2016 IEEE 36th International Conference on Distributed Computing Systems

On Source Dependency Models for Reliable Social Sensing: Algorithms and Fundamental Error Bounds

Shuochao Yao*, Shaohan Hu*, Shen Li*, Yiran Zhao*, Lu Su†, Lance Kaplan‡, Aylin Yener§, Tarek Abdelzaher*

*University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
§Pennsylvania State University, University Park, PA 16802, USA
†State University of New York at Buffalo, Buffalo, NY 14260, USA
‡Army Research Labs, Adelphi, MD 20783, USA

***Abstract*—This paper develops a simplified dependency model for sources on social networks that is shown to improve the quality of fact-finding – assessing veracity of observations shared on social media. Recent literature developed a mathematical approach for exploiting social networks, such as Twitter, as noisy sensor networks that report observations on the state of the physical world. It was shown that the quality of state estimation from such noisy data, known as fact-finding, was a function of assumptions made regarding the independence of sources or lack thereof. When sources propagate information they hear from others (without verification), correlated errors may arise that degrade fact-finding performance. This work advances the state of the art by developing a simplified model of dependencies between sources and designing an improved dependency-aware estimator to assess veracity of observations, taking into account the observed dependency structure. A fundamental error bound is derived for this estimator to understand the gap in its performance from optimal. It is shown that the new estimator outperforms state of the art fact-finders and, in some cases, yields an accuracy close to the fundamental error bound.**

***Index Terms*—Social Sensing; Error Bound; Source Dependency; EM;**

I. Introduction

This paper contributes to social sensing literature by developing a source dependency model that leads to an improved fact-finder for social network data. We take Twitter as the social network of choice and design a new (source dependency-aware) fact-finding algorithm that is shown to outperform the state of the art [16] both in simulations and based on empirical Twitter data. The work is motivated by the proliferation of social networks and the wealth of information that is voluntarily broadcast on them, which generates interest in fact-finding algorithms that assess veracity of observations reported on social media.

The paper builds on recent work on social sensing. Early fact-finders used heuristic solutions inspired by data mining literature to iteratively assess veracity of sources and claims [15], [22]. More recently, estimation-theoretic models were developed that represent social networks as noisy sensor networks [16] leading to a generation of maximum-likelihood truth estimation approaches with well-understood analytic properties. In these models, sources (the sensors) generate statements that convey claims about the state of their environment, thereby committing acts of sensing. A particularly attractive model is one that treats these statements as true or false, leading to a simple, yet expressive binary sensor abstraction, where each statement is treated as an information "bit" that may be correct or not. Since data are noisy, the goal of the fact-finder is to identify those bits that are actually true in the physical world. The binary model is expressive because a bit can represent any arbitrary statement, such as "#BREAKINGNEWS Bomb threat prompts schoolwide evacuation at Mira Costa High School in Manhattan Beach http://abc7.la/1m74xAc," or "Students beginning to gather at 16th/Mission for walkout to protest SFPD killing of #MarioWoods" (actual tweets, each modeled as a binary claim). An estimation-theoretic solution was proposed where a maximum-likelihood estimator was designed to assess the probability of correctness of individual claims.

The paper advances this maximum-likelihood estimation approach, instead of competing alternatives [15], in view of its appealing analytic properties. Specifically, under this approach, it becomes possible not only to offer optimal (in the sense of maximum-likelihood) guesses of truth values of claims, but also compute the fundamental error bound on claim misclassifications (i.e., labeling correct claims as false and vice versa). This error bound is the first contribution of the paper.

Prior work demonstrated that a key factor that affects the quality of fact-finding, under the maximum-likelihood estimation approach, lies in the assumptions made regarding source dependencies. Early work assumed that sources are independent [18]. The resulting fact-finders favored claims supported by a larger number of sources, since the probability that all of them would be wrong deminishes quickly with increased support (under the independence assumption). Those fact-finders were thus prone to believing rumors in cases where all sources would repeat information they heard from someone they trusted without independent verification. Subsequent work used retweet behaviors and other indicators to empirically construct a dependency network among sources [16], where a link indicated that a source tends to repeat claims of another. The work assumed that claims repeated by dependent sources do not offer value from the perspective of truth estimation.

The second contribution of this paper lies in an improved

1063-6927/16 $31.00 © 2016 IEEE
DOI 10.1109/ICDCS.2016.75

(yet simple) model of source dependencies that offers a middle ground between assuming independence among sources and assuming that dependent sources offer no information. The model is shown to lead to an improved fact-finder. It stems from the intuition that, in reality, all sources fall somewhere in between the above two extremes. When a source repeats something that another source had previously shared on the social medium, the new observation is not necessarily an automatic repeat that carries no value. The source may or may not have verified the information first. We do not know which is the case, but that uncertainty can be added as another variable to estimate by the maximum-likelihood estimator. Hence, a more nuanced model is possible, leading to an extended estimator.

The novel estimation algorithm developed in this paper has been integrated into a previous fact-finding tool, called Apollo.[1] We carried out simulation experiments to evaluate our algorithm's performance against various existing baseline algorithms [15], [16], [18], [22]. In addition, we also conducted real-world empirical evaluation using new Twitter datasets that we collected on various topics/events from January through November 2015. From the results we observed that our new estimator consistently outperforms the state of the art [16].

The rest of this paper is organized as follows. Section II provides the outline of problem formulation. Section III describes our fundamental error bound on claim misclassifications. A practical estimator considering source dependencies is described in Section IV and evaluated in Section V. Section VI surveys related work. Section VII concludes.

II. Problem Formulation

To formulate the problem of dependency-aware social sensing, we first define the basic terminology, then intrduce our model of dependent sources that leads to the problem addressed in the paper.

A. Basic Terminology

Consider a set of n sources, $\mathcal{S} = \{S_1, S_2, \cdots, S_n\}$ who jointly report a set of m statements, we call assertions, $\mathcal{C} = \{C_1, C_2, \cdots, C_m\}$. As assertion could be *anything* that evaluates to true or false. For example, a tweet that says *"Gunshots, explosion as insurgents attack Spanish embassy in Kabul, Afghanistan http://s.rplr.co/ezyVQ3K"* could be viewed as an assertion, because it can be evaluated as true or false.[2] Our model associates a single binary variable with each assertion. That is true if the assertion is correct (i.e., the statement of the assertion is true in the physical world) and false otherwise. We assume that truth values of assertions are not known.

The same assertion may be reported by one or more sources. We call the act of "a source reporting an assertion", a *claim* made by that source. If a source S_i reports assertion C_j, we say that the source made claim S_iC_j, denoted by $S_iC_j = 1$. Otherwise, $S_iC_j = 0$, meaning that S_i did not make the claim. Note that, the set of all claims made can be represented by a matrix, SC, of dimensions $n \times m$, where element $SC[i, j] = S_iC_j$. We call it the *source-claim matrix*. We also denote $SC[:, j]$ as SC_j.

In general, a source, S_i, may see and be influenced by claims made by a subset of other sources (e.g., by following them on Twitter). We call those sources the *ancestors* of S_i. We say that a claim by S_i is *independent* if no ancestor of S_i made the same assertion before. Otherwise, the claim is called *dependent*. We use the indicator D_{ij} to denote dependent claims. We say $D_{ij} = 1$ if the claim (by source S_i, asserting C_j) is dependent. Otherwise, we say that $D_{ij} = 0$.

