This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

Aggregating Crowdsourced Quantitative Claims: Additive and Multiplicative Models

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

Abstract—Truth discovery is an important technique for enabling reliable crowdsourcing applications. It aims to automatically discover the truths from possibly conflicting crowdsourced claims. Most existing truth discovery approaches focus on *categorical* applications, such as image classification. They use the accuracy, i.e., rate of exactly correct claims, to capture the reliability of participants. As a consequence, they are not effective for truth discovery in *quantitative* applications, such as percentage annotation and object counting, where similarity rather than exact matching between crowdsourced claims and latent truths should be considered. In this paper, we propose two Quantitative Truth Finders (QTFs) for truth discovery in quantitative crowdsourcing applications. One QTF explores an additive model and the other explores a multiplicative model to capture different relationships between crowdsourced claims and latent truths in different classes of quantitative tasks. These QTFs naturally incorporate the similarity between variables. Moreover, they use the bias and the confidence instead of the accuracy to capture participants' abilities in quantity estimation. These QTFs are thus capable of accurately discovering quantitative truths in particular domains. Through experiments, we demonstrate that these QTFs outperform other state-of-the-art approaches for truth discovery in quantitative crowdsourcing applications and they are also quite efficient.

Index Terms—Crowdsourcing, truth discovery, quantitative task, probabilistic graphical model

1 Introduction

Crowdsourcing is becoming increasingly popular as it provides an easy, time-, and cost-efficient way to collect a large volume of data for a variety of applications. Crowds have been explored to perform various human intelligence tasks such as image classification, image description, sentiment analysis, listing verification, object counting, transcription, translation, word processing, and logo design [2], [11], [12], [14], [17], [23], [24], [34], [36]. Besides performing such tasks on crowdsourcing platforms such as the Amazon Mechanical Turk[1] and CrowdFlower[2], crowds have also been utilized to detect events in the physical world, e.g., to detect earth quakes [28], to detect desired flora on campus [25], and to detect social disorder in public places [18].

However, the quality of data obtained from crowd participants is often much lower than the quality of data collected from traditional employees and experts [11]. Some participants are highly skilled and provide high-quality data, while some are unskilled or sloppy that often provide low-quality or even random data [9], [11]. As a result, truth discovery [13] is an important technique for enabling *reliable* crowdsourcing applications. It aims to automatically discover the truths from possibly conflicting and low-quality crowdsourced data.

In this paper, we address the truth discovery problem in *quantitative* crowdsourcing applications. This is motivated by the various quantitative applications that crowdsourcing can enable. For example, crowdsourced quantitative claims can be used as labeled data to train automatic sensor-based or vision-based systems for occupancy inference in buildings [10], wildlife monitoring [29], and people counting in public places [4], [16]. Crowdsourcing can also be explored to enable new real-world quantitative applications such as documenting the bird populations around the world [26], counting the ballots to protect against election fraud [15], and enabling well informed decision in smart city applications (e.g., real-time information about waiting line length and occupancy levels of interest), where physical sensor networks are too costly or have limited coverage [20].

Although a number of approaches such as [19], [21], [24], [30], [33]–[35] have been proposed for effective truth discovery, they focus on *categorical* crowdsourcing applications, such as image classification (e.g., does this image show a dog or a cat?). They rely on the agreement among claims from multiple crowd participants and define the reliability of crowd participants as the accuracy (i.e., rate of exactly correct claims). They are thus not effective for truth discovery in quantitative applications such as percentage annotation (e.g., in what percentage is

- *R. W. Ouyang is with the CAS Key Laboratory of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. This work was done when R. W. Ouyang was a postdoc at the University of California, Los Angeles, CA, USA. E-mail: ouyangwt@ict.ac.cn.*
- *L. M. Kaplan is with the Networked Sensing & Fusion Branch, US Army Research Laboratory, Adelphi, MD 20783, USA. E-mail: lance.m.kaplan@us.army.mil.*
- *A. Toniolo and T. J. Norman are with the Department of Computing Science, University of Aberdeen, Aberdeen, UK. E-mail: {a.toniolo, t.j.norman}@abdn.ac.uk.*
- *M. Srivastava is with the Department of Electrical Engineering and the Department of Computer Science, University of California, Los Angeles, CA, USA. E-mail: mbs@ucla.edu.*

1. https://www.mturk.com/mturk/welcome
2. http://www.crowdflower.com

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

this classroom occupied?) and object counting (e.g., how many people are in this image?), where similarity (rather than exact matching) between crowdsourced claims and latent truths should be considered. Moreover, it is not uncommon that each participant provides a different claim in a quantitative task, and the accuracy of any participant is small. As a result, the lack of an effective truth discovery method for quantitative crowdsourcing applications may impair their usefulness.

Truth discovery in quantitative crowdsourcing applications is a challenging problem. First, some crowd participants can accurately estimate a quantity, some tend to overestimate, some tend to underestimate, and some even randomly guess. However, such ability is unknown a priori and thus it is difficult to determine whose claims to trust. Second, the quantitative truths need to be found in an unsupervised manner as it is often expensive or difficult to manually collect and annotate the ground truth for supervised model training in reality. Third, as has been mentioned, most existing truth discovery methods [19], [21], [24], [30], [33]–[35] focus on aggregating categorical claims. Due to the different natures of categorical and quantitative claims, these methods cannot be easily modified to address the truth discovery problem in quantitative crowdsourcing applications. Fourth, different classes of tasks in quantitative crowdsourcing applications, such as percentage annotation and object counting, have their own properties, and tailored methods may need to be designed for truth discovery.

To tackle these challenges, in this paper, we propose Quantitative Truth Finders (QTFs) for truth discovery in quantitative crowdsourcing applications. In particular, we explore two models, namely, the QTF-Additive (QTF-A) and the QTF-Multiplicative (QTF-M) models, to capture different relationships between crowdsourced claims and latent truths in different classes of tasks. In QTF-A, we model that a crowdsourced claim is generated by adding an error term to the latent truth. QTF-A is thus suitable for truth discovery in tasks (such as percentage annotation) where errors in the claims are approximately independent of the latent truths. In QTF-M, we model that a crowdsourced claim is generated by multiplying a ratio term to the latent truth. QTF-M is thus suitable for truth discovery in tasks (such as object counting) where errors in the claims are positively impacted by the latent truths. In QTFs, we use the bias and the confidence instead of the accuracy to capture participants' abilities. These QTFs also naturally incorporate the similarity between crowdsourced claims and latent truths. As a result, they are capable of accurately discovering quantitative truths. Through experiments, we demonstrate that these QTFs are more effective than other state-of-the-art approaches for truth discovery in quantitative crowdsourcing applications.

In summary, our main contributions are as follows:

1) We propose QTF-A and QTF-M models for truth discovery in quantitative crowdsourcing applications. These models jointly assess latent quantitative truths and abilities of participants from noisy crowdsourced claims without any supervision.
2) We develop efficient model inference algorithms.
3) We perform extensive experiments to evaluate the performance of various truth discovery methods in quantitative crowdsourcing applications.

The remainder of this paper is organized as follows. We review related work in Section 2. We then formalize the problem, analyze crowdsourced claims in representative quantitative tasks, and discuss intuitions that motivate the components of QTFs in Section 3. We then present our proposed QTF-A and QTF-M models, along with model inference algorithms in Sections 4 and 5 respectively. Experimental results are presented in Section 6. We discuss other research problems in Section 7. Finally, we conclude the paper in Section 8.

2 Related Work

Truth discovery is an important technique for enabling reliable crowdsourcing applications. In the domain of truth discovery from conflicting Web information, Yin et al. [35] proposed truth finder, which is a transitive voting algorithm with rules specifying how votes iteratively flow from sources to claims and then back to sources. Pasternack and Roth [21] proposed AverageLog, Investment, and PooledInvestment algorithms. Zhao et al. [37] proposed a principled probabilistic approach which can automatically infer true claims and two-sided source quality. Pasternack et al. [22] proposed probabilistic latent credibility analysis models.

In the domain of aggregating conflicting claims in crowdsourcing applications, Dawid and Skene [8] modeled the generative process of the claims by introducing source ability parameters. Whitehill et al. [34] further included the task difficulty in the model. Welinder et al. [33] proposed a model consisting of worker compatibility for each task. Wang et al. [30] proposed a model for truth discovery in social sensing. They further extended their model to consider potentially dependent information sources in [32]. Baba et al. [1] proposed a model for truth discovery in general crowdsourcing tasks such as article writing and logo design. Ouyang et al. [19] proposed a model for truth discovery in crowdsourced detection of spatial events.

Nevertheless, these methods are mainly designed for categorical truth discovery. They rely on the agreement among claims and define the reliability of participants as the accuracy. As a consequence, these methods are not effective for the quantitative truth discovery problem addressed in this paper.

Two representative studies that specially address this problem are the one by Zhao et al. [38] and our previous work [20]. Zhao et al. proposed the Gaussian Truth Model (GTM) in [38], which addresses the truth discovery problem for numerical data. GTM models the source quality in terms of the variance parameter of a Gaussian distribution. It first performs outlier detection and data

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

normalization using the z-score before model inference. As the normalization is with respect to the claims for a target quantity, it cannot make the normalized claims of a participant Gaussian distributed (while a participant's claims are modeled as Gaussian distributed). Therefore, there may exist model mismatch. Moreover, GTM does not consider the source bias, but assumes that the claims are centered around the corresponding latent truth. However, we observe that biases are common in participants' claims (shown in Section 3.2).

