2017 IEEE 56th Annual Conference on Decision and Control (CDC)
December 12-15, 2017, Melbourne, Australia

Crowdsourcing with Multi-Dimensional Trust and Active Learning

Xiangyang Liu, and John S. Baras

Abstract— **We consider a typical crowdsourcing task that aggregates input from multiple workers as a problem in information fusion. To cope with the issue of noisy and sometimes malicious input from users, trust is used to model workers expertise. We propose a probabilistic model to jointly infer multi-dimensional trust of workers, multi-domain properties of questions, and true labels of questions. Our model is flexible and extensible to incorporate metadata associated with questions. To show that, we further propose two extended models, one of which handles input tasks with real-valued features and the other handles tasks with text features by incorporating topic models. In order to decrease entropies and reduce error rates more quickly with fewer annotations from workers, we further propose strategies for selecting which questions to ask and which workers to assign the questions to based on multi-dimension characteristics of questions and workers trust values in those dimensions. We evaluate our models and algorithms on real-world data sets. These results can be applied for fusion of information from multiple data sources like sensors, human input, machine learning results, or a hybrid of them.**

I. Introduction

In a crowdsourcing task, in order to estimate the true labels of questions, each question is distributed to the open crowd and is answered by a subset of users (or workers). The answers from workers are then aggregated, taking into account the reliability (or knowledge) of workers, to produce final estimates of true labels. Example questions are: image label inference with multiple annotators' input, topic-document pair relevance inference with crowd's judgements, Bayesian network structure learning given experts' partial knowledge, and test grading without knowing the answers. As we noted in our earlier work [1], most past research ignores the multiple domains involved in questions. For example in test grading without golden truth, bio-chemistry questions require knowledge in both biology and chemistry. Some are more related to biology while others are more related to chemistry. Similarly, workers also exhibit such multi-domain characteristics: people have different levels of knowledge in different subjects. These observations motivate our modeling of multi-domain characteristics for both questions and trust in workers' knowledge and the design of principled methods for aggregating knowledge input from various unreliable sources with different expertise in each domain.

In our work initiated with [1], we proposed to model each question by a *concept vector*, which is a real random vector where the value in a particular dimension indicates its relationship of the question with the knowledge or domain corresponding to this dimension. Back to the test grading example, each bio-chemistry question is represented by a two-dimensional hidden concept vector with the first dimension corresponding to chemistry and the second dimension corresponding to biology. So a concept vector $[0.7, 0.3]$ means the question is more related to chemistry. Each worker is also associated with a *trust vector* [1], which is a real random vector with each dimension representing the trustworthiness of the worker's knowledge in the domain associated with this dimension. The goal our work in [1] and continuing in this paper is to better estimate the true labels of question Q by fusing answers from multiple unreliable workers with varying trust values in each of the domains. Note that the concept vectors of questions and the trust vectors of workers are both hidden. In [1] we proposed a probabilistic model that incorporates questions' concept vectors, workers' trust vectors, answers submitted by workers and designed an inference algorithm that jointly estimates true label of questions along with concept vectors and trust vectors. The inference algorithm of [1] is based on a variational approximation of posterior distributions using a factorial distribution family. In addition, we extended [1] the model by incorporating continuously-valued features. In applications where each question is associated with a short text description, each dimension of the concept vector corresponds to a topic. Therefore we further proposed [1] an extended model that integrates topic discovery. In our work we assume that we send the questions to all workers, wait and gather answers from the workers, and update the posterior probability distributions after gathering all the input from workers.

The models used in this paper were first presented in [1] and we extend the work of [1] here first by providing some detailed algorithmic flows and proofs. In addition, we observe that annotations from workers are expensive in nature [2]. To help the crowdsourcing system learn more from fewer answers from workers, we propose in the present paper active learning strategies for selecting question-worker pairs that help reduce entropy and error rates. And this is the major difference and improvement on our work of [1]. We propose strategies for proactively selecting which questions to ask and which workers to solicit answers from, based on the multi-dimensional trust values of workers and multi-dimensional characteristics of questions.

II. Related Work

There are a lot of works on how to leverage trust models to better aggregate information from multiple sources. Conflicts

Research partially supported by grants AFOSR MURI FA-9550-10-1-0573, NSF CNS-1035655, NIST grant 70NANB11H148, and by the National Security Agency.

The authors are with the Institute for Systems Research and the Department of Electrical and Computer Engineering, University of Maryland, College Park, USA. Email: xyliu@umd.edu, baras@umd.edu

between information provided by different sources were used to revise trust in the information [3]. Trust was also used as weights of edges in the sensor network and was integrated into distributed Kalman filtering to more accurately estimate the state of a linear dynamical system in a distributed setting [4]. Local evidence was leveraged to establish local trust bewteen agents in a network and those local trusts were then used to isolate untrustworthy agents during sensor fusion [5].

In the context of crowdsourcing tasks to the open crowd, many works develop models for aggregating unreliable input from multiple sources to more accurately estimate true labels of questions. The authors in [6] combined multiple weak workers' input for constructing a Bayesian network structure assuming each worker is equally trustworthy. Workers' trust was considered to improve accuracy in aggregating answers in [7]–[10].

A model that jointly infers label of image, trust of each labeler and difficulty of image is proposed in [11]. However, they model questions and workers using scalar variables and they use the Expectation-Maximization inference algorithm, which has long been known to suffer from the existence of many local optima. Another work that went a step further based on signal detection theory is [12], where they assume that each question comes with a feature set and models each worker by a multidimensional classifier in an abstract feature space. Our model [1] can handle more general cases without such an assumption and when text information is available for each question, each dimension of a question becomes interpretable. Moreover, it is difficult to find analytical solutions for the posterior distributions of hidden variables in [12]. An approach in the spirit of test theory and item-response theory (IRT) was proposed in [13] and they relied on approximate message-passing for inference. Their model is not as flexible and extensivle as our model [1] because they have to redesign their model to incorporate rich metadata associated with each question.