- *Example:* To illustrate our notations, consider the example in Figure 1. John (denoted by source S_1 in Figure 1) follows Sally (denoted by source S_2) on Twitter, but does not follow Heather (denoted by source S_3). The three are in the habit of reporting which streets they find congested during their commute (hopefully not while driving). On a particular morning, at time, t_1, Sally tweeted "Main Street, Urbana, IL is congested". Let us denote this assertion by C_1. We say that $S_2C_1 = 1$. At the same time, Heather tweeted "University Ave., Urbana, IL is congested". Let us denote this assertion by C_2. We say that $S_3C_2 = 1$. Later, at time t_2, John tweeted "Main Street, Urbana, IL is congested" then at time, t_3, he tweeted "University Ave., Urbana, IL is congested". They are denoted by $S_1C_1 = 1$ and $S_1C_2 = 1$, respectively. The remaining entries in the source-claim matrix are zero. According to our model, we consider the second of John's tweets to be independent because no person who John follows made the same assertion earlier. (Note that, John does *not* follow Heather.) Hence, $D_{1,2} = 0$. We also say that Sally's and Heather's tweets are independent, because none of their ancestors asserted the same. Hence, $D_{2,1} = 0$ and $D_{3,2} = 0$. However, the first of John's tweets is dependent according to our model because Sally, who John follows, made the same assertion at an earlier time. Hence, $D_{1,1} = 1$, as shown in the figure.

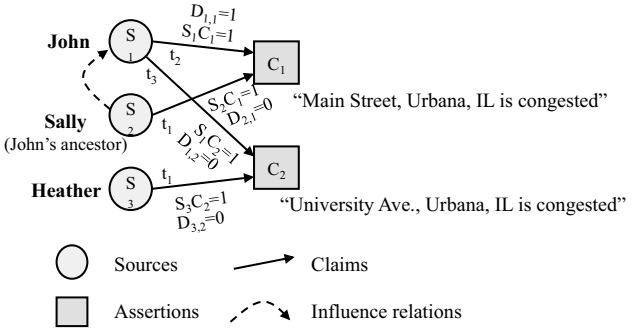


Fig. 1: An illustrative example.

B. A Model for Social Sources

With a slight abuse of notation, in the rest of this paper, we shall refer to an assertion and its truth value (true or false) by

[1] http://apollo3.cs.illinois.edu/

[2] For the sake of simplicity, in this paper, we consider partially true assertions as false.

the same variable. Hence, we say that $C_i = 1$ if assertion C_i is true, and $C_i = 0$ otherwise. We define the behavior of each source, S_i, by a model that defines four different probabilities that are *unknown* to the fact-finder:

- The probability of making independent claims that are true, denoted by $a_i = P(S_iC_j = 1|C_i = 1, D_{ij} = 0)$.
- The probability of making independent claims that are false, denoted by $b_i = P(S_iC_j = 1|C_j = 0, D_{ij} = 0)$.
- The probability of making dependent claims that are true, denoted by $f_i = P(S_iC_j = 1|C_j = 1, D_{ij} = 1)$.
- The probability of making dependent claims that are false, denoted by $g_i = P(S_iC_j = 1|C_j = 0, D_{ij} = 1)$.

In addition we denote $z = P(C = 1)$ as the probability of a general assertion C to be true.

We define the parameter set, θ_i, for each source S_i to be the set of unknowns $\{a_i, b_i, f_i, g_i\}$. The union of these sets over all sources and z constitutes the set of unknowns θ. Similarly, the collection of all dependency indicators, D_{ij}, is denoted by set, D. The goal is to estimate those unknowns in set θ together with the most likely truth value for each assertion, C_j, given the source claim matrix, SC, and the set of indicators, D.

In the next two sections, we provide an estimator error bound and a practical estimator based on the model and parameters we defined in this section.

III. Estimator Error Bound

Our goal is to estimate the truth value of individual assertions given the data received from the social network, and given the influence relations, D, determined (for example) by the network of retweets. We begin by computing a lower bound on *expected error* of the optimal estimator. By optimal, we mean that the estimator makes the best true/false judgement that is feasible given the data available to it.

To obtain a lower bound, we assume that the estimator determines all model parameters in the set θ *perfectly*. Remember that these parameters describe the probabilistic behavior of sources. Hence, any resulting error in assessing the true/false values of assertions is attributed solely to the inherent uncertainty resulting from the lack of source reliability in the given social network, as opposed to modeling error. Figure 2 shows an example, where sources who presumably witnessed an event report their observations. For simplicitly, the example shows only one assertion, C_j, reported by a subset of sources. In general, there may be many different assertions, some of which are true (they really happenned) and some are false (they did not).

Consider determining the veracity of assertion C_j by our hypothetical optimal estimator. There are only two possibilities for the value of ground truth: either C_j is true, or it is false. Given n sources, each of which either reports C_j or not, there are 2^n possible combinations of claims that may be observed (we will solve this exponential complexity problem in Section III-B). Let SC_j denote the set of actual claims observed. The optimal estimator with an exact knowledge of parameter set, θ, and dependency relations, D, will compare two values;

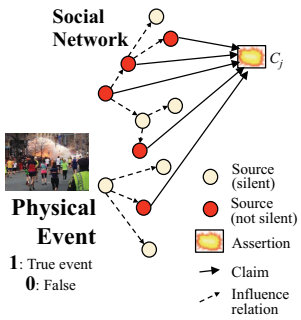


Fig. 2: An Example Assertion.

namely, $P(C_j = 1|SC_j, D, \theta)$ and $P(C_j = 0|SC_j, D, \theta)$, then decide on the truth value of C_j according to the higher of the two probabilities. The smaller of the two probabilities will then become the error probability in this case (i.e., the probability that ground truth differs from the estimate), given the set of received claims, SC_j, and the perfect knowledge of model parameters, θ, and dependencies, D. Let us denote this error by $P^{opt}(error|SC_j, D, \theta)$. Hence:

$$P^{opt}(error|SC_j)=\min\{P(C_j = 1|SC_j; D, \theta), P(C_j = 0|SC_j; D, \theta)\} \quad (1)$$

By definition of statistical expectation, the *expected* error probability of the optimal estimator, denoted $E^{opt}(error)$ is thus simply the weighted sum of the above probabilities (each weighted by the likelihood of observing the corresponding SC_j). Let the set of all 2^n possible combinations of received claims, SC_j, be denoted by A. Hence:

$$E^{opt}(error)=\sum_{SC_j \in A} \min\{P(C_j = 1|SC_j; D, \theta), P(C_j = 0|SC_j; D, \theta)\}P(SC_j|\theta) \quad (2)$$

We can now use the Bayesian rule to rewrite $P(C_j = 1|SC_j; D, \theta)$ and $P(C_j = 0|SC_j; D, \theta)$, used above, in terms of $P(SC_j|C_j = 1; D, \theta)$ and $P(SC_j|C_j = 0; D, \theta)$. After simplification, this yields:

$$E^{opt}(error)=\sum_{SC_j \in A} \min\{P(SC_j|C_j = 1; D, \theta)P(C_j = 1), P(SC_j|C_j = 0; D, \theta)P(C_j = 0)\} \quad (3)$$