We recently proposed the Truth, Bias, and Precision (TBP) model in [20] for truth discovery in quantitative crowdsourcing applications. TBP models the latent quantitative truths, the task difficulty levels, and the biases and precisions[3] of crowd participants. It can be shown that, the probability of observing a claim under the TBP model follows a Gaussian mixture distribution, which can well approximate any continuous distribution by tuning its parameters [3]. Therefore, TBP can well model the errors in crowdsourced claims in various classes of tasks. However, the need to find the optimal number K of model components in an unsupervised manner is non-trivial, and thus suboptimal results in truth discovery may be obtained. Moreover, overfitting may occur when the number of claims per participant is small, since TBP needs to infer $2K$ ability parameters per participant.

In this paper, we propose QTFs for truth discovery in quantitative crowdsourcing applications. QTFs also model latent quantitative truths and participants' ability parameters. But different from TBP, QTFs consider different models that capture different relationships between crowdsourced claims and latent truths in different classes of quantitative tasks. In particular, QTF-A explores an additive model and QTF-M explores a multiplicative model. Compared with GTM, QTF-A and QTF-M do not involve improper data normalization and they use robust scale estimation to deal with outliers. Moreover, they model participants' biases while GTM does not. Compared with TBP, QTF-A and QTF-M do not need to find the optimal number of model components, they have fewer parameters to infer, and they are also much easier to implement.

3 Overview

In this section, we first present the statement of the quantitative truth discovery problem addressed in this paper. We then examine the properties of crowdsourced claims in two representative classes of tasks, which are percentage annotation and object counting. Based on these properties, we discuss two general QTF models for quantitative truth discovery.

3.1 Problem Statement

For ease of presentation, we list the notations used in this paper in Table 1. Consider a task where M crowd participants u_i make quantitative claims x_{ij} (e.g., 5, 12, and 20) on N target quantities z_j (e.g., the number of people in the jth image). We only observe the crowdsourced claims x_{ij}, but not the true quantity values z_j. The truth discovery problem in quantitative crowdsourcing applications is to automatically recover the true quantity values z_j from crowdsourced claims x_{ij}.

TABLE 1
Notations.

Notation	Meaning
M	# of crowd participants
N	# of target quantities
u_i	ith participant
z_j	true value of the jth quantity
$\mathcal{U}_j$	set of participants who make a claim on z_j
$\mathcal{Z}_i$	set of quantities that u_i makes a claim on
h_i	u_i's bias
λ_i	u_i's confidence
x_{ij}	claim that u_i makes on z_j
μ_j, ν_j	hyperparameters for z_j
a_i, b_i	hyperparameters for λ_i

3.2 Data Analysis

We consider two representative classes of tasks in quantitative crowdsourcing applications:

1) **Percentage annotation.** In this class of tasks, participants are asked to annotate the percentage of interest in given images, video frames, or scenarios. Example tasks include annotating the occupancy rate (in terms of percentage) of seats in classrooms, annotating the occupancy rate of fitness machines in gyms, and annotating the occupancy rate of tables in restaurants [20].
2) **Object counting.** In this class of tasks, participants are asked to count the number of interest in given images, video frames, or scenarios. Example tasks include counting the number of people waiting in line or gathering in public places, and counting the number of birds in the wild [16], [20], [26].

The usefulness of applications that these tasks can enable has been discussed in Section 1.

In order to build appropriate models for aggregating crowdsourced claims in these tasks, we start by examining the errors in the claims. We define the error of the ith participant on the jth target quantity as $e_{ij} \equiv x_{ij} - z_j$.

Note that, in all the analysis and figures in this section, the latent true values z_j are assumed known, and we put our focus on examining the relationship between x_{ij} and z_j. In the next section, we design models that can automatically infer unknown z_j from known x_{ij}, based on the observations made in this section.

3.2.1 Data Collection

We conducted six sets of experiments to collect crowdsourced quantitative claims. Three sets of them are percentage annotation tasks, which are 1) annotating

3. Precision is the inverse of the variance parameter of a Gaussian distribution.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

TABLE 2
Statistics of crowdsourced datasets (pt - participant, avg - average, occ - occupancy, rest - restaurant, ct - count).

Task	# tasks	# pts	# total claims	avg # pts per task	avg # tasks per pt	ground truth (median, min, max)
Room occ	200	32	2000	10	62.5	(38.5, 0, 100)
Gym occ	200	35	2000	10	57.1	(32, 0, 86)
Rest occ	200	29	2000	10	69.0	(46, 0, 100)
People ct	200	36	2000	10	55.6	(22.5, 6, 98)
Vehicle ct	200	43	2000	10	46.5	(23, 6, 91)
Bird ct	200	41	2000	10	48.8	(21, 5, 100)

the occupancy rate (in terms of percentage) of seats in classrooms, 2) annotating the occupancy rates of fitness machines in gyms, and 3) annotating the occupancy rates of tables in restaurants. The other three sets are object counting tasks, which are 1) counting the number of people, 2) counting the number of vehicles, and 3) counting the number of birds. For each set of experiments, we first randomly searched 200 appropriate images from Google Images. Two people then carefully annotated the ground truth in each image. We then posted these images on CrowdFlower as tasks to collect crowdsourced claims (10 participants per task; each task contains a single target quantity). For the people counting task, we masked the people faces. The statistics of the collected data are listed in Table 2. The similar statistics of the ground truth values in the three object counting tasks may be resulted from randomization. Nevertheless, we observe quite different truth discovery results in these tasks (shown in Section 6).

3.2.2 Percentage Annotation

Fig. 1(a) plots the errors of 30 randomly selected participants in percentage annotation tasks. It is observed that most participants have clear biases, as their errors are not symmetric around zero. Some participants tend to overestimate with positive biases, some tend to underestimate with negative biases, while some are close to unbiased. It is also observed that, for most participants, their errors are relatively symmetric around respective centers. These observations suggest that a Gaussian distribution with an unknown mean and an unknown variance (referred to as "Gaussian" for short) may be used to model most participants' errors in percentage annotation tasks. We also consider whether a zero-mean Gaussian with an unknown variance (referred to as "zero-mean Gaussian" for short) suffices. We illustrate in Fig. 2(a) example model fitting results on the errors of a representative participant. It is clear that the Gaussian distribution closely matches the histogram of the errors.

We then test the goodness-of-fit of these distributions using the Kolmogorov-Smirnov test (K-S test) [7], which can be used to decide whether a sample comes from a population with a reference distribution. Among all the participants in the three percentage annotation tasks, 95.8% and 45.8% participants pass the K-S test (at the 5% significance level) when we fit their errors using a Gaussian and a zero-mean Gaussian distribution respectively. These results show that, we cannot simply assume that participants are unbiased, but have to infer their unknown biases as well as their unknown variances. We thus use a Gaussian distribution to model participants' errors in percentage annotation tasks.

3.2.3 Object Counting

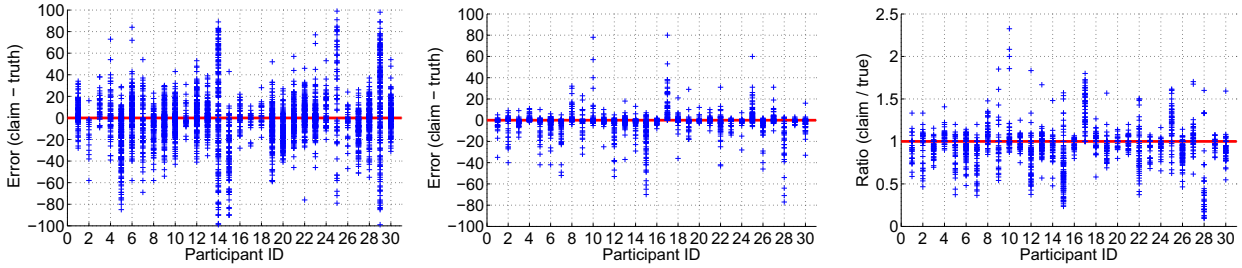
Fig. 1(b) plots the errors of 30 randomly selected participants in object counting tasks. It is observed that most participants' errors are also biased. Moreover, most participants' errors are skewed and even show multiple modes. These observations suggest that a standard parametric distribution is insufficient to model participants' errors. We illustrate in Fig. 2(b) the histogram of a representative participant's errors, along with a zero-mean Gaussian distribution and a Gaussian distribution fitted to the errors. It is observed that these distributions do not fit the errors well. K-S test results also show that only 26.7% and 13.3% participants pass the test when we fit their errors using a Gaussian distribution and a zero-mean Gaussian distribution respectively.

To tackle this problem, the TBP model [20] introduces the notion of task difficulty levels (such as easy, normal, and hard). When conditioning on a task difficulty level, the errors can be modeled by a Gaussian distribution. When the task difficulty level is unobserved, the errors then follow a Gaussian mixture distribution, which can well approximate any continuous distribution by tuning its parameters [3], including that with multiple modes (an example is shown in Fig. 2(b)). However, as discussed in Section 2, it is non-trivial for TBP to find the optimal number of model components in an unsupervised manner for each specific dataset. Moreover, overfitting may occur for complex models like TBP.