In cases where labels from workers are costly, either in terms of money or time, active learning plays a role to lower the cost. Examples of the criteria used in active learning are uncertainty sampling [14], minimization of the expected error estimate [15], and reduction of the size of the version space [16]. All these works assume there is a single labeler giving out annotations for questions. A more recent work [17] explores active learning methods in the case of multiple unreliable annotators where they proposed methods for choosing the question and the worker that provides the most useful information. They model workers' trust as a function of the questions' features. However, features might not be available all the time and the coefficients used in the function are not easily interpretable. Differently, in our work, our model works with and without question features and the trust of workers can be directly associated with the topics of questions. Furthermore in the present paper, we provide methods for actively selecting both questions to ask and workers to answer these selected questions by taking advantage of the multi-dimension characteristics of worker trust and questions domains.

III. Problem Definitions

We start with the model of [1]. Assume there are M workers available and N questions whose true labels need to be estimated. We use R_i to denote the true label variable of question i, where $R_i \in \{0, 1\}$. Each question is answered by a subset of workers M_i and we denote the answer of question i given by worker j by $l_{ij} \in \{0, 1\}$. The set of questions answered by worker j is denoted by N_j.

The multi-domain characteristics of question i are represented by a concept vector λ_i, a D-dimensional real-valued random vector, where D is the total number of domains. To simulate a probability distribution, we further require $\lambda_{il} \in [0, 1], l = 1, \ldots, D$ and $\sum_{l=1}^{D} \lambda_{il} = 1$, where λ_{il} denotes the lth dimension of the concept vector. We impose a Dirichlet prior distribution for concept vector λ_i with hyperparameter $\alpha = \{\alpha_l\}_{l=1}^{D}$, where α_l denotes the soft counts that specify which domain a question falls into a priori.

Workers contribute to the estimation of the true label of questions by providing their own guesses. However, workers' inputs may not be reliable and sometimes even malicious. In multi-domain crowdsourcing tasks, different workers may be good at different domains. The multi-dimensional characteristics of a worker is described by a D-dimensional trust vector $\beta_j = \{\beta_{j1}, \ldots, \beta_{jl}, \ldots, \beta_{jD}\}$, where β_{jl} denotes j-th worker's trust value in domain l and it takes either a continuous or a discrete value. In the discrete case, the inference is generally NP-hard and message-passing style algorithms are used. We consider the continuous case only where $\beta_j \in [0, 1]^D, \forall j$. Higher value of β_{jl} indicates that worker j is more trustworthy in domain l. The true value of β_{jl} is usually unknown to the crowdsourcing platform. It has to be estimated from answers provided by workers. We assume a Beta prior distribution for β_{il} with hyper-parameter $\theta = \{\theta_0, \theta_1\}$, where $\theta_0 > 0$ is the soft count for worker j to behave maliciously and $\theta_1 > 0$ is the soft count for worker j to behave reliably. This interpretation resembles the Beta reputation system [18] that models beliefs of workers.

We aim to estimate the true labels of questions and trust vectors of workers from answers provided by workers.

IV. Multi-Domain Crowdsourcing Model

We describe the generating process for the Multi-Domain Crowdsourcing (MDC) Model in this section following [1].

1) For each question $i \in \{1, \ldots, N\}$,
   a) draw the domain distribution $\lambda_i | \alpha \sim \text{Dir}(\alpha)$;
   b) draw domain $C_i | \lambda_i \sim \text{Discrete}(\lambda_i)$;
2) For each question i, draw the true label $R_i \sim \text{Uniform}(0, 1)$;
3) For each worker $j \in \{1, \ldots, M\}$ and domain $l \in \{1, \ldots, D\}$, draw the trust value $\beta_{jl} \sim \text{Beta}(\theta)$;
4) For each question-worker pair (i, j), draw observed answer $l_{ij} \sim F(R_i, \beta_j, C_i)$

In step 1, the domain for question i is then drawn according to a discrete distribution with parameter λ_i, i.e. generating $C_i = l$ with probability λ_{il}. In step 3, we profile each worker

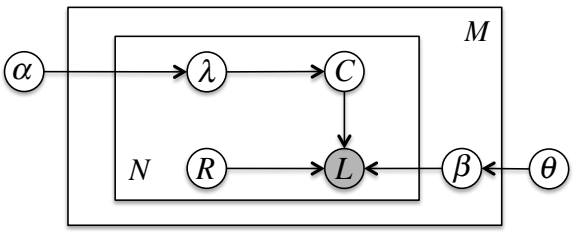


Fig. 1. The graphical model for observed data provided by workers L, multi-domain expertise β, true labels R, domain variables C, and concept vectors λ. M is the total number of workers. N the number of questions. α is the hyperparameter of the Dirichlet prior distribution for λ and θ is the hyperparameter of the Beta prior distribution for β.

by a vector β_j with β_{jl} drawn from a Beta distribution. In step 4, the observed answer of question i provided by worker j is drawn according to an output distribution F, a Bernoulli distribution. We will specify the form of this output distribution in the following paragraph.