In Equation (3), we have:

$$P(SC_j|C_j = 1; D, \theta)=\prod_i P(S_iC_j|C_j = 1; D_{ij}, \theta) \quad (4)$$

$$P(SC_j|C_j = 0; D, \theta)=\prod_i P(S_iC_j|C_j = 0; D_{ij}, \theta) \quad (5)$$

where, $P(S_iC_j|C_j = 1; D_{ij}, \theta)$, $P(S_iC_j|C_j = 0; D_{ij}, \theta)$, and $P(C_j)$ are explicit parameters in set θ, according to the model described in Section II-A. (Specifically, $P(S_iC_j|C_j = 1; D_{ij}, \theta)$ is a_i if $D_{ij} = 0$ and f_i if $D_{ij} = 1$. Similarly, $P(S_iC_j|C_j = 0, D_{ij}, \theta)$ is b_i if $D_{ij} = 0$ and g_i if $D_{ij} = 1$. And $P(C_j) = z$).

Since, in general, the parameter set θ may not be estimated exactly, the above constitutes a minimum *bound* on expected error of an optimal estimator. The approach used above is a fairly standard textbook technique, called Bayes risk. The contribution lies in expressing it in terms of the model parameters of the social channel.

A. A Walk-through Example

We use a simple example to illustrate the definition of the above bound. Suppose we have three sources, $\{S_1, S_2, S_3\}$. Table I lists the probabilities SC_j for each possible combination of received claim. The first column defines the combination. For example, "101" means that sources S_1 and S_3 reported C_j, but source S_2 did not. The second and third columns show the probabilities $P(SC_j|C_j = 1, D, \theta)$ and $P(SC_j|C_j = 0, D, \theta)$, respectively. We further assume, for simplicity, that $P(C_j = 1) = P(C_j = 0) = 0.5$.

TABLE I: Computing the Error Bound: An Example

SC_j	$P(SC_j\|C_j = 1, D, \theta)$	$P(SC_j\|C_j = 0, D, \theta)$
000	0.18546216	0.05851677
001	0.17606773	0.05300123
010	0.00033244	0.12803859
011	0.01971855	0.16032756
100	0.24427898	0.14231588
101	0.19063986	0.08222352
110	0.02321803	0.18716734
111	0.16028224	0.18840910

According to Equation (3), the error bound in this case should be:

$$\begin{aligned} Err=&0.5 \times (0.05851677 + 0.05300123 + 0.00033244 \\ &+0.01971855 + 0.14231588 + 0.08222352 \\ &+0.02321803 + 0.16028224) \\ =&0.26980433 \end{aligned}$$

Therefore for the given system model and parameters generating Table I, the expected error probability of any fact-finding algorithm is no less than 26.98%.

B. A Tractable Approximation

One obvious drawback of our error bound expression, derived above, is that it has exponential complexity in the number of sources. In order to make this computation scalable, a lower-complexity solution trading off accuracy against complexity is desired.

Our proposed error bound (3) can be regarded as a kind of marginal distribution, obtained by marginalizing over variable SC_j. Therefore well-studied marginal distribution approximation methods can be applied to make the error bound computing tractable.

Among all these choices [2], [3], Markov chain Monte Carlo methods are a class of algorithms for sampling from a probability distribution based on constructing a Markov chain that has the desired distribution as its equilibrium distribution [2], [7]. Therefore, we do not need to marginalize over all possible SC_j to provide the exact bound. It suffices to obtain a sufficiently large number, T, of samples to provide an approximate error bound instead.

In this paper, we use Gibbs sampling [5], [6] to approximate the marginal distribution $E^{opt}(error)$. Gibbs sampling obtains the samples based on the conditional probability of each variable. It generates an instance of variable $S_iC_j^{(t)} \in \{S_1C_j, \cdots, S_nC_j\}$ from the distribution of each variable in turn, conditioned on the current values of the other variables. In each sampling loop, Gibbs sampling provide a sample $\mathbf{s}^{(t)} = \{S_1C_j^{(t)}, \cdots, S_nC_j^{(t)}\}$ for marginal distribution estimation.

Gibbs sampling can provide a sequence of sampled observable claims efficiently. Therefore we can compute the approximated error bound by marginalizing over all samples $\mathbf{S} = \{\mathbf{s}^{(t)}\}$.

$$\begin{aligned} E^{opt}(error)\approx \Big(\sum_{\mathbf{s}^{(t)}\in\mathbf{S}} \min\{&P(SC_j^{(t)}|C_j = 1, D, \theta)P(C_j = 1), \\ &P(SC_j^{(t)}|C_j = 0, D, \theta)P(C_j = 0)\}\Big)/P(\mathbf{S}) \end{aligned} \tag{6}$$

where $P(\mathbf{S}) = \sum_{\mathbf{s}^{(t)}\in\mathbf{S}} P(SC_j^{(t)}|C_j = 1, D, \theta)P(C_j = 1) + P(SC_j^{(t)}|C_j = 0, D, \theta)P(C_j = 0)$.

The details of Gibbs sampling and approximate error bound computation method are given in Algorithm 1.

Algorithm 1 Approximate Error Bound

```
Initialize initial sample {S_1C_0, ..., S_nC_0}
t = 0, Err = 0.0, Total = 0.0, and ErrPart = 0.0
while Err not convergent do
    t ← t + 1
    for 1 ≤ i ≤ n do
        sample S_iC_j^(t) from conditional probability
        P(S_iC_j^(t) | S_1C_j^(t), ..., S_{i-1}C_j^(t), S_{i+1}C_j^(t-1)
            , ..., S_nC_j^(t-1); θ)
    end for
    if P(S_iC_j, C = 1; θ) > P(S_iC_j, C = 0; θ) then
        ErrPart ← ErrPart + P(S_iC_j, C = 0; θ)
    else
        ErrPart ← ErrPart + P(S_iC_j, C = 1; θ)
    end if
    Total ← Total + P(S_iC_j, C = 1; θ) + P(S_iC_j, C = 0; θ)
    Err = ErrPart/Total
end while
```

IV. A Practical Estimator

It remains to develop an estimator that solves our fact-finding problem with *no* knowledge of set θ. Such a fact-finder will need to estimate θ together with estimating the truth values of all assertions, C_j, given only the source claim matrix, SC, and the dependency indicators, D, as input. We model the problem posed above as a maximum likelihood estimation problem, where the log likelihood function is given by:

$$\begin{aligned} \mathcal{L}=&\ln\big(P(SC;\theta)\big) \\ =&\ln\left(\sum_{j=1}^{m}\sum_{C_j\in\{0,1\}} P(SC_j|C_j; D, \theta)P(C_j;\theta)\right) \end{aligned} \tag{7}$$

where $P(SC_j|C_j;D,\theta)$ is expressed in (4) (5).