Different from TBP, we consider to model alternative attributes other than the errors in crowdsourced claims. We instead examine the ratios which we define as $w_{ij} \equiv x_{ij}/z_j$. This is motivated by the intuition that in object counting tasks, the errors in crowdsourced claims seem to be positively impacted by the values of latent truths. For a difficult task with a large latent true count, the error in the claim may also be large; while for an easy task with a small latent true count, the error in the claim may also be small. Although a participant's errors e_{ij} can vary in a wide range, the corresponding ratios w_{ij} may be quite similar. For example, a participant may make a claim as $x_{ij} = 100$ when the latent true value is $z_j = 80$, and she may make a claim as $x_{ij} = 24$ when the latent true value is $z_j = 20$. Although the corresponding errors are 20 and 4 which differ a lot, the corresponding ratios are 1.25 and 1.2 which are quite similar.

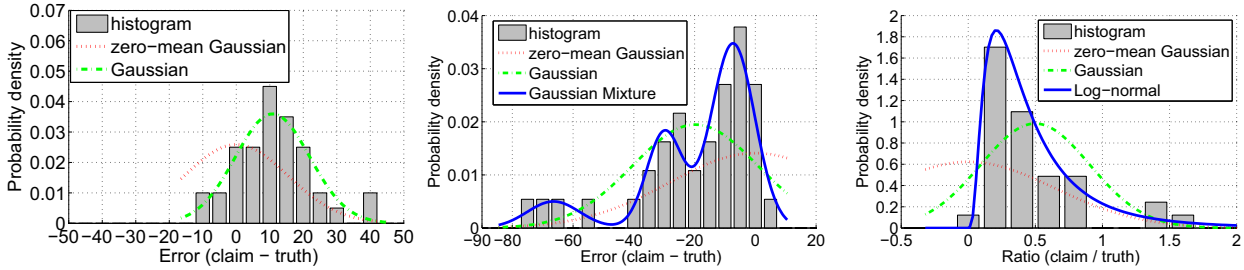
To verify this hypothesis, we plot in Fig. 1(c) the ratios of the same participants whose errors are shown in Fig. 1(b). We observe that the ratios indeed vary in a relatively small range and most participants' ratios do not

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering



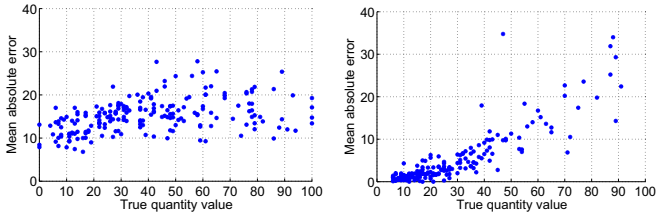
(a) Errors in percentage annotation tasks (b) Errors in object counting tasks (c) Ratios in object counting tasks

Fig. 1. (a) Example participants' errors in percentage annotation tasks. (b) Example participants' errors in object counting tasks. (c) Example participants' ratios (whose errors are shown in (b)) in object counting tasks.



(a) Error fitting in percentage anno. tasks (b) Error fitting in object counting tasks (c) Ratio fitting in object counting tasks

Fig. 2. (a) Fitting a participant's errors in percentage annotation tasks. (b) Fitting a participant's errors in object counting tasks. (c) Fitting a participant's ratios (whose errors are shown in (b)) in object counting tasks.



(a) Room occupancy dataset (b) Vehicle counting dataset

Fig. 3. Mean absolute error in claims vs. z on (a) the room occupancy dataset, and (b) the vehicle counting dataset.

exhibit multiple modes. Since the ratios w_{ij} are always positive, we consider to use the log-normal distribution [6] to model them, whose definition domain is also positive. Moreover, the log-normal distribution can well handle skewness in distributions. We illustrate in Fig. 2(c) example model fitting results using 1) zero-mean Gaussian, 2) Gaussian, and 3) log-normal distributions, fitted to the ratios of the same participant, whose errors are shown in Fig. 2(b). It is clear that the log-normal distribution closely matches the histogram of the ratios.

We then test the goodness-of-fit of these distributions using the K-S test. Among all the participants in the three object counting tasks, 84.2%, 30.0%, and 0% participants pass the K-S test when we fit their ratios using a log-normal, a Gaussian, and a zero-mean Gaussian distribution respectively. We thus use a log-normal distribution to model participants' ratios in object counting tasks.

3.3 Models

The observations above motivate us to design different models that capture different relationships between crowdsourced claims and latent truths in different classes of tasks. In particular, we design the following two general models:

1) **QTF - additive (QTF-A) model.** In QTF-A, we model that a crowdsourced claim x_{ij} is generated according to

$$x_{ij} = z_j + e_{ij}, \tag{1}$$

where e_{ij} is an error term. e_{ij} with respect to u_i follows a parametric distribution, e.g., Gaussian. The latent truth z_j and participants' abilities are modeled as parameters of this distribution. This model means, a participant adds an error term e_{ij} to the latent truth z_j to generate the claim x_{ij}, and the value of the latent truth z_j does not impact the value of the error e_{ij}. QTF-A is thus appropriate for truth discovery in tasks (such as percentage annotation) where errors in the claims are approximately independent of the values of latent truths (an example is shown in Fig. 3(a)).

2) **QTF - multiplicative (QTF-M) model.** In QTF-M, we model that a crowdsourced claim x_{ij} is generated according to

$$x_{ij} = w_{ij} \times z_j, \tag{2}$$

where w_{ij} is a ratio term. w_{ij} with respect to u_i follows a parametric distribution, e.g., log-normal. The latent truth z_j and participants' abilities are modeled as parameters of this distribution. This model means, a participant multiplies a ratio term w_{ij} to the latent truth z_j to generate the claim x_{ij}. (2) can be rewritten as $x_{ij} = z_j + (w_{ij} - 1)z_j$. Comparing this expression with (1), the error in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

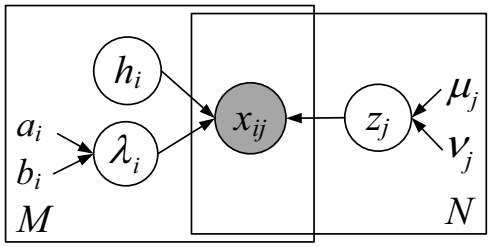


Fig. 4. Structure of the Quantitative Truth Finder (QTF).

QTF-M is given by $e_{ij} = (w_{ij} - 1)z_j$. Consequently, when z_j is small (large), the corresponding error e_{ij} is also small (large). QTF-M is thus appropriate for truth discovery in tasks (such as object counting) where errors in the claims are positively impacted by the values of latent truths (an example is shown in Fig. 3(b)).

In Sections 4 and 5, we detail these two models.

4 QTF-A Model

In this section, we present the design of the QTF-A model. We also present an efficient algorithm to infer its parameters. We illustrate its structure in Fig. 4, where each node represents a random variable. Dark shaded nodes indicate observed variables, and light nodes represent latent variables and model parameters.

4.1 Model Design

In QTF-A, we model the following variables: 1) a target's true quantity value z_j, 2) a participant's quantity estimation bias h_i and confidence λ_i, and 3) a participant's claim x_{ij}. Among these variables, only x_{ij} is observed and z_j is the parameter of interest that to be inferred.

4.1.1 Overview

In QTF-A, we use an additive model to capture the relationship between a crowdsourced claim and the latent truth. In particular, we model that a participant u_i's errors e_{ij} in her claims follow a Gaussian distribution whose probability density function is given by

$$p(e_{ij}|h_i, \lambda_i) = \mathcal{N}(e_{ij}|h_i, 1/\lambda_i) = \sqrt{\frac{\lambda_i}{2\pi}} \exp\left[-\frac{\lambda_i}{2}(e_{ij} - h_i)^2\right], \quad (3)$$

where h_i and $1/\lambda_i$ are the mean and the variance of e_{ij}. h_i and λ_i thus represent the bias and the confidence of participant u_i in quantity estimation. λ_i is also known as the precision parameter (i.e., the inverse of the variance) of a Gaussian distribution.

Since $e_{ij} = x_{ij} - z_j$, after a transformation of variables, we have the conditional distribution of the crowdsourced claim x_{ij}, given the true quantity value z_j and a participant's ability parameters (h_i, λ_i) as

$$p(x_{ij}|z_j, h_i, \lambda_i) = \mathcal{N}(x_{ij}|z_j + h_i, 1/\lambda_i) = \sqrt{\frac{\lambda_i}{2\pi}} \exp\left[-\frac{\lambda_i}{2}(x_{ij} - z_j - h_i)^2\right]. \quad (4)$$

It is observed that, in (4), the latent true value z_j and the participant's bias h_i impact the mean claim value (i.e., the claim is centered at $z_j + h_i$ instead of z_j); the participant's confidence λ_i impacts the precision of the claim. As such, the impact from the true quantity value z_j and the participant's ability (h_i, λ_i) are both considered when modeling the generation of a claim x_{ij}.

We also observe from (4) that $p(x_{ij}|z_j, h_i, \lambda_i) = p(x_{ij}|z_j + \epsilon, h_i - \epsilon, \lambda_i)$, where ϵ is an arbitrary number. That is to say, if $(z_j^*, h_i^*, \lambda_i^*)$ results in the maximum likelihood, then $(z_j^* + \epsilon, h_i^* - \epsilon, \lambda_i^*)$ results in the same maximum likelihood. It is therefore important to impose an *informative* prior distribution on z_j, h_i, or both z_j and h_i in order to restrict the freedom of the model. This observation also implies that we should use the maximum a posteriori (MAP) estimation [3] to infer model parameters. Otherwise, infinite numbers of (z_j, h_i, λ_i) tuples will be the optimal solutions using the maximum likelihood estimation (MLE).

