputation systems and has the potential to provide more refined and timely results for the reputation computation.

The recursive expectation maximization (EM) algorithm is an online version of the EM algorithm where a statistical approximation procedure is applied to estimate the parameters in a recursive and adaptive way [30]. The recursive EM has been used in a wide range of applications with large dynamics in sensor networks. For example, Guo et al. developed a methodology based on recursive EM algorithm to optimize sensor deployment and adaptively estimate the boundary of sensor locations [11]. Frenkel et al. applied the recursive EM algorithm in a multiple target tracking scenario and achieved a linear computational complexity with respect to the target number in the system [9]. Chung et al. derived a recursive EM procedure for direction of arrival (DOA) estimation under a deterministic model and independent Gaussian noise [5]. In this paper, we proposed a recursive EM algorithm to greatly reduce the computation overload of our previous iterative algorithm which runs on a increasing dataset for streaming data [36]. To the best of our knowledge, this is the first *on-line* algorithm that is developed to address truth discovery challenge in social sensing applications.

VI. Limitations and Future Work

This paper presented a streaming fact-finding approach to address the truth estimation challenge in social sensing applications on the fly. Several simplifying assumptions were made that offer directions for future work.

Participants (sources) were assumed to be independent. However, in reality, sources might be non-independent or even collude to mask the truth. For example, in Twitter, it could be that a large set of individuals report the same observation not because they independently observed it themselves, but because they heard it from a source they trust (which could in fact be wrong). Several techniques have been recently developed to discover source dependencies and copying relationships [6], [7]. Other approaches were shown to efficiently mitigate the source collusion attack by analyzing the network or interaction patterns of colluding sources [19]. Additionally, source dependencies could also be inferred from the underlying social network. An admission controller was designed to select independent sources for social sensing applications based on a few distance metrics derived from the social network topology [31]. It is reasonable to integrate the above techniques with our streaming fact-finding approach to effectively handle source dependencies in the future.

In this paper, we did not assume dependencies between measured variables. However, observations on different variables may often be correlated. For example, a fire report at location A in a city might imply a traffic congestion at location B that is a few blocks away from A. Several approaches have been proposed to take the underlying relations between measured variables as prior knowledge [25], [26]. Hence, we can possibly extend the model of the EM scheme to incorporate dependencies into the likelihood function. Moreover, all observations are assumed to be equally important in our model. It is interesting to extend current model to consider the "difficulty" of making different observations, and giving more weight to correct reporting of more difficult obervations. A few techniques have been proposed to consider the hardness of observations, which may be used together with our scheme [10]. Additionally, the measured variable were assumed to be Boolean in this paper. This assumption is sufficient in many social sensing scenarios, where the existence or lack thereof a given condition of interest (e.g., litter) can be represented by the Boolean variable. Our model can be extended to handle other discrete measured variables (e.g., weather in a city can be sunny, rainy, or snowy) by expanding the number of estimation parameters to cover all possible states of the variable. The general outline of the derivation still holds. Having a basic streaming fact-finding algorithm in place, we shall relax the above assumptions and accommodate the mentioned extensions in future work.

VII. Conclusion

This paper described a streaming fact-finding approach to address the truth estimation problem in social sensing applications that allows applications to process streaming data efficiently. The streaming approach is developed based on a recursive EM algorithm that computes the estimation parameters by only processing the newly updated data and combining the results with previous estimates. The performance of the streaming fact-finder is evaluated through extensive simulations. Results show that a better trade-off between estimation accuracy and algorithm execution time has been archived by the new streaming approach compared to the batch EM scheme and other baselines. Evaluation data from a real social sensing application are also presented.

Acknowledgements

Research reported in this paper was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

REFERENCES

[1] T. Abdelzaher, Y. Anokwa, P. Boda, J. Burke, D. Estrin, L. Guibas, A. Kansal, S. Madden, and J. Reich. Mobiscopes for human spaces. *IEEE Pervasive Computing*, 6(2):20–29, 2007.

[2] H. Ahmadi, N. Pham, R. Ganti, T. Abdelzaher, S. Nath, and J. Han. Privacy-aware regression modeling of participatory sensing data. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, pages 99–112, New York, NY, USA, 2010. ACM.

[3] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *7th international conference on World Wide Web (WWW'07)*, pages 107–117, 1998.

[4] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn. The rise of people-centric sensing. *IEEE Internet Computing*, 12(4):12–21, July 2008.

[5] P.-J. Chung and J. Bohme. Recursive em and sage-inspired algorithms with application to doa estimation. *Signal Processing, IEEE Transactions on*, 53(8):2664 – 2677, aug. 2005.

[6] X. Dong, L. Berti-Equille, and D. Srivastava. Truth discovery and copying detection in a dynamic world. *VLDB*, 2(1):562–573, 2009.

[7] X. L. Dong, L. Berti-Equille, and D. Srivastava. Integrating conflicting data: the role of source dependence. *Proc. VLDB Endow.*, 2:550–561, August 2009.

[8] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell. The bikenet mobile sensing system for cyclist experience mapping. In *Proceedings of the 5th international conference on Embedded networked sensor systems*, SenSys '07, pages 87–101, New York, NY, USA, 2007. ACM.

[9] L. Frenkel and M. Feder. Recursive expectation-maximization (em) algorithms for time-varying parameters with applications to multiple target tracking. *Signal Processing, IEEE Transactions on*, 47(2):306 –320, feb 1999.

[10] A. Galland, S. Abiteboul, A. Marian, and P. Senellart. Corroborating information from disagreeing views. In *WSDM*, pages 131–140, 2010.