The maximum likelihood estimator must find a solution that satisfies:

$$\arg\max_{\theta} \ln(\mathcal{L}) \tag{8}$$

The general way of solving maximum likelihood estimation problems with hidden variables is to use the expectation-maximization (EM) algorithm, composed of an expectation step and a maximization step. These steps are presented in the Appendix, where it is shown that they reduce to the following iterative computation:

$$P(C_j=1|SC_j;D,\theta)=\frac{P(SC_j|C_j=1;D,\theta)P(C_j=1;\theta)}{\sum_{C_j\in\{0,1\}}P(SC_j|C_j;D,\theta)P(C_j;\theta)} \tag{9}$$

where the values of $P(S_iC_j|C_j;\theta,D_{ij})$, under different parameter settings, are shown in Table II.

TABLE II: Values of $P(S_iC_j|C_j;\theta,D_{ij})$

C_j	D_{ij}	S_iC_j	$P(S_iC_j\|C_j;\theta,D_{ij})$
1	0	1	a_i
1	0	0	$1-a_i$
0	0	1	b_i
0	0	0	$1-b_i$
1	1	1	f_i
1	1	0	$1-f_i$
0	1	1	g_i
0	1	0	$1-g_i$

$P(SC_j|C_j;D,\theta)$ is calculated similar as (4) (5), and

$$a_i=\frac{\sum_{S_iC_j\in S_iC_1^{D_0}}P(C_j=1|S_iC_j;D,\theta)}{\sum_{S_iC_j\in S_iC_1^{D_0}\bigcup S_iC_0^{D_0}}P(C_j=1|S_iC_j;D,\theta)} \tag{10}$$

$$f_i=\frac{\sum_{S_iC_j\in S_iC_1^{D_1}}P(C_j=1|S_iC_j;D,\theta)}{\sum_{S_iC_j\in S_iC_1^{D_1}\bigcup S_iC_0^{D_1}}P(C_j=1|S_iC_j;D,\theta)} \tag{11}$$

$$b_i=\frac{\sum_{S_iC_j\in S_iC_1^{D_0}}P(C_j=0|S_iC_j;D,\theta)}{\sum_{S_iC_j\in S_iC_1^{D_0}\bigcup S_iC_0^{D_0}}P(C_j=0|S_iC_j;D,\theta)} \tag{12}$$

$$g_i=\frac{\sum_{S_iC_j\in S_iC_1^{D_1}}P(C_j=0|S_iC_j;D,\theta)}{\sum_{S_iC_j\in S_iC_1^{D_1}\bigcup S_iC_0^{D_1}}P(C_j=0|S_iC_j;D,\theta)} \tag{13}$$

$$z=\frac{\sum_{j=1}^{m}P(C_j=1|SC_j;D,\theta)}{m} \tag{14}$$

where $S_iC_1^{D_0}=\{S_iC_j:\forall S_iC_j\in SC\&S_iC_j=1\&D_{ij}=0\}$, $S_iC_1^{D_1}=\{S_iC_j:\forall S_iC_j\in SC\ \&\ S_iC_j=1\ \&\ D_{ij}=1\}$, $S_iC_0^{D_0}=\{S_iC_j:\forall S_iC_j\in SC\ \&\ S_iC_j=0\ \&\ D_{ij}=0\}$, and $S_iC_0^{D_1}=\{S_iC_j:\forall S_iC_j\in SC\ \&\ S_iC_j=0\ \&\ D_{ij}=1\}$. Denote that $SC_1^{D_0}\bigcup SC_1^{D_1}\bigcup SC_0^{D_0}\bigcup SC_0^{D_1}=SC$.

Hence, we jointly estimate the truth values of assertions and model parameters by iteratively solving Equation (9) and (10) - (14) until they converge. The convergence of EM algorithm is beyond the scope of this paper and well-studied in other work [20]. We summarize the estimator psuedocode in Algorithm 2.

V. Evaluation

In this section, we evaluate our proposed methods both via simulation experiments and through real-world datasets.

Algorithm 2 Joint Estimator

```
Initialize parameter set {θ^(t)} with random probability
while {θ} are not convergent do
    for j in xrange(m) do
        compute P(C_j = 1|SC_j; θ^(t)) according to Equ. (9)
    end for
    for i in xrange(n) do
        compute a_i, f_i(a_i, γ_T), b_i, and g_i(b_i, γ_F) according to
        Equ. (10), (11), (12), and (13) respectively.
    end for
    t+ = 1
end while
```

A. Simulation for Approximate Error Bound

We first generate synthetic data of fictional events. The synthetic data generator is parameterized to generate claims for n sources collectively making m different assertions.

In order to capture the different dependency characteristics of claims in a systematic way, we generate source dependency graph as a forest of τ level-two trees, where each source appears only once in this dependency graph. Therefore, source dependency ranges from a single source being followed by all other sources, to all sources stand independent.

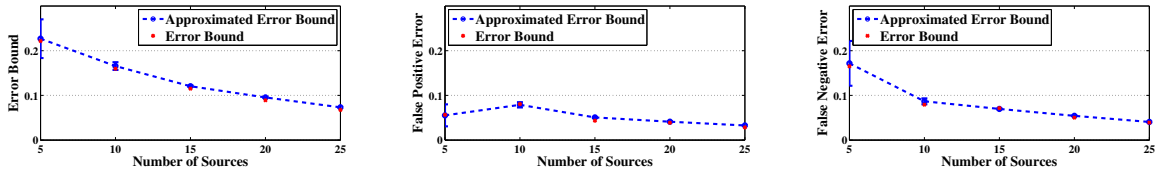
The total set of assertions $\{C_1,C_2,\cdots,C_m\}$ is divided into two pools, a "True Assertion" pool and a "False Assertion" pool, according to a value, d, that controls the ratio of true and false assertions. The total set of sources $\{S_1,S_2,\cdots,C_n\}$ is divided into two subsets, a "Root Sources" subset containing the independent nodes in the dependency graph, and "Leaf Sources" the dependent.

We first start with the "Root Sources". All sources in this set make independent claims. Whether source S_i makes an assertion or not is controlled by a probability p_i^{on}. When a source decides to participate in making the current assertion, a parameter p_i^{indepT} indicates whether S_i will make a true assertion, as opposed to a false one. The generator will then pick one assertion from the "True Assertion" or "False Assertion" accordingly.

For "Leaf Sources", whether source S_i makes an assertion is still controlled by the probability, p_i^{on}. For each leaf source, its candidate assertion set consists of a "Dependent Assertion" subset containing assertions that have already been made by its root, and an "Independent Assertion" subset containing assertions that have not been previously made by its root. These two subsets are controlled by two different probability parameters, p_i^{depT} and p_i^{indepT}, deciding whether S_i will make a true assertion or a false assertion. The generator will then pick one assertion from "True Assertion" or "False Assertion" from the corresponding subset accordingly.

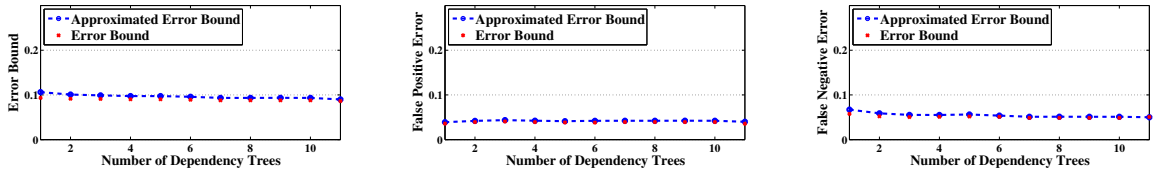
Hence, each source is personalized by a different degree of participation, p_i^{on}, and different reliabilities p_i^{depT} and p_i^{indepT}.