The generating process is illustrated in Fig. 1 from [1]. The joint probability distribution is

$$p\left(L,R,\beta,C,\lambda\right)=\prod_{i=1}^{N}p\left(r_i\right)p\left(\lambda_i|\alpha\right)p\left(C_i|\lambda_i\right)\cdot \prod_{j=1}^{M}p\left(\beta_j\right)\prod_{l=1}^{D}p\left(l_{ij}|r_i,C_i=l,\beta_j\right) \tag{1}$$

where N is the total number of questions, M is the total workers, and D is the total number of domains. $p\left(l_{ij}|r_i,C_i=l,\beta_j\right)$ is the output distribution F in Fig. 1 and is the likelihood of worker j's answer given its expertise vector and the domain variable of question i, and the true label. $p\left(r_i\right)$, and $p\left(\beta_j\right)$ are prior distributions. F can be compactly expressed as:

$$p\left(l_{ij}|r_i,C_i=l,\beta_j\right)=\beta_{jl}^{\mathbb{1}\{l_{ij}=r_i\}}\left(1-\beta_{jl}\right)^{\mathbb{1}\{l_{ij}\neq r_i\}} \tag{2}$$

where $\mathbb{1}\{l_{ij}=r_i\}$ is an indicator function taking the value of 1 if the observed label given by worker j to question i is equal to the ground truth. We assume a non-informative prior for true label $p(r_i=1)=p(r_i=0)=\frac{1}{2}$.

A. Inference And Parameter Estimation

In order to estimate the questions' true labels $r_i, i=1,\ldots,N$ and workers' trust vectors $\beta_j, j=1,\ldots,M$, their posterior distributions need to be computed. However, the computation of posterior distributions involves integrating out a large number of variables, making the computation intractable. We propose to use a variational approximation of the posterior distribution of variables in equation (1) with a factorized distribution family:

$$q\left(R,\beta,C,\lambda\right)=\prod_i q(r_i)\prod_i q\left(\lambda_i|\tilde{\alpha}_i\right)\prod_i q(C_i)\prod_{j,l}q\left(\beta_{jl}|\tilde{\theta}_{jl}\right) \tag{3}$$

The optimal forms of these factors are obtained by maximizing the following lower bound of the log likelihood of observed labels $\ln p(L)$:

$$\ln p(L)\geq \mathop{\mathbb{E}}_{q}\ln p\left(L,R,\beta,C,\lambda\right)-\mathop{\mathbb{E}}_{q}\ln q\left(R,\beta,C,\lambda\right) \tag{4}$$

We show inference details in Algorithm 1. Upon convergence of Algorithm 1, we obtain the approximate posterior distributions of the questions' true labels $\{r_i\}'s$ and of the workers' trust vectors $\{\beta_j\}$'s.

Algorithm 1: Multi-Domain Crowdsourcing

Input: initial values of hyperparameters α, θ
Output: approximate posterior $q\left(R,\beta,C,\lambda\right)$
Do the following updates repeatedly until convergence.
1) First update $q(\beta_j),\forall j=1,\ldots,M,l=1,\ldots,D$, sequentially, in the following way:

$$\beta_{jl}\sim \text{Beta}\left(\tilde{\theta}_{jl0},\tilde{\theta}_{jl1}\right) \tag{5}$$

where $\tilde{\theta}_{jl0}=\theta_{jl0}+\sum_{i\in N_j}q(C_i=l)q(R_i\neq l_{ij})$ and $\tilde{\theta}_{jl1}=\theta_{jl1}+\sum_{i\in N_j}q(C_i=l)q(R_i=l_{ij})$.
2) Then update $q(r_i),\forall i=1,\ldots,N$, sequentially, in the following way:

$$\ln q(r_i)\propto \ln p(r_i)+ \sum_{j\in M_i}\sum_{l=1}^{D}q(C_i=l)\Big[\delta_{ij}\left(\psi(\tilde{\theta}_{jl1})-\psi(\tilde{\theta}_{jl1}+\tilde{\theta}_{jl0})\right)+ (1-\delta_{ij})\left(\psi(\tilde{\theta}_{jl0})-\psi(\tilde{\theta}_{jl1}+\tilde{\theta}_{jl0})\right)\Big] \tag{6}$$

where $\psi(\cdot)$ is digamma function. Then normalize $q(r_i), r_i\in\{0,1\}$ to make them valid probabilities.
3) Then update $q(\lambda_i)$:

$$q(\lambda_i)\sim \text{Dir}\left(\{\tilde{\alpha}_{il}\}_{l=1}^{D}\right) \tag{7}$$

where $\text{Dir}(\cdot)$ is Dirichlet distribution and $\tilde{\alpha}_{il}=\alpha_l+q(C_i=l)$.
4) Then update $q(C_i=l)$:

$$\ln q(C_i)\propto \psi(\tilde{\alpha}_{il})-\psi\left(\sum_{k=1}^{D}\tilde{\alpha}_{ik}\right)+\sum_{j\in M_i}\Big[q(r_i=l_{ij})\left(\psi(\tilde{\theta}_{jl1})-\psi(\tilde{\theta}_{jl1}+\tilde{\theta}_{jl0})\right)+q(r_i\neq l_{ij})\left(\psi(\tilde{\theta}_{jl0})-\psi(\tilde{\theta}_{jl1}+\tilde{\theta}_{jl0})\right)\Big] \tag{8}$$

B. Integration with Features

Algorithm 1 ignores features of questions. In most cases we do have features associated with questions. These features help us better estimate both the questions' true labels and the workers' trust vectors. Our proposed model MDC can be easily extended to incorporate question features. The extended graphical model is shown in Fig. 2 from [1], where x denotes the features observed. We call this extended model MDFC. Intuitively, the features associated with questions allow us to better estimate the questions' concept vectors and

the workers' trust vectors so that true labels of questions can be more accurately inferred.

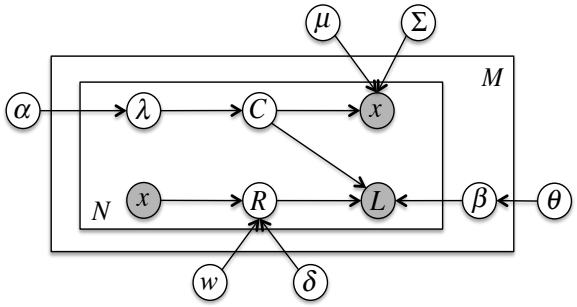


Fig. 2. The graphical model for observed data provided by workers L, features x, multi-domain expertise β, true labels R, domain variables C, and parameter for domain distribution λ. μ, Σ, w, and δ are model parameters.