In particular, we choose to impose an informative prior on z_j for computational convenience and for not to introduce unnecessary errors. This is because a prior estimate of z_j can be readily obtained from the observed claims x_{ij} (such as taking the median), while a prior estimate of h_i need to be obtained based on both observed x_{ij} and the estimates of unobserved z_j. Moreover, imposing a prior on h_i will make the model inference procedure much more complex, and the initialization of many variables (i.e., z_j and λ_i) is required.

In the following, we detail the model components.

4.1.2 Latent Truth

We model that the target quantity in each task has a latent true value z_j, which is generated from a Gaussian distribution as

$$p(z_j|\mu_j, \nu_j) = \mathcal{N}(z_j|\mu_j, 1/\nu_j) = \sqrt{\frac{\nu_j}{2\pi}} \exp\left[-\frac{\nu_j}{2}(z_j - \mu_j)^2\right],$$

where μ_j and ν_j are hyperparameters, encoding our prior belief on the mean and the precision of z_j. A small ν_j represents low confidence. The choice of using a Gaussian distribution to model $p(z_j)$ is because such a distribution is the conjugate prior of the mean parameter of a Gaussian distribution given the data likelihood (4).

4.1.3 Participant's Bias and Confidence

We model that each participant's quantity estimation ability is characterized by a pair of (bias, confidence) parameter, which is denoted as (h_i, λ_i). A small $|h_i|$ means a participant is almost unbiased (i.e., the mean of her errors is close to 0). A large λ_i means a participant is very confident (i.e., her errors spread in a small range).

To mitigate numerical problems in computation, we also impose a prior distribution on λ_i as

$$p(\lambda_i|a_i, b_i) = \mathsf{Gamma}(\lambda_i|a_i, b_i) = \frac{1}{\Gamma(a_i)} b_i^{a_i} \lambda_i^{a_i - 1} \exp(-b_i \lambda_i), \quad (5)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

where a_i, b_i are hyperparameters of a Gamma distribution, encoding our prior belief on λ_i. The choice of using a Gamma distribution to model $p(\lambda_i)$ is because such a distribution is the conjugate prior of the precision parameter of a Gaussian distribution given the data likelihood (4). We impose a prior on λ_i because the denominator in (7) that will be shown later can easily be 0 without any prior.

4.1.4 Crowdsourced Claim

Finally, we model the conditional probability that a participant u_i makes a claim x_{ij} on a quantity z_j, given z_j and her estimation ability (h_i, λ_i) as in (4).

4.2 Inference Algorithm

In this section, we discuss how to perform model inference to estimate the true quantity values z_j and participants' ability parameters (h_i, λ_i) in QTF-A.

In particular, we treat $\boldsymbol{\theta} = \{z_j, h_i, \lambda_i\}$ as model parameters. We can write out the joint probability of observing the claims $\mathbf{X} = \{x_{ij}\}$ and model parameters $\boldsymbol{\theta}$ as

$$\begin{aligned} & p(\mathbf{X}, \boldsymbol{\theta}) = p(\boldsymbol{\theta}) p(\mathbf{X}|\boldsymbol{\theta}) \\ = & \prod_j p(z_j|\mu_j, \nu_j) \prod_i p(\lambda_i|a_i, b_i) \prod_j \prod_{i \in \mathcal{U}_j} p(x_{ij}|z_j, h_i, \lambda_i) \\ = & \prod_j \mathcal{N}(z_j|\mu_j, 1/\nu_j) \prod_i \mathsf{Gamma}(\lambda_i|a_i, b_i) \\ & \times \prod_j \prod_{i \in \mathcal{U}_j} \mathcal{N}(x_{ij}|z_j + h_i, 1/\lambda_i), \end{aligned}$$

where $\mathcal{U}_j$ is the set of participants who make a claim on z_j.

We can then compute the logarithm of $p(\mathbf{X}, \boldsymbol{\theta})$ as

$$\begin{aligned} \ln p(\mathbf{X}, \boldsymbol{\theta}) = & \sum_j \sum_{i \in \mathcal{U}_j} \Big[\frac{1}{2} \ln \lambda_i - \frac{\lambda_i}{2} (x_{ij} - z_j - h_i)^2 \Big] \\ & + \sum_j \Big[-\frac{\nu_j}{2} (z_j - \mu_j)^2 \Big] + \sum_i \Big[(a_i - 1) \ln \lambda_i - b_i \lambda_i \Big], \end{aligned}$$

where the constant term is ignored.

We then use the MAP estimation to infer the model parameters $\boldsymbol{\theta}$, which is to maximize $\ln p(\mathbf{X}, \boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$. By setting $\frac{\partial \ln p(\mathbf{X}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0$, we obtain the following system of equations

$$h_i = \frac{\sum_{j \in \mathcal{Z}_i} (x_{ij} - z_j)}{|\mathcal{Z}_i|} \tag{6}$$

$$\lambda_i = \frac{\frac{1}{2} |\mathcal{Z}_i| + a_i - 1}{\frac{1}{2} \sum_{j \in \mathcal{Z}_i} (x_{ij} - z_j - h_i)^2 + b_i} \tag{7}$$

$$z_j = \frac{\nu_j \mu_j + \sum_{i \in \mathcal{U}_j} [\lambda_i (x_{ij} - h_i)]}{\nu_j + \sum_{i \in \mathcal{U}_j} \lambda_i}, \tag{8}$$

where $i = 1, \ldots, M$, $j = 1, \ldots, N$, and $\mathcal{Z}_i$ is the set of quantities that u_i makes a claim on.

We observe that (6)-(8) are not closed-form expressions for model parameters, since they are coupled together (i.e., $h_i = f(z_j)$, $\lambda_i = f(z_j, h_i)$, and $z_j = f(h_i, \lambda_i)$). As a consequence, we initialize z_j and then iterate over (6)-(8) until convergence to obtain the optimal h_i^*, λ_i^*, and z_j^*.

This model inference procedure can be interpreted as iterating between participants' abilities (h_i, λ_i) (according to (6) and (7)) and latent quantity values z_j (according to (8)) until convergence. This is analogous to the model inference procedure in categorical truth discovery methods [21], [30], [35], [37], where iterations are performed between source quality and claim trustworthiness until convergence.

4.2.1 Initialization and Hyperparameter Setting

As median is more robust than mean to outliers, we initialize z_j as the median of the corresponding claims for it. That is,

$$\check{z}_j = \text{median}_{i \in \mathcal{U}_j}(x_{ij}).$$

We set the hyperparameters as

$$\mu_j = \check{z}_j, \nu_j = \frac{1}{[\mathcal{S}_{i \in \mathcal{U}_j}(x_{ij})]^2}, a_i = 1 + 10^{-4},\ b_i = 10^{-4},$$

where $\mathcal{S}$ is a robust scale estimator (also known as the absolute pairwise difference) proposed in [27]. $\mathcal{S}$ is defined as $\mathcal{S} = 1.1926\ \text{median}_c\{\text{median}_d |x_c - x_d|\}$. That is, for each c, we compute the median of $\{|x_c - x_d|\}$ with respect to all d. This yields $C = |\{x_c\}|$ numbers, the median of which gives the final estimate $\mathcal{S}$. $\mathcal{S}$ has the advantages of both robustness and superior efficiency on contaminated data, and thus it is less affected by outliers compared with the standard deviation. As ν_j incorporates our prior belief on the precision of z_j and $\mathcal{S}$ is analogous to the standard deviation, we set $\nu_j = \frac{1}{[\mathcal{S}_{i \in \mathcal{U}_j}(x_{ij})]^2}$. μ_j and ν_j are thus informative prior hyperparamters for z_j, complying with the discussion in Section 4.1.1. a_i and b_i are set to small constants in order to avoid numerical problems in computation.

4.3 Model Analysis

In this section, we discuss some properties of the proposed QTF-A model.

1) **QTF-A uses more appropriate parameters to capture participants' abilities in quantitative crowdsourcing tasks.** In particular, QTF-A uses the bias h_i and the confidence λ_i to capture participants' abilities. A small $|h_i|$ means a participant is almost unbiased (i.e., the mean of her errors is close to 0). A large λ_i means a participant is very confident (i.e., her errors spread in a small range). These two parameters play an important role in truth discovery as in (8), where each claim x_{ij} should be rectified by subtracting the participant's bias h_i and then be weighted by the participant's confidence λ_i. In this way, even if none of the claims x_{ij} is accurate enough, the aggregated z_j may be quite close to the true value. In contrast, categorical truth discovery methods [19], [21], [24], [30], [33]–[35] rely on the agreement among claims and use the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

accuracy to capture participants' abilities. These methods are thus not effective or even not applicable for quantitative crowdsourcing applications, where it is not uncommon that each participant provides a different claim in a task, and the accuracy is small for any participant.

2) **QTF-A naturally incorporates the similarity between crowdsourced claims and the latent truths.** It is observed from (4) that the conditional probability of observing x_{ij} depends on the difference between $x_{ij} - h_i$ (after bias correction) and z_j. The more similar $x_{ij} - h_i$ is to z_j, the more likely we can observe such a x_{ij}. As a consequence, we can more accurately recover z_j. In contrast, in most categorical truth discovery methods [19], [21], [24], [30], [33], [34], the probability of observing a claim is the same for any claim that differs from the truth. The Truth Finder proposed in [35] considers the similarity between claims, but not the similarity between crowdsourced claims and latent truths.
3) **QTF-A can easily incorporate prior belief.** We can easily incorporate our prior belief on the latent truth z_j through μ_j and ν_j for more accurate truth discovery. Similarly, we can encode our prior belief on λ_i through a_i and b_i.