In order to evaluate the precision of our approximate error bound compared to the exact one, we run approximated and exact error bound algorithms for different model parameter settings. We take $n=20$, $m=50$, $p_i^{on}\in[0.5,0.7]$, $\tau\in[8,10]$, $p_i^{dep}\in[0.4,0.6]$, $d\in[0.55,0.75]$, $p_i^{indepT}\in[7/12,3/4]$,



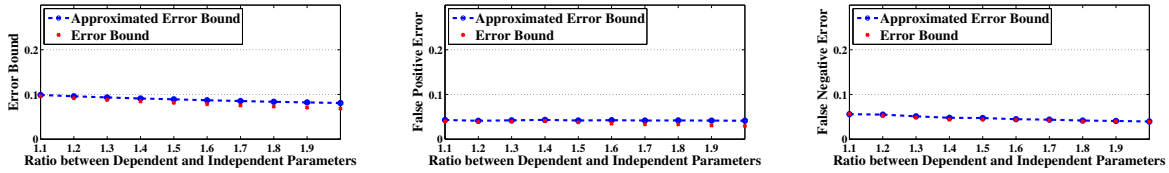
(a) Error Bound (b) False Positive Bound (c) False Negative Bound

Fig. 3: Bound with varying # of sources.



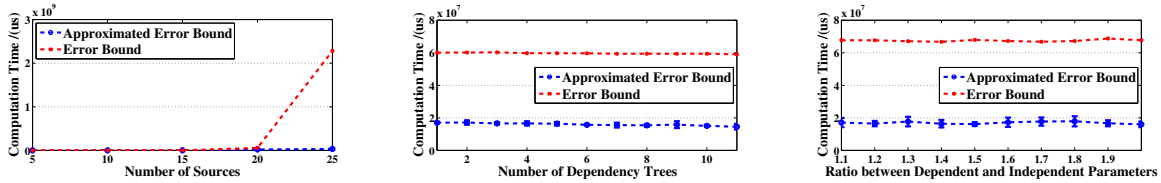
(a) Error Bound (b) False Positive Bound (c) False Negative Bound

Fig. 4: Bound with varying # of dependency trees.



(a) Error Bound (b) False Positive Bound (c) False Negative Bound

Fig. 5: Bound with varying varying source reliability.



(a) Varying # of sources (b) Varying # of dependency trees (c) Varying source reliability

Fig. 6: Bound Computation Time.

and $p_i^{depT} \in [0.4, 0.6]$ as default values except where specifically mentioned. Parameters with ranges are chosen uniformly within the range. We conduct 20 independent experiments for each simulation.

In following simulations, false positive bound and false negative bound represent the portion of error bound caused by regarding false assertions as true and true assertions as false respectively.

In the first simulation, we change the total number of sources, n, while keeping other parameters at default values. We change n from 5 to 25 in steps of 5. The precision of the approximate error bound compared with exact value is shown in Figure 3. The maximum difference between exact and approximated error bound is 0.0064 when $n = 20$.

In the second simulation, we vary the number of dependency trees from 1 to 11 in steps of 1, while keeping other parameters at their default values. From Figure 4, the maximum difference between exact and approximated error bound is 0.0127 when $\tau = 1$, indicating that approximated values are acceptable.

In the third simulation, we keep $p_i^{indepT}/(1-p_i^{indepT}) = 2$, and change $p_i^{depT}/(1 - p_i^{depT})$ from 1.1 to 2.0 with steps of 0.1, while keeping other parameters at default values. This parameter indicates how effectively one can discriminate true and false assertions, which will be fully discussed in the next subsection. From Figure 5, the differences between approximate and exact error bounds are still acceptable, where the maximum is 0.0116 at $p_i^{depT}/(1 - p_i^{depT}) = 2.0$.

Figure 6 shows that computing the approximate error bound is much faster than computing the exact error bound. The latter quickly becomes intractable as the number of sources n increases. Both computing times remain constant under different dependency structure and source reliability settings.

B. Simulation of Dependency-Aware Estimator

In order to systematically evaluate our proposed estimation method, we generate synthetic data with a wide range of parameters and compare the performance of the following four algorithms:

- EM-Ext: Our proposed estimator as described in this paper.
- EM (IPSN 2012): This algorithm jointly estimates source reliability and assertion truth value, assuming all sources

are *independent*.

- EM-Social (IPSN 2014): This algorithm improves upon EM (IPSN 2012) by ignoring dependent claims, reasoning that they do not add new information pertinent to truth determination.
- Optimal: This value refers to a transformed error bound, (i.e., $1 - Err$), introduced earlier. No fact-finder can outperform this bound on average. (Some estimators may perform better on false negatives or false positives but not on overall accuracy).

Model parameter settings are the same as in Section V-A except that now we take $n = 50$ for default. We conduct 300 independent experiments for each simulation experiment and take the average.

In the first simulation, we change the total number of sources, n, while keeping other parameters at default values. We change n from 20 to 50 in steps of 5. Simulation results are illustrated in Figure 7. We can see that increasing the number of sources improves performance of most algorithms except for EM. As shown in Figure 7-(b), the false positive rate grows with the number of sources, because the EM algorithm has no ways of dealing with dependencies. Increasing the number of sources without adding more assertions creates the illusion of more substantiated assertions if dependencies are not considered. In Figure 7-(b), although the false negative rate of EM-Ext is large than other two EM algorithm, the absolute value is relatively small. And the false negative rate of EM-Ext is similar as that of the optimal bound, while other two algorithm seem to be a little biased to predict assertion to be true.

In the second simulation, we change the total number of assertions, m, and set $n = 100$, while keeping other parameters at default values. We change m from 10 to 100 in steps of 10. Simulation results are illustrated in Figure 8. We see that increasing the number of assertions improves the performance of all algorithms. The difference between EM-Ext and the optimal algorithm shrinks as the number of assertions increases.

In the third simulation, we see how the number of dependency trees affects the performance of each algorithm. We change the number of dependency trees, τ from 1 to 11 in steps of 1, while keeping other parameters at default values. Results are shown in Figure 9. The EM-Ext algorithm outperforms the other two algorithms across the board.