Let's assume question i's feature vector x_i is a K-dimensional real-valued vector. The likelihood of feature x_i, given domain variable C_i, is modeled by a multivariate Gaussian distribution with μ_l' as the K-dimensional mean vector of the l-th domain and Σ_l as the $K \times K$ covariance matrix:

$$\ln p(x_i|C_i = l) \propto -\frac{1}{2}\left(x_i - \mu_l\right)^\top \Sigma_l^{-1}\left(x_i - \mu_l\right) - \frac{1}{2}\ln|\Sigma_l|, \tag{9}$$

where $|\Sigma_l|$ denotes the determinant of the covariance matrix of the l-th domain. The conditional distribution of the true label variable R_i, given feature variable x_i, can take various forms. We use the logistic regression model:

$$p(r_i = 1|x_i) = \left(1 + \exp\left(-w^\top x_i - \delta\right)\right)^{-1} \tag{10}$$

where w is the regression coefficient and δ is the intercept for the regression model.

The inference and parameter estimation of MDFC differs from Algorithm 1 in three ways: first, the update of $q(C_i)$ includes an extra term $\ln p(x_i|C_i = l)$; second, the update of $q(r_i)$ includes an additional term $p(r_i|x_i)$; third, there is an additional M-step to estimate model parameters μ_l's, Σ_l's, w, and δ given current approximate posteriors. The details of variational inference and model parameter estimation of MDFC is similar to that of MDTC.

C. Integration with Topics Models

In many crowdsourcing applications, we can often get access to questions' text descriptions. Given the text description, we can use the latent Dirichlet allocation to extract topic distribution of a question [19]. The advantage of topic models over the Gaussian mixture model in Section IV-B is that the domains (topics) are of low dimensions and are easier to interpret. For example, using topic models, a question might be assigned to the domain of sports while another question assgined to music domain. For a crowdsourcing platform, it needs to profile a worker's trust in all these interpretable topics instead of some latent unexplainable domain. We call this extended model with topic discovery MDTC and we will exploit the topic discovery of questions in the experiments section.

Algorithm 2: Multi-Domain Crowdsourcing With Features

Input: initial values of hyperparameters α, θ
Output: approximate posterior $q\left(R, \beta, C, \lambda\right)$
E-step and M-step are repeated until convergence
E-step: Given current estimation of model parameters μ_l's, Σ_l's, w, and δ: Do the following updates repeatedly until convergence.
1) First update $q(\beta_j), \forall j = 1, \ldots, M, l = 1, \ldots, D$, sequentially, in the following way:

$$\beta_{jl} \sim \text{Beta}\left(\tilde{\theta}_{jl0}, \tilde{\theta}_{jl1}\right) \tag{11}$$

where $\tilde{\theta}_{jl0} = \theta_{jl0} + \sum_{i \in N_j} q(C_i = l) q(R_i \neq l_{ij})$ and $\tilde{\theta}_{jl1} = \theta_{jl1} + \sum_{i \in N_j} q(C_i = l) q(R_i = l_{ij})$.
2) Then update $q(r_i), \forall i = 1, \ldots, N$, sequentially, in the following way:

$$\begin{aligned}\ln q(r_i) \propto \ln p(r_i) - \log\left(1 + \exp\left(-w^T x - \delta\right)\right) + \\ \sum_{j \in M_i} \sum_{l=1}^{D} q(C_i = l) \Big[\delta_{ij}\left(\psi(\tilde{\theta}_{jl1}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0})\right) + \\ (1 - \delta_{ij})\left(\psi(\tilde{\theta}_{jl0}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0})\right)\Big]\end{aligned} \tag{12}$$

where $\psi(\cdot)$ is digamma function. Then normalize $q(r_i), r_i \in \{0, 1\}$ to make them valid probabilities.
3) Then update $q(\lambda_i)$:

$$q(\lambda_i) \sim \text{Dir}\left(\{\tilde{\alpha}_{il}\}_{l=1}^{D}\right) \tag{13}$$

where $\text{Dir}(\cdot)$ is Dirichlet distribution and $\tilde{\alpha}_{il} = \alpha_l + q(C_i = l)$.
4) Then update $q(C_i = l)$:

$$\begin{aligned}\ln q(C_i = l) \propto \psi(\tilde{\alpha}_{il}) - \psi\left(\sum_{k=1}^{D} \tilde{\alpha}_{ik}\right) \\ + \sum_{j \in M_i} \Big[q(r_i = l_{ij})\left(\psi(\tilde{\theta}_{jl1}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0})\right) \\ + q(r_i \neq l_{ij})\left(\psi(\tilde{\theta}_{jl0}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0})\right)\Big] \\ - \frac{1}{2}\left(x_i - \mu_l\right)^T \Sigma_l^{-1}\left(x_i - \mu_l\right) - \frac{1}{2}\ln|\Sigma_l|\left(x_i - \mu_l\right)\end{aligned} \tag{14}$$

M-step: Given current approximate posterior distributions, obtain the estimates of μ_l's, Σ_l's, w, and δ by maximizing the expectation of the logarithm of the posterior:

$$\mu_l^{new} = \frac{\sum_{i=1}^{N} q(C_i = l) x_i}{\sum_{i=1}^{N} q(C_i = l)}$$

$$\Sigma_l^{new} = \frac{\sum_i q(C_i = l)\left(x_i - \mu_l^{new}\right)\left(x_i - \mu_l^{new}\right)^T}{\sum_{i=1}^{N} q(C_i = l)}$$

$$w^{new}, \delta^{new} = \operatorname*{argmax}_{w,\delta} \mathop{\mathbb{E}}_{q} \ln p\left(L, R, \beta, C, \lambda | \{\mu_l^{new}\}_{l=1}^{D}, \{\Sigma_l^{new}\}_{l=1}^{D}, w, \delta\right)$$

using L-BFGS quasi-Newton method

(15)