5 QTF-M Model

In this section, we present the design of the QTF-M model. We also present an algorithm to infer its parameters. Its structure is also illustrated in Fig. 4, as after derivation, QTF-M differs from QTF-A in the respective distributions and the meaning of certain parameters, but not the model structure.

5.1 Model Design

In QTF-M, we model the following variables: 1) a target's true quantity value z_j, 2) a participant's quantity estimation bias h_i and confidence λ_i in the logarithm domain, and 3) a participant's claim x_{ij}. Among these variables, only x_{ij} is observed and z_j is the parameter of interest that to be inferred.

5.1.1 Overview

In QTF-M, we use a multiplicative model to capture the relationship between a crowdsourced claim and the latent truth. In particular, we model that a participant u_i's ratios w_{ij} in her claims follow a log-normal distribution [6] whose probability density function is given by

$$\begin{aligned} p(w_{ij}|h_i,\lambda_i) &= \mathsf{log\text{-}normal}(w_{ij}|h_i, 1/\lambda_i) \\ &= \frac{1}{w_{ij}}\sqrt{\frac{\lambda_i}{2\pi}}\exp\left[-\frac{\lambda_i}{2}(\ln w_{ij} - h_i)^2\right], \end{aligned} \tag{9}$$

where h_i and $1/\lambda_i$ are the mean and the variance of $\ln w_{ij}$. h_i and λ_i thus represent the bias and the confidence of u_i in quantity estimation in the logarithm domain.

It can be shown that the distribution of $\ln w_{ij}$ then follows a Gaussian distribution [6] as

$$p(\ln w_{ij}|h_i,\lambda_i) = \mathcal{N}(\ln w_{ij}|h_i, 1/\lambda_i). \tag{10}$$

Since $\ln w_{ij} = \ln(x_{ij}/z_j) = \ln x_{ij} - \ln z_j$, after a transformation of variables, we have the conditional distribution of the logarithm of the crowdsourced claim $\ln x_{ij}$, given the true quantity value z_j and a participant's ability parameters (h_i, λ_i) in the logarithm domain as

$$\begin{aligned} p(\ln x_{ij}|z_j,h_i,\lambda_i) &= \mathcal{N}(\ln x_{ij}|\ln z_j + h_i, 1/\lambda_i) \\ =&\sqrt{\frac{\lambda_i}{2\pi}}\exp\left[-\frac{\lambda_i}{2}(\ln x_{ij} - \ln z_j - h_i)^2\right]. \end{aligned} \tag{11}$$

The choice of modeling the distribution of $\ln x_{ij}$ instead of x_{ij} is for computational convenience. To avoid the impact of claims with zero values, we replace such claims by a small positive number 10^{-4}.

In the following, we detail the model components.

5.1.2 Latent Truth

For computational convenience, we assign a prior distribution on $\ln z_j$ instead of z_j. In particular, we model that the logarithm of the target quantity in each task has a latent true value $\ln z_j$, which is generated from a Gaussian distribution as

$$\begin{aligned} p(\ln z_j|\mu_j,\nu_j) &= \mathcal{N}(\ln z_j|\mu_j, 1/\nu_j) \\ &= \sqrt{\frac{\nu_j}{2\pi}}\exp\left[-\frac{\nu_j}{2}(\ln z_j - \mu_j)^2\right], \end{aligned}$$

where μ_j and ν_j are hyperparameters, encoding our prior belief on the mean and the precision of $\ln z_j$.

5.1.3 Participant's Bias and Confidence

We model that each participant's quantity estimation ability is characterized by a pair of (bias, confidence) parameter in the logarithm domain, which is denoted as (h_i, λ_i). We also impose a prior distribution on each λ_i as that in (5).

5.1.4 Crowdsourced Claim

Finally, we model the conditional probability that a participant u_i makes a claim $\ln x_{ij}$ on a quantity z_j, given z_j and her estimation ability (h_i, λ_i) as in (11).

5.2 Inference Algorithm

In this section, we discuss how to perform model inference to estimate the true quantity values z_j and participants' ability parameters (h_i, λ_i) in QTF-M. We again treat $\boldsymbol{\theta} = \{z_j, h_i, \lambda_i\}$ as model parameters.

Similar to that in the QTF-A model, we can obtain

$$\begin{aligned} \ln p(\mathbf{X},\boldsymbol{\theta}) = &\sum_j \sum_{i\in\mathcal{U}_j}\left[\frac{\ln\lambda_i}{2} - \frac{\lambda_i}{2}(\ln x_{ij} - \ln z_j - h_i)^2\right] \\ &+ \sum_j\left[\frac{\ln\nu_j}{2} - \frac{\nu_j}{2}(\ln z_j - \mu_j)^2\right] + \sum_i (a_i - 1)\ln\lambda_i - b_i\lambda_i, \end{aligned}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

where the constant term is ignored.

By maximizing $\ln p(\mathbf{X}, \boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$, we obtain the following system of equations

$$h_i = \frac{\sum_{j \in \mathcal{Z}_i} (\ln x_{ij} - \ln z_j)}{|\mathcal{Z}_i|} \tag{12}$$

$$\lambda_i = \frac{\frac{1}{2}|\mathcal{Z}_i| + a_i - 1}{\frac{1}{2}\sum_{j \in \mathcal{Z}_i} (\ln x_{ij} - \ln z_j - h_i)^2 + b_i} \tag{13}$$

$$\ln z_j = \frac{\nu_j \mu_j + \sum_{i \in \mathcal{U}_j} [\lambda_i (\ln x_{ij} - h_i)]}{\nu_j + \sum_{i \in \mathcal{U}_j} \lambda_i}. \tag{14}$$

Note that, we treat $\ln z_j$ instead of z_j as a variable in the above system of equations for computational convenience. We initialize $\ln z_j$ and then iterate over (12)-(14) until convergence to obtain h_i^*, λ_i^*, and $\ln z_j^*$. The estimated truth z_j^* is then given by $z_j^* = \exp(\ln z_j^*)$.

This inference procedure can be interpreted similarly as that in Section 4.2, which is to iterate between participants' abilities in the logarithm domain (h_i, λ_i) (according to (12) and (13)) and the logarithm of latent quantity values $\ln z_j$ (according to (14)) until convergence.

5.2.1 Initialization and Hyperparameter Setting

We initialize $\ln z_j$ as the median of the logarithm of the corresponding claims for it. That is,

$$\ln \check{z}_j = \text{median}_{i \in \mathcal{U}_j} (\ln x_{ij}).$$

We set the hyperparameters as

$$\mu_j = \ln \check{z}_j, \nu_j = \frac{1}{[\mathcal{S}_{i \in \mathcal{U}_j} (\ln x_{ij})]^2}, a_i = 1 + 10^{-4},\ b_i = 10^{-4},$$

where the robust scale estimator $\mathcal{S}$ has been introduced in Section 4.2.1.

5.3 Model Analysis

The QTF-M model holds similar properties as the QTF-A model, but these properties are with respect to the logarithm domain.

6 Experiments

In this section, we present our experimental results to demonstrate the effectiveness of these QTFs compared with other state-of-the-art methods. We also evaluate the time efficiency of these methods.

6.1 Methods in Comparison

We compare the following methods for truth discovery in quantitative crowdsourcing applications.

1) `MV`: the widely used majority voting method.
2) `AvgLog`: the AverageLog method proposed in [21].
3) `Invest`: the Investment method proposed in [21].
4) `TF`: the Truth Finder method proposed in [35]. We set the implication score between x_{ij} and $x_{i'j}$ as $\exp(-|x_{ij} - x_{i'j}|/20)$, which increases as the difference between x_{ij} and $x_{i'j}$ decreases. It means that similar claims support each other.
5) `LCA`: the Simple Latent Credibility Analysis method proposed in [22].
6) `Median`: using the median (more robust to outliers than the mean) of crowdsourced claims as the estimated truth.
7) `GTM`: the Gaussian Truth Model proposed in [38].
8) `TBP`: the Truth, Bias, and Precision model proposed in [20].
9) `QTF-A`: the QTF-A model proposed in Section 4.
10) `QTF-M`: the QTF-M model proposed in Section 5.

Among these methods, the first five are designed for truth discovery in categorical applications, while the last five are for quantitative applications. The first five methods aim to find the best possible truth from the set $\mathcal{C}_j$ of distinct claims associated with a target quantity, i.e., $\mathcal{C}_j = \{x_{ij} | i \in \mathcal{U}_j\}$. For example, if for the jth target quantity, the claims are $x_{1j} = 10$, $x_{2j} = 8$, $x_{3j} = 10$, and $x_{4j} = 20$, we then have $\mathcal{C}_j = \{8, 10, 20\}$. The best possible truth can only be one out of the three values in $\mathcal{C}_j$. In contrast, the last five methods estimate the truth by numerically manipulating the claims (they round their outputs to the closest integers). They may output a value that is not in $\mathcal{C}_j$. In case no participant makes an accurate claim, the last five methods may still be able to output an accurate estimated truth.

`AvgLog`, `Invest`, and `TF` are iterative methods that propagate the reliabilities of information sources (e.g., crowd participants) to the trustworthiness of claims and then back. `LCA` models the latent categorical truth and the source quality in terms of the probability that a source makes an honest, accurate claim.