In the final simulation, we vary the parameter p_i^{indepT} and p_i^{depT} to see how reliability of sources affects fact-finding performance. We choose $\frac{p_i^{depT}/(1-p_i^{depT})}{p_i^{indepT}/(1-p_i^{indepT})}$ as our tuning knob, as $p_i^{depT}/(1-p_i^{depT})$ and $p_i^{indepT}/(1-p_i^{indepT})$ are good indicators of how effectively one can distinguish true and false assertions. For this simulation, we keep $p_i^{indepT}/(1-p_i^{indepT}) = 2$ and change $p_i^{depT}/(1-p_i^{depT})$ from 1.1 to 2.0 in steps of 0.1, as illustrated in Figure 10. When $p_i^{depT}/(1-p_i^{depT})$ increases, the dependent claims contribute additional information for telling true and false assertions apart, thus all algorithms benefit, except EM-social, as it "deletes" dependent claims. Another interesting observation is that when $p_i^{depT}/(1-p_i^{depT}) \approx p_i^{indepT}/(1-p_i^{indepT})$, EM algorithm tends to perform similarly or even slightly better than EM-Social. The reason is that dependent and independent claims tend to be equivalent in this case. So the EM algorithm, which treats all claims as independent, uses more data to learn less latent parameters more accurately. When $p_i^{depT}/(1-p_i^{depT}) \approx 1$, EM-Ext tends to perform similarly as EM-Social. The reason is that dependent claims provide very less information at that time. Therefore ignoring the dependent claims leads to little information loss.

C. Empirical Evaluation

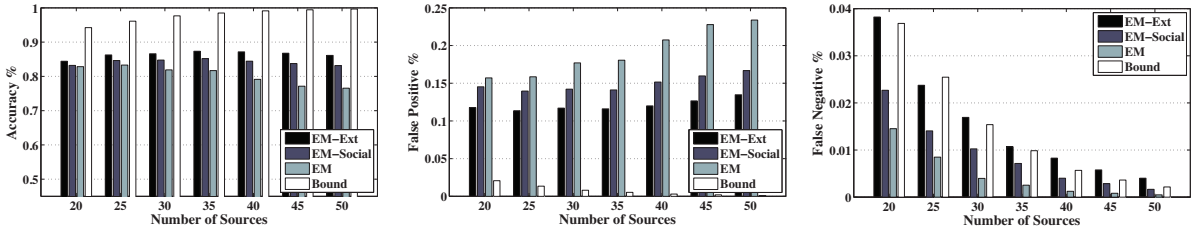
In this section we provide empirical evaluation of our EM-Ext algorithm on five Twitter datasets[3] that we collected in 2015 with different keyword triplets and geo-locations. Table III provides a brief summarization of these five tasks, (i) Ukraine, (ii) Kirkuk, (iii) Superbug, (iv) LA Marathon, and (v) Paris Attack, detailed as follows:

- Ukraine: On March 14th 2015, The Russian President Vladimir V. Putin has not shown in public for more than one week. During that week, he had postponed a treaty signing with representatives from South Ossetia and canceled the trip to Kazakhstan. Speculations appeared in news and social media. Some rumors even said that the Russian president was dead, which was not true. These noisy messages provide a good environment for testing our state estimator.
- Kirkuk: On March 10th 2015, Kurdish forces attacked the Islamic State of Iraq and Syria (ISIS) locations around the oil-rich city of Kirkuk in northern Iraq. A lot of commentaries were posted and followed on social media.
- Superbug: On Mar 4th 2015, Second Los Angeles Hospital Reports four patients have been infected with an antibiotic-resistant "superbug".
- LA Marathon: Los Angeles 2015 Marathon was on Mar 15th. A lot of people posted event of Marathon on social media along its route from Dodger Stadium to Santa Monica Pier.
- Paris Attack: We show results from Nov 14th. On the night of November 13th, a series of coordinated terrorist attacks occurred in Paris.

We apply seven different algorithms to evaluate these datasets. Besides the "EM-Ext", "EM-Social", and "EM" algorithms introduced earlier, four additional heuristics are included, as follows:

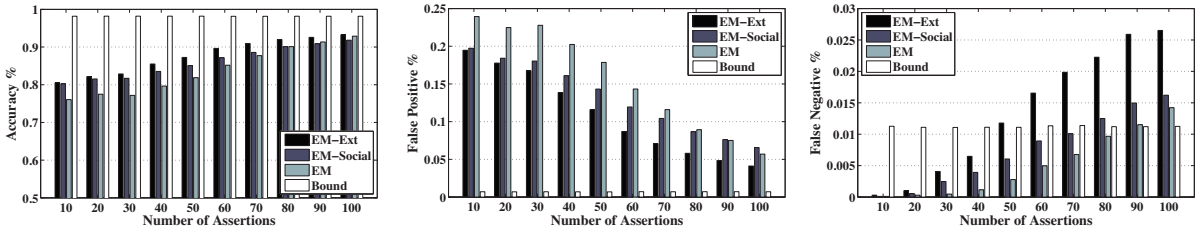
- *Voting:* This algorithm ranks assertions according to the total number of times being made (e.g., total number of tweets making the same statement). The larger this number, the more credence is given to the assertion.
- *Sums:* An iterative algorithm [15]. It estimates the reliability of assertions and sources in turn by counting the number of sources and assertions that support them.

[3]Available for download at http://apollo3.cs.illinois.edu/



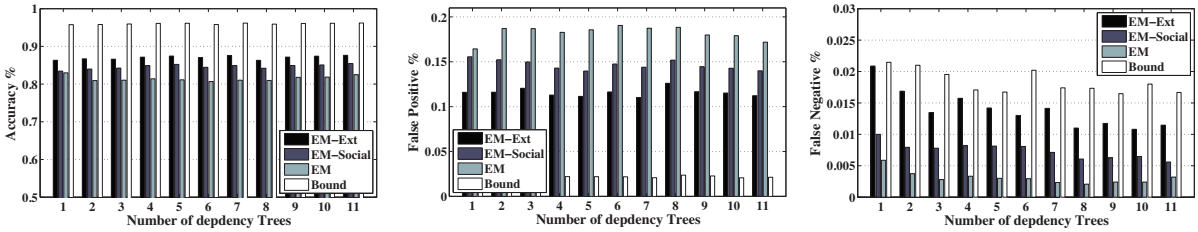
(a) Accuracy (b) False Positive Error (c) False Negative Error

Fig. 7: Varying # of sources.



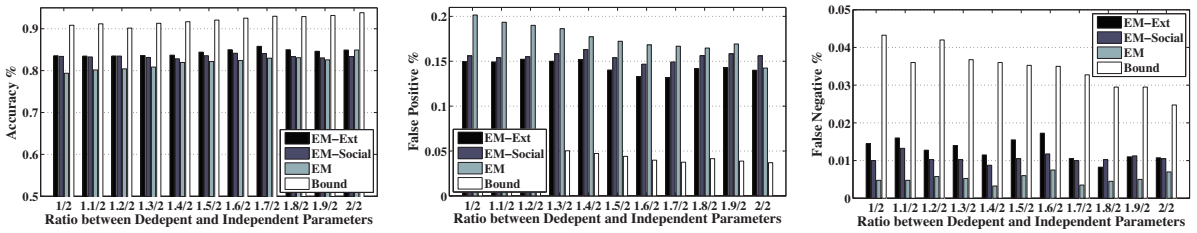
(a) Accuracy (b) False Positive Error (c) False Negative Error

Fig. 8: Varying # of assertions.



(a) Accuracy (b) False Positive Error (c) False Negative Error

Fig. 9: Varying # of dependency trees.



(a) Accuracy (b) False Positive Error (c) False Negative Error

Fig. 10: Varying source reliability.