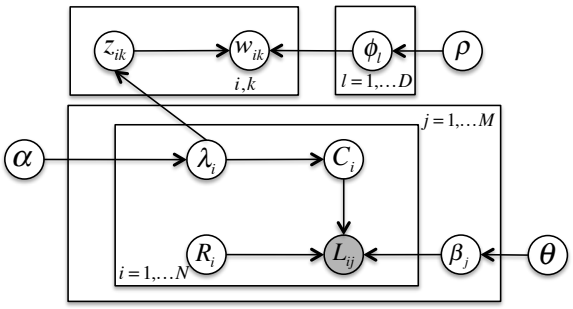


Fig. 3. The graphical model for MDTC. L are observed answers from workers, w_{ik} is word k observed in question i, multi-domain expertise β, true labels R, domain variables C, parameter for domain distribution λ, topic distribution for word k in question i : z_{ik}, word distribution for domain l : ϕ_l.

Each topic corresponds to one domain of a question. The learned topic distribution can then be used as a damping prior for domain variable C. We show that our MDC is flexible to incorporate topics models and it is an easy extension to jointly infer topic distribution and the true labels of quesitons and the workers' trust vectors in equation (1).

In addition to obtaining posterior probability distributions for R, β, C, λ, we can also obtain the posterior distribution for the topic distribution for the k-th word in the i-th question z_{ik}, and the word distribution for l-th topic ϕ_l simultaneously. Denote n_{iw} as the number of occurances of word w in question i and η_{iwl} as the probability that the word w in question i is associated with domain l. The variational inference process differs from Algorithm 1 in the following ways:

1) The λ_i's have a Dirichlet posterior distribution with parameter $\alpha_l + q(C_i = l) + \sum_w n_{iw}\eta_{iwl}$ where $\sum_w n_{iw}\eta_{iwl}$ is the additional term introduced by topic discovery.
2) The update of $q(z_{iw} = l) = \eta_{iwl}$ follows:

$$\ln \eta_{iwl} \propto \mathop{\mathbb{E}}_{q} \ln p(z_{iw} = l|\lambda_i) + \mathop{\mathbb{E}}_{q} \ln \phi_{lw} \tag{16}$$

where $\phi_{lw} = p\left(w_{ik} = w|\phi, z_{ik} = l\right)$.
3) The ϕ_l's have a Diricilet posterior distribution with parameter $\tilde{\Upsilon}_l$ as follows:

$$\tilde{\Upsilon}_{lw} = \Upsilon + \sum_i n_{iw}\eta_{iwl} \tag{17}$$

where Υ is the hyper-parameter of the Dirichlet prior distribution.

V. Active Learning

Given that MDC and MDTC can estimates workers' trust values in different dimensions, we explore the effect of optimally selecting both which question to ask and which worker(s) to assign the question to. The intuition is that instead of choosing random questions and assigning them to random workers, we seek to choose the most informative questions and solicit answers of those questions from workers that are most trustworthy in the same domains as the questions. In this section, we propose information gain

Algorithm 3: Multi-Domain Crowdsourcing With Topic Model

Input: initial values of hyperparameters α, θ
Output: approximate posterior $q\left(R, \beta, C, \lambda\right)$
Do the following updates repeatedly until convergence.
1) First update $q(\beta_j), \forall j = 1, \ldots, M, l = 1, \ldots, D$, sequentially, in the following way:

$$\beta_{jl} \sim \text{Beta}\left(\tilde{\theta}_{jl0}, \tilde{\theta}_{jl1}\right) \tag{18}$$

where $\tilde{\theta}_{jl0} = \theta_{jl0} + \sum_{i \in N_j} q(C_i = l) q(R_i \neq l_{ij})$ and $\tilde{\theta}_{jl1} = \theta_{jl1} + \sum_{i \in N_j} q(C_i = l) q(R_i = l_{ij})$.
2) Then update $q(r_i), \forall i = 1, \ldots, N$, sequentially, in the following way:

$$\begin{aligned} &\ln q(r_i) \propto \ln p(r_i) + \\ &\sum_{j \in M_i} \sum_{l=1}^{D} q(C_i = l) \Big[\delta_{ij} \left(\psi(\tilde{\theta}_{jl1}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0}) \right) + \\ &(1 - \delta_{ij}) \left(\psi(\tilde{\theta}_{jl0}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0}) \right) \Big] \end{aligned} \tag{19}$$

where $\psi(\cdot)$ is digamma function. Then normalize $q(r_i), r_i \in \{0, 1\}$ to make them valid probabilities.
3) Then update $q(\lambda_i)$:

$$q(\lambda_i) \sim \text{Dir}\left(\{\tilde{\alpha}_{il}\}_{l=1}^{D}\right) \tag{20}$$

where $\text{Dir}(\cdot)$ is Dirichlet distribution and $\tilde{\alpha}_{il} = \alpha_l + q(C_i = l) + \sum_w n_{iw}\eta_{iwl}.\ \alpha_l$.
4) Then update $q(C_i = l)$:

$$\begin{aligned} \ln q(C_i) \propto\ & \psi(\tilde{\alpha}_{il}) - \psi\left(\sum_{k=1}^{D} \tilde{\alpha}_{ik}\right) \\ &+ \sum_{j \in M_i} \Big[q(r_i = l_{ij}) \left(\psi(\tilde{\theta}_{jl1}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0}) \right) \\ &+ q(r_i \neq l_{ij}) \left(\psi(\tilde{\theta}_{jl0}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0}) \right) \Big] \end{aligned} \tag{21}$$