`GTM` models the latent quantitative truth and the source quality (in terms of the variance parameter of a Gaussian distribution). `GTM` first performs outlier detection and data normalization. After model inference, the discovered truth is then scaled back. In particular, a claim x_{ij} is normalized as $x'_{ij} = (x_{ij} - \tilde{z}_j)/\tilde{\sigma}_j$, where $\tilde{z}_j$ is the prior of z_j (which is set to the median of $\{x_{ij} | i \in \mathcal{U}_j\}$) and $\tilde{\sigma}_j$ is the standard deviation of $\{x_{ij} | i \in \mathcal{U}_j\}$ excluding outliers. As the normalization is with respect to a target quantity z_j, it cannot make the normalized claims with respect to a participant u_i Gaussian distributed (while the claims of u_i are modeled as Gaussian distributed). Therefore, there may exist model mismatch. Moreover, such a normalization cannot well tackle the case where the errors in the claims are positively impacted by the latent truths, as the claims are subtracted by, but not divided by the estimated truth. Furthermore, `GTM` does not consider the source bias, but assumes that the claims are centered around the corresponding latent truth. However, it is observed in Section 3.2 that biases are common in participants' claims.

`TBP` models latent quantitative truths, task difficulty levels, and biases and precisions of crowd participants under different task difficulty levels. It can be shown that, in `TBP`, the probability of observing a claim follows a Gaussian mixture distribution, which can well approximate any continuous distribution by tuning its

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

parameters [3]. Therefore, TBP can well model the errors in crowdsourced claims in various classes of tasks. However, the need to find the optimal number K of model components for each specific dataset in an unsupervised manner is non-trivial. TBP thus uses default settings for truth discovery, which sets $K = 2$ and $K = 3$ difficulty levels for percentage annotation and object counting tasks. We also use such default settings in our experiments. Moreover, as TBP needs to infer a large number of model parameters, overfitting may occur when the number of claims per participant is small.

QTF-A and QTF-M proposed in this paper consider different models that capture different relationships between crowdsourced claims and latent truths in different classes of quantitative tasks. QTF-A is appropriate for truth discovery in tasks (such as percentage annotation) where the errors in crowdsourced claims are approximately independent of the values of latent truths, and can be well explained by an additive model. QTF-M is appropriate for truth discovery in tasks (such as object counting) where the errors in crowdsourced claims are positively impacted by the values of latent truths, and can be well explained by a multiplicative model.

Compared with GTM, QTF-A and QTF-M do not involve improper data normalization, they use robust scale estimation to deal with outliers, and they model participants' biases while GTM does not. Compared with TBP, QTF-A and QTF-M do not need to find the optimal number of model components, they have fewer parameters to infer, and they are also much easier to implement.

6.2 Evaluation Metric

RMSE: We use the root mean square error (RMSE) to evaluate the effectiveness of truth discovery methods. It takes into account both the mean and the standard deviation of the estimation errors. It is defined as

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_j (\hat{z}_j - z_j)^2}.$$

The RMSE penalizes larger errors more and a smaller RMSE indicates better performance.

MAPE: We also use the mean absolute percentage error (MAPE) to evaluate the effectiveness of truth discovery methods, since in some scenarios, the relative accuracy is of more interest. It is defined as

$$\text{MAPE} = \frac{1}{N}\sum_j \left|\frac{\hat{z}_j - z_j}{z_j}\right|.$$

A smaller MAPE indicates better performance. Note that, a small RMSE does not necessarily imply a small MAPE. For example, when $z_j = 50$ and $\hat{z}_j = 52$, the resulting RMSE is 2 and the MAPE is 0.04. When $z_j = 5$ and $\hat{z}_j = 6$, the resulting RMSE is 1 (smaller than 2) but the MAPE is 0.2 (larger than 0.04).

CPU time: We use the CPU time to evaluate the time efficiency of an algorithm. A shorter CPU time implies a faster algorithm.

All the experiments are performed using Matlab on a desktop with 4GB RAM and Intel Core i3-2120 CPU. Each experiment is performed 10 times by randomly sampling the desired number of participants and tasks, unless all of them are sampled. We only retain tasks with at least 3 crowdsourced claims and retain participants with at least 10 claims in each experiment.

In addition to using real-world datasets as listed in Table 2 to evaluate different methods, we also use synthetic datasets to examine the robustness of them. The details of these synthetic datasets will be presented before we describe the corresponding experimental results.

6.3 Effectiveness

In this section, we examine the effectiveness of various truth discovery methods.

6.3.1 Crowdsourced Percentage Annotation

Table 3 lists the errors of different methods in percentage annotation tasks, where we set the number of participants per task as 5 (i.e., only a few participants work on a task) and set the number of tasks as 150. As we address the truth discovery problem in quantitative crowdsourcing applications, Median serves as a reasonable baseline. Besides listing out the raw errors of different methods, we also annotate in parentheses the relative error increase or decrease with respect to Median. In particular, "$+p\%$" means that a method's error is $p\%$ larger than that of Median and this method performs worse than Median. "$-q\%$" means that a method's error is $q\%$ smaller than that of Median and this method performs better than Median.

It is observed that all the categorical truth discovery methods, which are MV, AvgLog, Invest, TF, and LCA, perform worse than Median and result in larger errors. These results show that since categorical truth discovery methods rely on the agreement among claims and use the accuracy to capture participants' abilities, they are not effective for quantitative truth discovery. TF performs best among these categorical truth discovery methods. This is because TF assigns implication scores between pair-wise claims, and similar claims support each other. As a result, TF can better discover truths in quantitative tasks than other categorical methods. However, TF still performs much worse than Median.

GTM sometimes performs slightly better, sometimes performs slightly worse, and sometimes have identical performance as Median (may be resulted by numerical rounding). This is because GTM does not model participants' biases, and there may exist mismatch between the model and the distribution of normalized data. TBP performs best in terms of the RMSE on the restaurant occupancy dataset, and performs second best elsewhere. QTF-A performs best in most cases, and it can lead to up to 4.09% smaller MAPE than that of TBP. The RMSEs of QTF-A on these datasets are 5.90% to 10.73% smaller than those of Median, and the MAPEs of QTF-A are

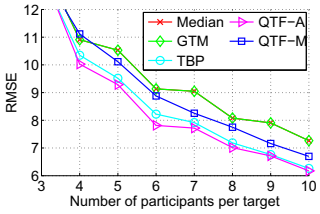
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

TABLE 3
Effectiveness of different methods in truth discovery in percentage annotation tasks (5 participants per task).

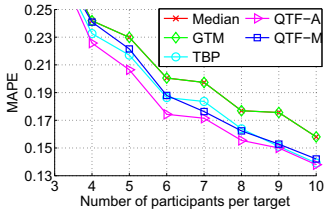
	Room occ		Gym occ		Rest occ	
	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
MV	19.46 (+84.81%)	0.367 (+57.51%)	20.38 (+74.64%)	0.300 (+56.25%)	19.25 (+80.24%)	0.401 (+48.52%)
AvgLog	15.51 (+47.29%)	0.325 (+39.48%)	15.38 (+31.81%)	0.255 (+32.79%)	15.13 (+41.67%)	0.373 (+38.15%)
Invest	16.00 (+51.95%)	0.335 (+43.77%)	17.54 (+50.30%)	0.282 (+46.88%)	14.61 (+36.80%)	0.353 (+30.74%)
TF	12.56 (+19.28%)	0.263 (+12.87%)	12.39 (+6.17%)	0.208 (+8.33%)	12.21 (+14.33%)	0.299 (+10.74%)
LCA	18.86 (+79.11%)	0.425 (+82.40%)	18.66 (+59.90%)	0.286 (+48.96%)	18.79 (+75.94%)	0.439 (+62.59%)
Median	10.53	0.233	11.67	0.192	10.68	0.270
GTM	10.53 (+0.00%)	0.233 (+0.00%)	11.67 (+0.00%)	0.192 (+0.00%)	10.67 (−0.09%)	0.271 (+0.37%)
TBP	9.58 (−9.02%)	0.220 (−5.58%)	10.97 (−6.00%)	0.184 (−4.17%)	**9.96 (−6.74%)**	0.260 (−3.70%)
QTF-A	**9.40 (−10.73%)**	**0.211 (−9.44%)**	**10.73 (−8.05%)**	**0.181 (−5.73%)**	10.05 (−5.90%)	**0.256 (−5.19%)**
QTF-M	10.14 (−3.70%)	0.223 (−4.29%)	11.18 (−4.20%)	0.190 (−1.04%)	10.30 (−3.56%)	0.268 (−0.74%)

TABLE 4
Effectiveness of different methods in truth discovery in object counting tasks (5 participants per task).