- *Average.Log:* A variant of Sums. It makes a trade-off to trust more on sources who make more true assertions. During each iteration, source reliability is weighted by claims it has made.
- *Truth-Finder:* An iterative algorithm [22]. It utilizes the relationship between source and assertion reliabilities to find trustworthy sources and true assertions.

For actual evaluation, we collected the top-100 tweets as identified by each algorithm according to their computed truth probabilities, and merged and mixed them together, with the generating algorithms anonymized. Human graders then manually graded all tweets without knowing which algorithm generated which tweet to prevent bias. Graders were required to do background research on each tweet and mark it as "True", "False", or "Opinion" according to the following rule:

- True: Tweets making a verifiable assertion that was confirmed to be true by the grader
- False: Tweets making a verifiable assertion that was confirmed to be false by the grader.
- Opinion: Tweets making a subjective assessment such as "President Arthur is good" or tweets that do not constitute an act of sensing (e.g., "Please support dolphins in Australia").

The algorithms were then de-anonymized, and we computed the percentage of assertions found True in the output of each algorithm. That is to say, we computed the ratio #True/(#True + #False + #Opinion).

TABLE III: Information Summary of Twitter Datasets.

	Total Start Time (UTC)	Total End Time (UTC)	Evaluation Day	#Assertions	#Sources	#Total Claims	#Original Claims	Locations
Ukraine	Feb 20 12:15:28 2015	Mar 31 23:10:12 2015	Mar 14 2015	3703	5403	7192	4242	Ukraine
Kirkuk	Jan 31 01:47:25 2015	Apr 02 02:41:15 2015	Mar 10 2015	2795	4816	6188	3079	Kirkuk
Superbug	Feb 19 17:42:39 2015	Apr 09 18:29:01 2015	Mar 4 2015	2873	7764	9426	5831	LA
LA Marathon	Mar 12 01:38:29 2015	Mar 18 02:14:42 2015	Mar 15 2015	3537	5174	7148	4332	LA
Paris Attack	Nov 14 18:17:14 2015	Nov 24 17:28:02 2015	Nov 14 2015	23513	38844	41249	38794	Paris

The accuracy of each algorithm is shown in Figure 11. As seen, the EM-Ext algorithm outperforms all other algorithms.

Let us now take a closer look at Figure 11. The basic EM algorithm performs better than Voting, because it takes source reliability into consideration, providing better estimation accuracy than majority vote.

The EM-Social algorithm usually performs better than other baseline algorithms, but is beaten by our EM-Ext algorithm. Although EM-Social takes source dependencies into consideration, it simply ignores dependent claims as an additional data cleaning process. This flaw limits the amount of data it can use, resulting in less accurate estimation.

Three iterative algorithms: Sums, Average.Log, and Truth Finder perform with high variance. In different datasets, they sometimes perform better than EM and Social EM algorithms, but other times not. The main reason is that their models fail to take source dependencies into consideration, and use suboptimal algorithms to estimate source reliability.

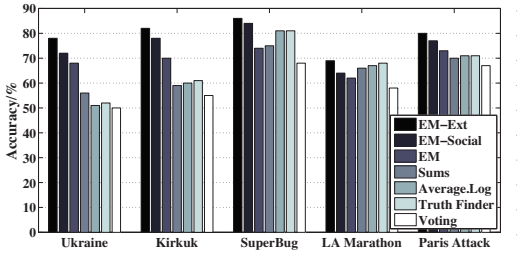


Fig. 11: Empirical Accuracy Results.

VI. Related Work

Due to the proliferation of mobile sensor and smart devices as well as the popularity of social media, social sensing has become a key research topic in sensor networks research attracting growing attentions. During social sensing processes, people act as sensor carriers [4] or sensors themselves [16]. We focus on humans as sensors in this work.

Quality and trustworthiness of data is a key problem for social sensing. Human sensor can easily bring noise to reported data via distortion, fabrication, omissions, and duplication. Recent work focus on estimating reliability of reported sensing data, which can found in both recent machine learning data mining literature [11], [12], [22], crowd sourcing literature [8], [9], [14] and sensor network literature [16], [18], [19], [21].

The basic fact-finder [10] is one of the earliest efforts in this domain. Later Yin et al. introduced an unsupervised fact-finder, TruthFinder [22], for analyzing veracity of providers-facts networks. Pasternack et al. proposed several extended algorithms [15]. Wang et al. [18] first proposed an algorithm that jointly estimates assertion and source reliability, and further extended the maximum-likelihood estimation by assuming that claims repeated by dependent sources do not offer added values from the perspective of truth estimation [16]. Yao et al. propose a recursive estimator for sensing streaming data [21] The above work on reliability estimation fails to fully capture source dependencies. This paper takes into account the observed source dependency structure, develops a simplified dependency model, and designs an improved dependency-aware estimator to assess veracity of observations. What's more, this work is also motivated by the goal of providing performance bound for social sensing. Performance and capacity bounds have been studied in communication and networking systems [1], [13]. For the area of social sensing, [17] provided a way of estimating error bounds for social sensing algorithms, by computing the quantified confidence of estimated parameters with Cramer-Rao lower bound with attainable approximation to trade accuracy with scalability.

This paper is different from the former effort in that a fundamental error bound, on claim misclassification rate, for the performance of social social sensing models is derived. The error bound helps to understand the gap between the accuracy of proposed algorithm and the theoretical optimal performance of social sensing model. The performance error bound proposed in this paper is also a good indicator of what performance gain can be achieved with advanced estimating algorithms under the existing social sensing model.

VII. Conclusion

In this paper, we developed fundamental error bounds on fact-finding accuracy in social channels, based on a novel model that approximates social media as noisy communication channels. While computing the exact bound takes an exponential time in the number of sources, we also developed a tractable approximation of the bound and demonstrated its accuracy. A practical maximum-likelihood estimator was then developed . It is evaluated both in simulation and using empirical data. Results demonstrate improved performance compared with state of the art fact-finders.

Acknowledgements

We sincerely thank the anonymous reviewers for their invaluable comments. Research reported in this paper was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement W911NF-09-2-0053, DTRA grant HDTRA1-10-10120, and NSF grants CNS 09-05014 and CNS 10-35736. 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.

APPENDIX

To solve the expectation maximization problem, we first derive the expectation step:

$$\mathcal{Q}(\theta|\theta^{(t)}) = \sum_{C\in\{0,1\}^m} P(C|SC;\theta^{(t-1)}) \ln\Big(P(SC|C;\theta)P(C;\theta)\Big) \tag{15}$$

Since C and S can be separated into $\{C_1,\cdots,C_m\}$ and $\{S_1,\cdots,S_n\}$ independently, we can converted equ. (15) into:

$$\mathcal{Q}(\theta|\theta^{(t)}) = \sum_{j=1}^{m} P(C_j|SC_j;\theta^{(t)}) \sum_{C_j\in\{0,1\}} \ln(P(C_j;\theta)) \Big(\sum_{i=1}^{n} \ln(P(S_iC_j|C_j;\theta,D_{ij})\Big) \tag{16}$$

where $P(S_iC_j|C_j;\theta,D_{ij})$ is given by Table II, and

$$P(C_j=1|SC_j;\theta^{(t)}) = \frac{P(SC_j|C_j=1;\theta^{(t)})P(C_j=1;\theta^{(t)})}{\sum_{C_j\in\{0,1\}} P(SC_j|C_j;\theta^{(t)})P(C_j;\theta^{(t)})} \tag{17}$$