[11] Z. Guo, M. Zhou, and G. Jiang. Recursive online em algorithm for adaptive sensor deployment and boundary estimation in sensor networks. In *Networking, Sensing and Control, 2006. ICNSC '06. Proceedings of the 2006 IEEE International Conference on*, pages 862 –867, 0-0 2006.

[12] J. Han. Mining heterogeneous information networks by exploring the power of links. In *Proceedings of the 20th international conference on Algorithmic learning theory, ser. ALT'09. Berlin, Heidelberg: Springer-Verlag*, pages 3–3, 2009.

[13] D. Houser and J. Wooders. Reputation in auctions: Theory, and evidence from ebay. *Journal of Economics & Management Strategy*, 15(2):353–369, 2006.

[14] J.-H. Huang, S. Amjad, and S. Mishra. CenWits: a sensor-based loosely coupled search and rescue system using witnesses. In *SenSys'05*, pages 180–191, 2005.

[15] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden. Cartel: a distributed mobile sensor computing system. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, SenSys '06, pages 125–138, New York, NY, USA, 2006. ACM.

[16] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2):618–644, Mar. 2007.

[17] G. Kasneci, J. V. Gael, D. Stern, and T. Graepel. Cobayes: bayesian knowledge corroboration with assessors of unknown areas of expertise. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 465–474, New York, NY, USA, 2011. ACM.

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

[19] Q. Lian, Z. Zhang, M. Yang, B. Y. Zhao, Y. Dai, and X. Li. An empirical study of collusion behavior in the maze p2p file-sharing system. In *Proceedings of the 27th International Conference on Distributed Computing Systems*, ICDCS '07, pages 56–, Washington, DC, USA, 2007. IEEE Computer Society.

[20] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, SenSys '08, pages 337–350, New York, NY, USA, 2008. ACM.

[21] A. Mohaisen, N. Hopper, and Y. Kim. Keep your friends close: Incorporating trust into social network-based sybil defenses. In *INFOCOM, 2011 Proceedings IEEE*, pages 1943 –1951, april 2011.

[22] S. Nawaz, C. Efstratiou, C. Mascolo, and K. Soga. Social sensing in the field: challenges in detecting social interactions in construction sites. In *Proceedings of the 1st ACM workshop on Mobile systems for computational social science*, MCSS '12, pages 28–32, New York, NY, USA, 2012. ACM.

[23] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[24] T. Park, J. Lee, I. Hwang, C. Yoo, L. Nachman, and J. Song. E-gesture: a collaborative architecture for energy-efficient gesture recognition with hand-worn sensor and mobile devices. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, SenSys '11, pages 260–273, New York, NY, USA, 2011. ACM.

[25] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). In *International Conference on Computational Linguistics (COLING)*, 2010.

[26] J. Pasternack and D. Roth. Generalized fact-finding (poster paper). In *World Wide Web Conference (WWW'11)*, 2011.

[27] J. Pasternack and D. Roth. Making better informed trust decisions with generalized fact-finding. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, pages 2324–2329. AAAI Press, 2011.

[28] K. K. Rachuri, C. Mascolo, M. Musolesi, and P. J. Rentfrow. Sociablesense: exploring the trade-offs of adaptive sampling and computation offloading for social sensing. In *Proceedings of the 17th annual international conference on Mobile computing and networking*, MobiCom '11, pages 73–84, New York, NY, USA, 2011. ACM.

[29] Sensor Network. Cab Sense. http://www.cabsense.com.

[30] D. M. Titterington. Recursive parameter estimation using incomplete data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 46(2):pp. 257–267, 1984.

[31] M. Uddin, M. Amin, H. Le, T. Abdelzaher, B. Szymanski, and T. Nguyen. On diversifying source selection in social sensing. In *Networked Sensing Systems (INSS), 2012 Ninth International Conference on*, pages 1 –8, june 2012.

[32] D. Wang, T. Abdelzaher, H. Ahmadi, J. Pasternack, D. Roth, M. Gupta, J. Han, O. Fatemieh, and H. Le. On bayesian interpretation of fact-finding in information networks. In *14th International Conference on Information Fusion (Fusion 2011)*, 2011.

[33] D. Wang, T. Abdelzaher, L. Kaplan, and C. C. Aggarwal. On quantifying the accuracy of maximum likelihood estimation of participant reliability in social sensing. In *DMSN11: 8th International Workshop on Data Management for Sensor Networks*, August 2011.

[34] D. Wang, L. Kaplan, T. Abdelzaher, and C. C. Aggarwa. On credibility tradeoffs in assured social sensing. *IEEE Journal On Selected Areas in Communication (JSAC)*, 2013.

[35] D. Wang, L. Kaplan, T. Abdelzaher, and C. C. Aggarwal. On scalability and robustness limitations of real and asymptotic confidence bounds in social sensing. In *The 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 12)*, June 2012.

[36] D. Wang, L. Kaplan, H. Le, and T. Abdelzaher. On truth discovery in social sensing: A maximum likelihood estimation approach. In *The 11th ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN 12)*, April 2012.

[37] X. Yin, J. Han, and P. S. Yu. Truth discovery with multiple conflicting information providers on the web. *IEEE Trans. on Knowl. and Data Eng.*, 20:796–808, June 2008.

[38] X. Yin and W. Tan. Semi-supervised truth discovery. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 217–226, New York, NY, USA, 2011. ACM.

[39] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. Sybilguard: defending against sybil attacks via social networks. *SIGCOMM Comput. Commun. Rev.*, 36:267–278, August 2006.

[40] B. Zhao, B. I. P. Rubinstein, J. Gemmell, and J. Han. A bayesian approach to discovering truth from conflicting sources for data integration. *Proc. VLDB Endow.*, 5(6):550–561, Feb. 2012.