5) Then update $q(\phi_l)$:

$$q(\phi_l) \sim \text{Dir}\left(\{\tilde{\Upsilon}_{lw}\}\right) \tag{22}$$

where $\tilde{\Upsilon}_{lw} = \Upsilon + \sum_i n_{iw}\eta_{iwl}$.
6) Then update $q(z_{iw})$:

$$\begin{aligned} \ln p\left(z_{iw} = l\right) = \ln \eta_{iwl} =& \psi(\tilde{\alpha}_{il}) - \psi\left(\sum_{k=1}^{D} \tilde{\alpha}_{ik}\right) + \\ & \psi(\tilde{\Upsilon}_{lw}) - \psi(\sum_{w'} \tilde{\Upsilon}_{lw'}) \end{aligned} \tag{23}$$

For each i, w, normalize $\{\eta_{iwl}\}_{l=1}^{D}$ to make them valid probabilities.

metrics based on which we choose questions and metrics of probability of a worker's correctly answering a question based on which we choose workers. The results of this section are major innovations and additions to our work in [1].

A. Question Selection

There are many strategies for choosing which questions to ask. There might be monetization values associated to each question. In this case, questions of higher monetization values have higher priority. We propose entropy-based metric to measure the information gain of a question. Questions that have higher entropy are more uncertain and are therefore preferred over questions of lower entropy. Formally, we select the question that satisfies the following:

$$\underset{i}{\operatorname{argmin}} - \sum_{r_i \in \{0,1\}} q\left(r_i\right) \ln q\left(r_i\right) \tag{24}$$

B. Worker Selection

Given the chosen question i according to equation (24), we propose to choose a worker that has the highest probability of answering the question correctly. Formally, we choose the worker that satisfies:

$$\begin{aligned} &\underset{j}{\operatorname{argmax}}\, p\left(r_i = l_{ij}\right) \\ &= \underset{j}{\operatorname{argmax}} \sum_{l=1}^{D} p(C_i = l) p\left(r_i = l_{ij} | C_i = l\right) \\ &= \underset{j}{\operatorname{argmax}} \sum_{l=1}^{D} q(C_i = l)\, \mathbb{E}\, \beta_{jl} \\ &= \underset{j}{\operatorname{argmax}} \sum_{l=1}^{D} q(C_i = l) \frac{\tilde{\theta}_{jl1}}{\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0}} \end{aligned} \tag{25}$$

where $q(C_i = l)$ is the approximate posterior probability of question i belonging to topic l using variational inference methods in Section IV and $\tilde{\theta}_{jl1}, \tilde{\theta}_{jl0}$ are beta distribution parameters of worker j's trust in topic l. The intuition of the metric in equation (25) is that we choose the worker that has the largest cross product between its trust vector and the given question's topic distribution. The metric in equation (25) assumes that the selected worker will eventually answer the question given to him. However, in real applications, this is not the case. Workers might not be interested in answering the question. So we could model the probability of a worker answering a question into the metric in selecting workers:

$$\underset{j}{\operatorname{argmax}}\, p\left(j \text{ answers question } i\right) p\left(r_i = l_{ij}\right) \tag{26}$$

Equation (26) is the expected utility of choosing worker j to answer question i. Worker j's probability of answering question i, $p\left(j \text{ answers question } i\right)$, can be estimated using various supervised learning as long as we have workers' profile data, questions' metadata, and workers' history of engagement with questions (whether a worker answered or skipped a question).

TABLE I
WORKER SETTINGS FOR UCI DATASETS

worker type	domain 0	domain 1
type 1	0.5	0.5
type 2	0.95	0.5
type 3	0.5	0.95
type 4	0.95	0.95

VI. EXPERIMENTS

In this section, we compare our proposed models MDC, MDFC, and MDTC with crowdsourcing models with single dimensional trust (SDC) and show that our models have superior performance on both the UCI dataset and scientific text dataset. In addition as shown in [1], our models can effectively recover the workers' trust vectors which can be used to match the right workers to a given task in the future. The models we consider for comparison are listed as below (as in [1]):

1) MDC: our proposed multi-domain crowdsourcing model without features.
2) MDFC: extended model of MDC with continuously-valued features.
3) MDTC: another extended model of MDC that combines topic model given text descriptions associated with questions.
4) MV: the majority vote as the baseline algorithm.
5) SDC: the state-of-the-art in [10]. We call this algorithm SDC because it is equivalent to MDC when each worker is represented by only a scalar variable (single domain in our case)

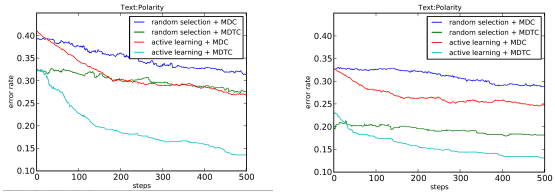
A. UCI datasets

We conducted experiments on the **pima** dataset from UCI Machine Learning Repository[1] [20]. Each data instance corresponds to a 8-dimensional feature of an anonymous patient. The dataset consists of 768 data intances and we ask the following question for each instance: should the patient be tested positive for diabetes. Since there are no worker-provided labels in this dataset, we simulate workers with varying reliability in different domains. We adopt k-means clustering to cluster the data into two clusters (domains). Therefore, each worker is profiled by a two-dimensional random vector. Details of the simulated workers are shown in Table I.

B. Text Data

To evaluate MDTC, we tested our model on 1000 sentences from the corpus of biomedical text with each sentence annotated by 5 workers [21]. Each worker answers whether a given sentence contains contradicting statements (Polarity). Each sentence has the scientific text along with the labels provided by 5 experts. However, since the labels provided by experts are almost consensus and the naive majority vote

[1] http://archive.ics.uci.edu/ml/datasets.html?sort=nameUp&view=list



(a) Error rates for bootstrap of 1500 random worker-question pairs

(b) Error rates for bootstrap of 2000 random worker-question pairs

Fig. 4. Error rates for active learning strategy and random selection with underlying models MDC and MDTC to estimate posterior probability distributions. Before using active learning/random selection, we randomly select 1500 and 2000 worker-questions pairs in Fig. 4(a) and Fig. 4(b) for initial setup.

algorithm gives ground truth answers, we need to simulate workers of varying trust of knowledge in different topics. When the number of topics (domains) is D, we simulate D workers in total, where worker j answers topic j close to perfectly (probabililty of right guess 0.97) and answers questions in topics other than j nearly randomly (probability of right guess 0.64). For each simulation setting, we repeat 30 times and report the mean error rate.