Then we need to go through the maximisation step:

$$\theta^{(t+1)} = \arg\max_{\theta} \mathcal{Q}(\theta|\theta^{(t)}) \tag{18}$$

To solve equ (18) analytically, the general solution is taking gradient of all parameter θ with Equ. (15) and making them equal to 0.

$$\frac{\partial\mathcal{Q}(\theta|\theta^{(t)})}{\partial a_i} = \frac{\sum_{C_j\in S_iC_1^{D_0}} Z_j}{a_i} - \frac{\sum_{C_j\in S_iC_0^{D_0}} Z_j}{1-a_i} \tag{19}$$

$$\frac{\partial\mathcal{Q}(\theta|\theta^{(t)})}{\partial f_i} = \frac{\sum_{C_j\in S_iC_1^{D_1}} Z_j}{f_i} - \frac{\sum_{C_j\in S_iC_0^{D_1}} Z_j}{1-f_i} \tag{20}$$

$$\frac{\partial\mathcal{Q}(\theta|\theta^{(t)})}{\partial b_i} = \frac{\sum_{C_j\in S_iC_1^{D_0}} Y_j}{b_i} - \frac{\sum_{C_j\in S_iC_0^{D_0}} Y_j}{1-b_i} \tag{21}$$

$$\frac{\partial\mathcal{Q}(\theta|\theta^{(t)})}{g_i} = \frac{\sum_{C_j\in S_iC_1^{D_1}} Y_j}{g_i} - \frac{\sum_{C_j\in S_iC_0^{D_1}} Y_j}{1-g_i} \tag{22}$$

$$\frac{\partial\mathcal{Q}(\theta|\theta^{(t)})}{\partial z} = \frac{\sum_{j=1}^{m} Z_j}{z} - \frac{\sum_{j=1}^{m} Y_j}{1-z} \tag{23}$$

where $Z_j = P(C_j=1|SC_j;\theta^{(t)})$ and $Y_j = P(C_j=0|SC_j;\theta^{(t)})$.

We let gradient, Equ (19) - (23), of each parameter to be 0. Then we are able to obtain the answer shown as follow,

$$a_i^{(t+1)} = \frac{\sum_{C_j\in S_iC_1^{D_0}} P(C_j=1|S_iC_j;\theta^{(t)})}{\sum_{C_j\in S_iC_1^{D_0}\bigcup S_iC_0^{D_0}} P(C_j=1|S_iC_j;\theta^{(t)})} \tag{24}$$

$$f_i^{(t+1)} = \frac{\sum_{C_j\in S_iC_1^{D_1}} P(C_j=1|S_iC_j;\theta^{(t)})}{\sum_{C_j\in S_iC_1^{D_1}\bigcup S_iC_0^{D_1}} P(C_j=1|S_iC_j;\theta^{(t)})} \tag{25}$$

$$b_i^{(t+1)} = \frac{\sum_{C_j\in S_iC_1^{D_0}} P(C_j=0|S_iC_j;\theta^{(t)})}{\sum_{C_j\in S_iC_1^{D_0}\bigcup S_iC_0^{D_0}} P(C_j=0|S_iC_j;\theta^{(t)})} \tag{26}$$

$$g_i^{(t+1)} = \frac{\sum_{C_j\in S_iC_1^{D_1}} P(C_j=0|S_iC_j;\theta^{(t)})}{\sum_{C_j\in S_iC_1^{D_1}\bigcup S_iC_0^{D_1}} P(C_j=0|S_iC_j;\theta^{(t)})} \tag{27}$$

$$z^{(t+1)} = \frac{\sum_{j=1}^{m} P(C_j=0|S_iC_j;\theta^{(t)})}{m} \tag{28}$$

Given the above description, we can estimate the assertion and source reliabilities jointly with Equ (17) and (24) - (28) iteratively until they converge.

REFERENCES

[1] T. F. Abdelzaher, S. Prabh, and R. Kiran. On real-time capacity limits of multihop wireless sensor networks. In *Proc. RTSS*, 2004.

[2] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan. An introduction to mcmc for machine learning. *Machine learning*, 2003.

[3] H. Attias. Inferring parameters and structure of latent variable models by variational bayes. In *Proceedings of UAI*, 1999.

[4] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava. Participatory sensing. In *In: Workshop on World-Sensor-Web*, 2006.

[5] C. K. Carter and R. Kohn. On gibbs sampling for state space models. *Biometrika*, 1994.

[6] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1984.

[7] W. R. Gilks. *Markov chain monte carlo*. Wiley Online Library, 2005.

[8] H. Jin, L. Su, D. Chen, K. Nahrstedt, and J. Xu. Quality of information aware incentive mechanisms for mobile crowd sensing systems. In *MobiHoc*, 2015.

[9] H. Jin, L. Su, H. Xiao, and K. Nahrstedt. Inception: Incentivizing privacy-preserving data aggregation for mobile crowd sensing systems. In *MobiHoc*, 2016.

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

[11] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *SIGMOD*, 2014.

[12] Y. Li, J. Gao, C. Meng, Q. Li, L. Su, B. Zhao, W. Fan, and J. Han. A survey on truth discovery. *ACM SIGKDD Explorations Newsletter*, 2016.

[13] S. Liu, M. Chen, S. Sengupta, M. Chiang, J. Li, P. Chou, et al. P2p streaming capacity under node degree bound. In *Proc. ICDCS*, 2010.

[14] C. Meng, W. Jiang, Y. Li, J. Gao, L. Su, H. Ding, and Y. Cheng. Truth discovery on crowd sensing of correlated entities. In *SenSys*, 2015.

[15] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). In *Proceedings of COLING*, 2010.

[16] D. Wang, M. T. Amin, S. Li, T. Abdelzaher, L. Kaplan, S. Gu, C. Pan, H. Liu, C. C. Aggarwal, R. Ganti, et al. Using humans as sensors: An estimation-theoretic perspective. In *Proceedings of IPSN*, 2014.

[17] 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 *Proc. SECON*, 2012.

[18] D. Wang, L. Kaplan, H. Le, and T. Abdelzaher. On truth discovery in social sensing: A maximum likelihood estimation approach. In *Proceedings of IPSN*, 2012.

[19] S. Wang, L. Su, S. Li, S. Hu, M. T. Amin, H. Wang, S. Yao, L. K. Kaplan, and T. Abdelzaher. Scalable social sensing of interdependent phenomena. In *Proc. IPSN*, 2015.

[20] C. J. Wu. On the convergence properties of the em algorithm. *The Annals of statistics*, pages 95–103, 1983.

[21] S. Yao, M. T. Amin, L. Su, S. Hu, S. Li, S. Wang, Y. Zhao, T. Abdelzaher, L. Kaplan, C. C. Aggarwal, and A. Yener. Recursive ground truth estimator for social data streams. In *Proceedings of IPSN*, 2016.

[22] X. Yin, J. Han, and P. S. Yu. Truth discovery with multiple conflicting information providers on the web. *Knowledge and Data Engineering, IEEE Transactions on*, 2008.