	People count		Vehicle count		Bird count	
	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
MV	15.25 (+61.03%)	0.277 (+83.44%)	14.00 (+157.83%)	0.166 (+144.84%)	14.68 (+89.18%)	0.161 (+108.28%)
AvgLog	12.66 (+33.69%)	0.221 (+46.36%)	7.25 (+33.52%)	0.0869 (+28.17%)	7.86 (+1.29%)	0.0842 (+8.93%)
Invest	17.57 (+85.53%)	0.241 (+59.60%)	7.78 (+43.28%)	0.0885 (+30.53%)	8.62 (+11.08%)	0.0870 (+12.55%)
TF	10.87 (+14.78%)	0.160 (+5.96%)	6.13 (+12.89%)	0.0676 (−0.29%)	7.83 (+0.90%)	0.0769 (−0.52%)
LCA	16.91 (+78.56%)	0.271 (+79.47%)	8.01 (+47.51%)	0.112 (+65.19%)	7.56 (−2.58%)	0.0893 (+15.52%)
Median	9.47	0.151	5.43	0.0678	7.76	0.0773
GTM	9.50 (+0.32%)	0.151 (+0.00%)	5.28 (−2.76%)	0.0675 (−0.44%)	7.61 (−1.93%)	0.0765 (−1.03%)
TBP	9.26 (−2.22%)	0.145 (−3.97%)	4.64 (−14.55%)	0.0629 (−7.23%)	7.17 (−7.60%)	0.0712 (−7.89%)
QTF-A	9.38 (−0.95%)	0.153(+1.32%)	4.89 (−9.94%)	0.0664 (−2.06%)	7.26 (−6.44%)	0.0745 (−3.62%)
QTF-M	**8.58 (−9.40%)**	**0.136 (−9.93%)**	**4.35 (−19.89%)**	**0.0585 (−13.72%)**	**6.82 (−12.11%)**	**0.0676(−12.55%)**

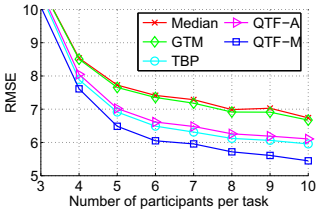


(a) RMSE vs. # pts per task

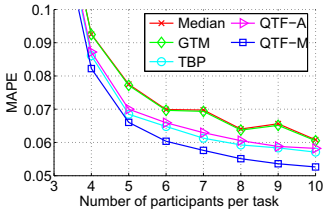


(b) MAPE vs. # pts per task

Fig. 5. Room occupancy dataset: (a) RMSE and (b) MAPE vs. the number of participants per task.



(a) RMSE vs. # pts per task



(b) MAPE vs. # pts per task

Fig. 6. Bird counting dataset: (a) RMSE and (b) MAPE vs. the number of participants per task.

5.19% to 9.44% smaller than those of Median. These results show that QTF-A is very effective for percentage annotation tasks. As QTF-M aims to tackle a different class of tasks, it performs worse than TBP and QTF-A.

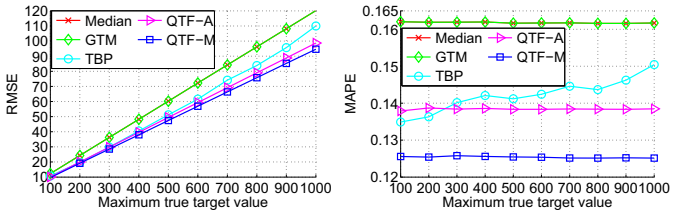
Fig. 5 plots the errors of the five quantitative truth discovery methods on the room occupancy dataset, where we set the number of tasks as 180 and vary the number of participants per task from 3 to 10. It is observed from Fig. 5(a) that the RMSEs of all the methods clearly decrease when more participants join a task. QTF-A performs best in all cases. Its RMSE is 15.0% smaller than that of Median when there are 10 participants per task. Similar results are observed in terms of the MAPE in Fig. 5(b).

6.3.2 Crowdsourced Object Counting

Table 4 lists the errors of different methods in object counting tasks, where we set the number of participants per task as 5 and set the number of tasks as 150. It is observed that all the categorical methods still perform poorly, except that TF and LCA sometimes result in slightly smaller errors than Median. QTF-M performs best on all the three datasets, showing its effectiveness in tackling truth discovery in object counting tasks. QTF-M models the multiplicative relationship between crowdsourced claims and latent truths, which better reflects the underlying data distribution. The RMSEs of QTF-M on these datasets are 9.40% to 19.89% smaller than those of Median, and the MAPEs of QTF-M are 9.93% to 13.72% smaller than those of Median. QTF-M can lead to up to 7.34% (7.00%) smaller RMSE (MAPE) than that of TBP. QTF-A performs worse than TBP and QTF-M, as it aims to tackle a different class of tasks.

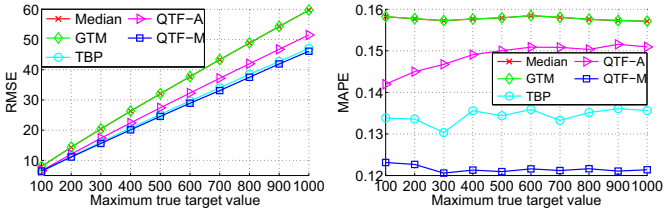
Fig. 6 plots the errors of the five quantitative truth discovery methods on the bird counting dataset, where we set the number of tasks as 180 and vary the number of participants per task from 3 to 10. It is observed from Fig.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering



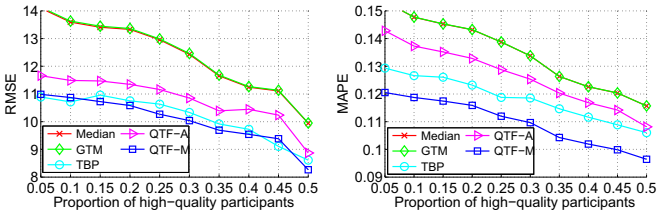
(a) RMSE vs. maximum z (b) MAPE vs. maximum z

Fig. 7. Synthetic dataset (uniform z): (a) RMSE and (b) MAPE vs. the maximum target quantity value.



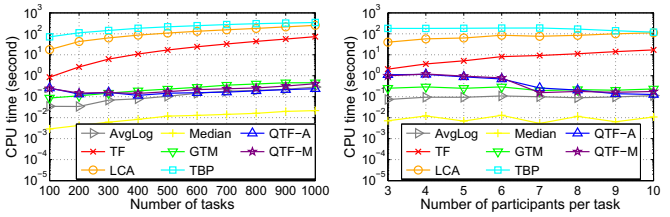
(a) RMSE vs. maximum z (b) MAPE vs. maximum z

Fig. 8. Synthetic dataset (non-uniform z): (a) RMSE and (b) MAPE vs. the maximum target quantity value.



(a) RMSE vs. high-quality rate (b) MAPE vs. high-quality rate

Fig. 9. Synthetic dataset (non-uniform z, max 200): (a) RMSE and (b) MAPE vs. the high-quality rate.



(a) CPU time vs. # of tasks (b) CPU time vs. # of pts per task

Fig. 10. (a) CPU time vs. the total number of tasks (5 participants per task). (b) CPU time vs. the number of participants per task (totally 500 tasks).

6(a) that the RMSEs of all the methods clearly decrease when more participants join a task. Compared with the RMSE of `Median`, when there are 10 participants per task, the RMSE of `GTM` is only 1.0% smaller; the RMSEs of `QTF-A` and `TBP` are 9.4% and 11.7% smaller respectively; the RMSE of `QTF-M` is 19.3% smaller. Similar results are observed in terms of the MAPE in Fig. 6(b).

6.3.3 Simulations

We now examine the robustness of different quantitative truth discovery methods on synthetic datasets. We focus on object counting tasks, as the maximum latent truth value in this class of tasks can be unbound, while that in percentage annotation tasks is bounded by 100.

We simulate crowdsourced claims according to the QTF-M model with $M = 100$ participants and $N = 1000$ target quantities, where participants' abilities are uniformly sampled as $h_i \in [-0.5, 0.5]$ and $\lambda_i \in [3, 35]$. We then enlarge the value of the maximum latent truth (denoted as $\max(z)$) from 100 to 1000. Fig. 7 plots the resulting errors, where we uniformly sample the latent true values z_j from 5 to $\max(z)$. It is observed from Fig. 7(a) that the RMSEs clearly increase when $\max(z)$ increases. `QTF-M` outperforms `QTF-A`, `QTF-A` outperforms `TBP`, and `TBP` outperforms `GTM` and `Median`. The reason that `TBP` performs worse than `QTF-A` may be because when the latent true values are uniformly distributed, the errors in the claims are also close to uniformly distributed, since errors are positively impacted by the true quantity values. In such a scenario, modeling multiple difficulty levels is unnecessary and overfitting may be resulted. The RMSE of `QTF-M` is 21.2% smaller than that of `Median` when $\max(z) = 1000$. In terms of the MAPE, we observe a different trend in Fig. 7(b). The MAPEs of different methods, except that of `TBP`, are relatively stable regardless of $\max(z)$. This means that the errors of these methods, except `TBP`, are absolutely larger but proportionally stable when $\max(z)$ enlarges. The MAPE of `QTF-M` is around 22% smaller than that of `Median` in all cases.

Fig. 8 plots the results when we non-uniformly sample the latent true values. In particular, we uniformly sample $\ln z_j$ from $\ln 5$ to $\ln \max(z)$. As a result, the errors are skewed and show obvious mode(s). `TBP` is thus expected to perform better. We observe from Fig. 8(a) that, `TBP` indeed performs better than `QTF-A` and its RMSEs are close to those of `QTF-M`. In Fig. 8(b), it is observed that `TBP` outperforms `QTF-A`, but it performs worse than `QTF-M`, in terms of the MAPE.

We then examine how the errors of different methods change with respect to the proportion ρ of high-quality participants (we term ρ as the high-quality rate). We define a high-quality participant as one whose $h_i \in [-0.05, 0.05]$ and $\lambda_i \in [30, 35]$. Other participants have $h_i \in [-0.5, 0.5]$ and $\lambda_i \in [3, 35]$. The latent truths z_j take values in $[5, 200]$ non-uniformly. It is observed from Fig. 9 that the errors of all the methods decrease when the high-quality rate ρ increases. `QTF-M` again performs best in most cases. When the high-quality rate is 0.05, the RMSE (MAPE) of `QTF-M` is 22.1% (21.4%) smaller than that of `Median`. When the high-quality rate is 0.5, the RMSE (MAPE) of `QTF-M` is 16.9% (16.6%) smaller than that of `Median`.