C. Active Learning on Text Data

We evaluated our new algorithms MDC, MDFC, MDTC, MV, against SDC in [1]. Here we present the evaluations of these same algorithms when active learning is added, with same datasets. Thus we evaluate our proposed strategies for selecting which questions to ask and selecting which workers to answer those questions. We tested the new algorithms with active learning on the same experimental setting as in [1], where we consider eight topics and eight workers in total. Worker j answers topic j close to perfectly (probabililty of right guess 0.97) and answers questions in topics other than j nearly randomly (probability of right guess 0.64). The models and the active learning strategies we use for comparison are:

1) MDC+ random selection: randomly select worker-question pairs and send the randomly-selected worker-question pair to MDC to update posterior probability distributions.
2) MDC+ active learning: select question according to equation (24) and select worker according to equation (25). Use MDC as the underlying model to estimate posterior probability distributions.
3) MDTC+ random selection: randomly select worker-question pairs and use MDTC as the underlying model.
4) MDTC+ active learning: select question according to equation (24) and select worker according to equation (25) and use MDTC as the underlying model.

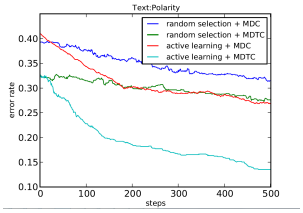
In Fig. 4, we plot the error rates of the four combinations of active learning strategies and underlying models at each learning step. Fig. 4(a) is for the case where we use 1500 randomly-selected worker-question pairs to initialize the model and Fig. 4(b) is for initial setup using 2000 pairs. In both cases, our proposed active learning strategy combined with the MDTC has the lowest error rates. Random selection combined with MDTC performs slightly better than active learning with MDC. This also further demonstrates the superiority of our proposed MDTC opposed to MDC. Using the MDC as the underlying, active learning consistently performs better than random selection. The error rates start at higher values and decrease much more in Fig. 4(a) than in Fig. 4(b). This also demontrates the powerfulness of active learning to help the system learn much faster when the crowdsourcing system does not have sufficient labels from workers.

In Fig. 5, the first row Fig. 5(a) - Fig. 5(d) shows the error rates for different combinations of active learning strategies and underlying models for batch sizes from 1 to 4. Increasing learning batch size is helpful to reducing time complexity because selecting worker-question pairs is much less time-consuming than updating posterior probability distributions which is an alternating iteration process. We observe that increasing learning batch size does not increase error rates. The second row Fig. 5(e) - Fig. 5(h) shows the sum of true label entropy of questions. For all learning batch sizes, the true label entropy for active learning combined with MDTC has the lowest entropy at all learning steps. Lower entropy values indicate that the crowdsourcing system has lower uncertainty over true label variables. Though we try to minimize the true label entropy in equation (24) which is effective in reducing true label entropy as shown in the figures in the second row, the worker trust entropy is also reduced as the learning step increases as illustrated in Fig. 5(i) - Fig. 5(l).

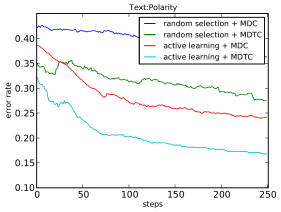
VII. Conclusion

In this paper, we propose a probabilistic model (MDC) that captures multi-domain characteristics of crowdsourcing questions and multi-dimensional trust of workers' knowledge. To show that our model MDC is very flexible and extensible to incorporate additional metadata associated with questions, we propose an extended model MDFC that incorporates continuously-valued features of questions and MDTC that also combines topic discovery. MDTC has the advantage that the domains are interpretable. To investigate the usage of our proposed models in adaptive task assignment setting, we propose strategies for choosing which questions to ask and which workers to assign the questions. We show that our proposed active learning strategies coupled with the models proposed to estimate posterior probability distributions can effectively decrease true label and worker trust entropy and reduce error rates.

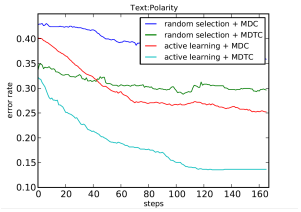
The results in this paper can be applied for fusion of information from multiple unreliable data sources instead of just workers in the open crowd. Examples of data sources are sensors, human input, and inference results given by another system backed by a different set of machine learning algorithms. Each of the data sources can be treated as a "worker" in this paper and we can thereafter use models in this paper to estimate the multi-domain trust values of the data sources and true labels of questions.



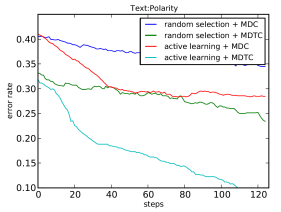
(a) Error rates for batch of 1 worker-question pairs



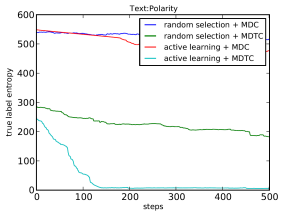
(b) Error rates for batch of 2 worker-question pairs



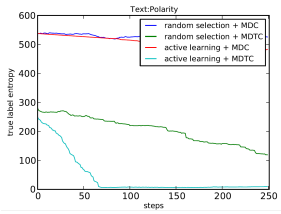
(c) Error rates for batch of 3 worker-question pairs



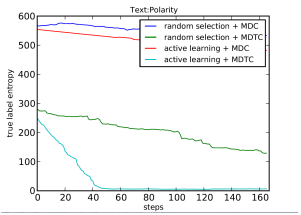
(d) Error rates for batch of 4 worker-question pairs



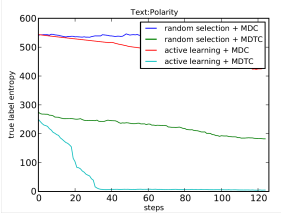
(e) Questions' true label entropy for batch of 1 worker-question pair



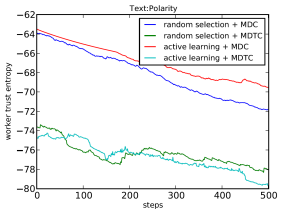
(f) Questions' true label entropy for batch of 2 worker-question pair



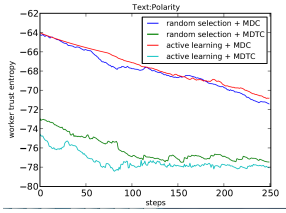
(g) Questions' true label entropy for batch of 3 worker-question pair



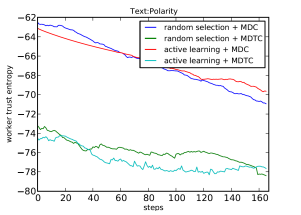
(h) Questions' true label entropy for batch of 4 worker-question pair



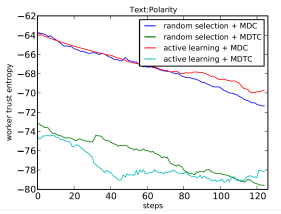
(i) Workers' trust entropy for batch of 1 worker-question pair



(j) Workers' trust entropy for batch of 2 worker-question pair



(k) Workers' trust entropy for batch of 3 worker-question pair



(l) Workers' trust entropy for batch of 4 worker-question pair

Fig. 5. Error rates, questions' true label entropy, and workers' trust value entropy versus steps of batch size 1, 2, 3, and 4.

REFERENCES

[1] X. Liu, H. He, and J. S. Baras, "Crowdsourcing with multi-dimensional trust," in *Information Fusion (Fusion), 2015 18th International Conference on*. IEEE, 2015, pp. 574–581.

[2] ——, "Trust-aware optimal crowdsourcing with budget constraint," in *Communications (ICC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1176–1181.

[3] M. Sensoy, G. de Mel, L. Kaplan, T. Pham, and T. J. Norman, "Tribe: Trust revision for information based on evidence," in *Information Fusion (FUSION), 2013 16th International Conference on*. IEEE, 2013, pp. 914–921.

[4] I. Matei, J. S. Baras, and T. Jiang, "A composite trust model and its application to collaborative distributed information fusion," in *Information Fusion, 2009. FUSION'09. 12th International Conference on*. IEEE, 2009, pp. 1950–1957.

[5] X. Liu and J. S. Baras, "Using trust in distributed consensus with adversaries in sensor and other networks," in *Information Fusion (FUSION), 2014 17th International Conference on*. IEEE, 2014, pp. 1–7.

[6] M. Richardson and P. Domingos, "Learning with knowledge from multiple experts," in *Proceedings of the 20th International Conference on Machine Learning (ICML 2003)*, 2003, pp. 624–631.

[7] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng, "Cheap and fast - but is it good? evaluating non-expert annotations for natural language tasks," in *Empirical Methods on Natural Language Processing (EMNLP)*, 2008, pp. 254–263.

[8] V. C. Raykar, S. Yu, L. H. Zhao, A. Jerebko, C. Florin, G. H. Valadez, L. Bogoni, and L. Moy, "Supervised learning from multiple experts: Whom to trust when everyone lies a bit," in *Proceedings of the 26th International Conference on Machine Learning (ICML 2009)*, 2009, pp. 889–896.

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

[10] Q. Liu, J. Peng, and A. Ihler, "Variational inference for crowdsourcing," in *Advances in Neural Information Processing Systems (NIPS)*, 2012, pp. 701–709.

[11] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan, "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise," in *Advances in Neural Information Processing Systems*, 2009, pp. 1207–1216.

[12] P. Welinder, S. Branson, S. Belongie, and P. Perona, "The multidimensional wisdom of crowds," in *Advances in Neural Information Processing Systems (NIPS)*, 2010, pp. 2424–2432.

[13] Y. Bachrach, T. Minka, J. Guiver, and T. Graepel, "How to grade a test without knowing the answers - a bayesian graphical model for adaptive crowdsourcing and aptitude testing," in *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, 2012, pp. 1183–1190.

[14] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. Springer-Verlag New York, Inc., 1994, pp. 3–12.

[15] N. Roy and A. McCallum, "Toward optimal active learning through monte carlo estimation of error reduction," *ICML, Williamstown*, pp. 441–448, 2001.

[16] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 1992, pp. 287–294.

[17] Y. Yan, R. Rosales, G. Fung, and J. G. Dy, "Active learning from crowds," in *Proceedings of the 28th International Conference on Machine Learning (ICML 2011)*, 2011, pp. 1161–1169.

[18] A. Jsang and R. Ismail, "The beta reputation system," in *Proceedings of the 15th bled electronic commerce conference*, 2002, pp. 41–55.

[19] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[20] D. J. Newman, S. Hettich, C. L. Blake, and C. J. Merz, "{UCI} repository of machine learning databases," 1998.

[21] A. Rzhetsky, H. Shatkay, and W. J. Wilbur, "How to get the most out of your curation effort," *PLoS computational biology*, vol. 5, no. 5, p. e1000391, 2009.

[22] S. J. Wright and J. Nocedal, *Numerical optimization*. Springer New York, 1999, vol. 2.