These results demonstrate that `QTF-M` is not only more effective but also more robust than other methods for truth discovery in object counting tasks.

6.4 Efficiency

In this section, we examine the efficiency of different methods in terms of the CPU time. In order not to make the figure cluttered, we only plot out the CPU time of 8

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535383, IEEE Transactions on Knowledge and Data Engineering

methods (excluding two ineffective methods which are MV and Invest). From Fig. 10, it is observed that the CPU time of most methods generally increases when the total number of tasks increases, and it also increases when the number of participants per task increases. The sudden decrease of the CPU time of QTF-A (QTF-M) at some point may be because they converge faster when a sufficient number of claims are available.

TBP is the most time-consuming, followed by LCA and TF. TBP needs to estimate several pairs of ability parameters per participant, in addition to all the latent truths. LCA needs to evaluate the probability that each distinct claim is the truth for each target. TF needs to assign pairwise implication scores between distinct claims for each target. These operations are time-consuming. Median is the most time efficient, as it is a model-less method.

The CPU time curves of QTF-A and QTF-M are similar, and they reside in the middle of all the CPU time curves. QTF-A and QTF-M can finish processing 1000 tasks in around 1.5 seconds, while TBP can finish such a job in around 350 seconds (232x slower). As QTF-A (QTF-M) is the most effective on a class of tasks that it aims to handle, these QTFs achieve good tradeoffs between effectiveness and efficiency.

7 Discussion and Future Work

Handling other classes of quantitative tasks. Percentage annotation and object counting that investigated in this paper are two representative classes of quantitative crowdsourcing tasks. However, they do not form the complete set of quantitative tasks. Nevertheless, QTF-A and QTF-M are quite general models (with general design principles). They can handle truth discovery in other classes of quantitative tasks as well, as long as the property of a new class of tasks matches respective model's assumptions.

Size of the crowd. Statistical models need sufficient data to work well and thus our proposed QTF-A and QTF-M models may fail if the size of the crowd is small. It is thus useful to investigate the Cramer-Rao lower bound (CRLB) [5], [31] and to examine how the lower bound on the estimation error changes with respect to different sizes of the crowd.

Handling other human factors. In addition to our modeled ability parameters (h_i, λ_i), there are other human factors such as *copying* (copy others' claims) and *lying* (intentionally provide erroneous claims) that can impact the crowdsourced claims. It is thus interesting to extend our models to accommodate these human factors in the future.

8 Conclusion

In this paper, we propose two models, i.e., QTF-A and QTF-M, for truth discovery in quantitative crowdsourcing applications. These QTFs are unsupervised, and they use the bias and the confidence instead of the accuracy to capture a participant's ability in quantity estimation. QTF-A models an additive relationship between crowdsourced claims and latent truths. It is suitable for truth discovery in tasks where errors in the claims are approximately independent of the values of latent truths. QTF-M models a multiplicative relationship between crowdsourced claims and latent truths. It is suitable for tasks where errors in the claims are positively impacted by the values of latent truths. These QTFs are shown to be more effective than other state-of-the-art methods for quantitative truth discovery in particular domains. They are also quite efficient and are easy to implement.

Acknowledgements

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

References

[1] Y. Baba and H. Kashima. Statistical quality estimation for general crowdsourcing tasks. In *KDD*, pages 554–562. ACM, 2013.
[2] M. S. Bernstein et al. Soylent: a word processor with a crowd inside. In *UIST*, pages 313–322. ACM, 2010.
[3] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*. Springer New York, 2006.
[4] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *CVPR*, pages 1–7. IEEE, 2008.
[5] H. Cramér. *Mathematical methods of statistics*, volume 9. Princeton university press, 1999.
[6] E. L. Crow and K. Shimizu. *Lognormal distributions: Theory and applications*, volume 88. Dekker New York, 1988.
[7] W. W. Daniel et al. *Applied nonparametric statistics*. PWS-Kent Boston, 1990.
[8] A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied Statistics*, pages 20–28, 1979.
[9] P. G. Ipeirotis. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM Magazine for Students*, 17(2):16–21, 2010.
[10] A. Khan et al. Occupancy monitoring using environmental & context sensors and a hierarchical analysis framework. In *BuildSys*. ACM, 2014.
[11] A. Kittur, E. H. Chi, and B. Suh. Crowdsourcing user studies with mechanical turk. In *CHI*, pages 453–456. ACM, 2008.
[12] A. Kittur et al. Crowdforge: Crowdsourcing complex work. In *UIST*, pages 43–52. ACM, 2011.
[13] Y. Li, J. Gao, C. Meng, Q. Li, L. Su, B. Zhao, W. Fan, and J. Han. A survey on truth discovery. *arXiv preprint arXiv:1505.02463*, 2015.
[14] G. Little et al. Exploring iterative and parallel human computation processes. In *Proceedings of the ACM SIGKDD workshop on human computation*, pages 68–76. ACM, 2010.
[15] E. Lukman. Indonesian voters are crowdsourcing ballot counts to protect against election fraud. https://www.techinasia.com/kawal-suara-indonesia-voters-crowdsourcing-ballot-counts-protect-election-fraud, 2014. [Online; accessed 16-Dec-2015].

[16] Z. Ma and A. B. Chan. Crossing the line: Crowd counting by integer programming with local features. In *CVPR*, pages 2539–2546. IEEE, 2013.

[17] M. Marge et al. Using the amazon mechanical turk for transcription of spoken language. In *ICASSP*, pages 5270–5273. IEEE, 2010.

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

[19] R. W. Ouyang et al. Truth discovery in crowdsourced detection of spatial events. In *CIKM*, pages 461–470. ACM, 2014.

[20] R. W. Ouyang et al. Debiasing crowdsourced quantitative characteristics in local businesses and services. In *IPSN*, pages 190–201. ACM, 2015.

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

[22] J. Pasternack and D. Roth. Latent credibility analysis. In *WWW*, pages 1009–1020. International World Wide Web Conferences Steering Committee, 2013.

[23] A. J. Quinn and B. B. Bederson. Human computation: a survey and taxonomy of a growing field. In *CHI*, pages 1403–1412. ACM, 2011.

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

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

[26] J. Robbins. Crowdsourcing, for the birds. http://www.nytimes.com/2013/08/20/science/earth/crowdsourcing-for-the-birds.html, 2013. [Online; accessed 16-Dec-2015].

[27] P. J. Rousseeuw and C. Croux. Alternatives to the median absolute deviation. *Journal of the American Statistical association*, 88(424):1273–1283, 1993.

[28] T. Sakaki et al. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW*, pages 851–860. ACM, 2010.

[29] F. Sardà-Palomera et al. Fine-scale bird monitoring from light unmanned aircraft systems. *Ibis*, 154(1):177–183, 2012.

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

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

[32] D. Wang et al. Using humans as sensors: An estimation-theoretic perspective. In *IPSN*, pages 35–46. IEEE, 2014.

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

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

[35] X. Yin et al. Truth discovery with multiple conflicting information providers on the web. *IEEE TKDE*, 20(6):796–808, 2008.

[36] O. F. Zaidan and C. Callison-Burch. Crowdsourcing translation: Professional quality from non-professionals. In *HLT*, pages 1220–1229. ACL, 2011.

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

[38] B. Zhao and J. Han. A probabilistic model for estimating real-valued truth from conflicting sources. *QDB*, 2012.

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

Lance M. Kaplan received the B.S. degree with distinction from Duke University, Durham, NC, in 1989 and the M.S. and Ph.D. degrees from the University of Southern California, Los Angeles, in 1991 and 1994, respectively, all in Electrical Engineering. From 1987-1990, Dr. Kaplan worked as a Technical Assistant at the Georgia Tech Research Institute. He held a National Science Foundation Graduate Fellowship and a USC Dean's Merit Fellowship from 1990-1993, and worked as a Research Assistant in the Signal and Image Processing Institute at the University of Southern California from 1993-1994. Then, he worked on staff in the Reconnaissance Systems Department of the Hughes Aircraft Company from 1994-1996. From 1996-2004, he was a member of the faculty in the Department of Engineering and a senior investigator in the Center of Theoretical Studies of Physical Systems (CTSPS) at Clark Atlanta University (CAU), Atlanta, GA. Currently, he is in the Networked Sensing and Fusion branch of the U.S. Army Research Laboratory (ARL). Dr. Kaplan serves as Editor-In-Chief for the IEEE Transactions on Aerospace and Electronic Systems (AES) and he is Vice President of Conference for the International Society of Information Fusion (ISIF). In addition, he also served on the Board of Governors of the IEEE AES Society, 2009-2014, and on the ISIF Board, 2012-2014. He served as Technical Co-Chair (with Neil Gordon) for the 2011 ISIF/IEEE International Conference on Information Fusion in Chicago, IL. From 2004-2014, he served as the Remote Sensing Co-Organizer (with Peter Kahn) for the IEEE Aerospace Conference in Big Sky, MT. He is a three time recipient of the Clark Atlanta University Electrical Engineering Instructional Excellence Award from 1999-2001. Dr. Kaplan has published over 180 technical articles. His current research interests include signal and image processing, information/data fusion, resource management, and network science.

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

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

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

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

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