18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

Crowdsourcing with Multi-Dimensional Trust

Xiangyang Liu*, He He†, John S. Baras‡
*‡Institute for Systems Research and Dept. of Electrical and Computer Engineering, University of Maryland College Park
†Dept. of Computer Science, University of Maryland College Park
Email: *xyliu@umd.edu, †hhe@cs.umd.edu, ‡baras@umd.edu

***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 workers, trust is used to model workers' expertise. In a multi-domain knowledge learning task, however, using scalar-valued trust to model a worker's performance is not sufficient to reflect the worker's trustworthiness in each of the domains. To address this issue, 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 very 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. Finally, we evaluate our model on real-world datasets and demonstrate that our model is superior to state-of-the-art and the two extended models have even better performance. In addition, our models can effectively recover trust vectors of workers, which can be very useful in task assignment adaptive to workers' trust in the future. 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 workers. The answers from workers are then aggregated, taking into account reliability 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 answers. Most past research ignores the multi-domain property present in the questions above. 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 are good at different subjects. The above 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 this paper, we propose to model each question by a *concept vector*, which is a real random vector where the value in a particular dimension indicates its association in that dimension. Back to the test grading example, each bio-chemistry question is represented by a two-dimensional hidden concept vector with the first dimension being chemistry and the second dimension being biology. So a concept vector $[0.7, 0.3]$ means the question is more associated with chemistry. Note that the concept vector can be far more general than this. In the case of identifying causal relationships between entities, reasoning ability and past experience are two dimensions of the concept vector. Each worker is modeled also by a trust vector, which is a real random vector with each dimension representing the trustworthiness of the worker in that dimension. The multi-domain property of questions and workers for the biology-chemistry example is illustrated in Fig. 1. Our goal 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. We therefore propose a probabilistic model that incorporates questions' concept vectors, workers' trust vectors, answers submitted by workers and design an inference algorithm that jointly estimates true label of questions along with concept vectors and trust vectors. The inference algorithm is based on a variational approximation of posterior distributions using a factorial distribution family. In addition, we extend 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 propose an extended model that integrates topic discovery.

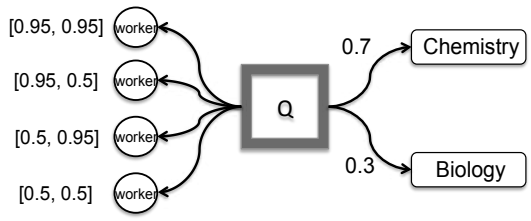


Fig. 1. Multi-domain property of questions and workers in the test grading example. Q represents a question with concept vector $[0.7, 0.3]$ shown on the edges. Several workers with different two-dimensional trust vectors provide answers.

Our contributions are as follows:

- We formulate a probabilistic model of crowdsourcing tasks with *multi-domain* characteristics and propose a novel inference method based on variational inference.
- Our model is very flexible and can be easily extended. In applications where each question comes with a feature vector, we further develop an extended model that handles questions with continuously-valued features.
- We further extend the model by combining a multi-domain crowdsourcing model with topic discovery based on questions' text descriptions and derive an

analytical solution to the collective variational inference.

II. Related Work

There are a lot of works on how to leverage trust models to better aggregate information from multiple sources. Conflicts between information provided by different sources were used to revise trust in the information [12]. 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 [7]. 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 [6].

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. [10] 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 [3], [5], [9], [13].

A model that jointly infers label of image, trust of each labeler and difficulty of image is proposed in [15]. 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 many local optima difficulties. Another work that went a step further based on signal detection theory is [14], where they assume each question comes with a feature set and models each worker by a multidimensional classifier in an abstract feature space. Our model 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 to posterior distributions of hidden variables in [14]. An approach in the spirit of test theory and item-response theory (IRT) was proposed in [1] and they relied on approximate message-passing for inference. Their model is not as flexible and extensivle as our model because they have to redesign their model to incorporate rich metadata associated with each question.

III. Problem Definitions

We 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 [4] 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.

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 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. 2. The joint probability distribution is

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

where N is the total number of questions, M is the total workers, and D is the total number of domains. $p(l_{ij}|r_i, C_i = l, \beta_j)$ is the output distribution *F* in Fig. 2 and is the likelihood of worker j's answer given its expertise vector and the domain variable of question i, and the true label. $p(r_i)$, and $p(\beta_j)$ are

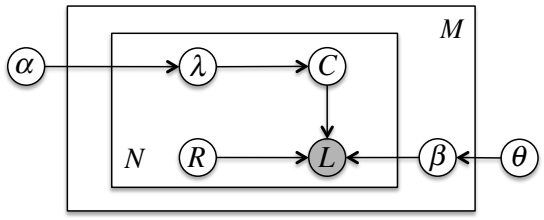


Fig. 2. The graphical model for observed data provided by workers L, multi-domain expertise β, true labels R, domain variables C, and concept vectors λ. M is the total number of workers. N the number of questions. α is the hyperparameter of the Dirichlet prior distribution for λ and θ is the hyperparameter of the Beta prior distribution for β.

prior distributions. F can be compactly expressed as:

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

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:

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

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

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

We show inference details in Algorithm 1. Updates for each factor are derived in the Appendix. 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.

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. 3, 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.

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:

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

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:

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

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 $q(C_i)$

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:

$$q(\beta_{jl}) \propto \ln p(\beta_{jl}) + \sum_{i \in N_j} \mathop{\mathbb{E}}_{q} \ln p\left(l_{ij}|r_i, C_i = l, \beta_j\right) \quad (5)$$

2) Then update $q(r_i), \forall i = 1, \ldots, N$, sequentially, in the following way:

$$\ln q(r_i) \propto \sum_{j \in M_i} \mathop{\mathbb{E}}_{q} \ln p\left(l_{ij}|r_i, C_i = l, \beta_j\right) \quad (6)$$

and then normalize $q(r_i), r_i \in \{0, 1\}$ to make them valid probabilities.
3) Then update $q(\lambda_i)$:

$$\ln q(\lambda_i) \propto \ln p(\lambda_i) + \mathop{\mathbb{E}}_{q} \ln p(C_i|\lambda_i) \quad (7)$$

4) Then update $q(C_i = l)$:

$$\ln q(C_i) \propto \mathop{\mathbb{E}}_{q(\lambda_i)} \ln p(C_i|\lambda_i) + \mathop{\mathbb{E}}_{q} \ln p\left(l_{ij}|r_i, C_i = l, \beta_j\right) \quad (8)$$

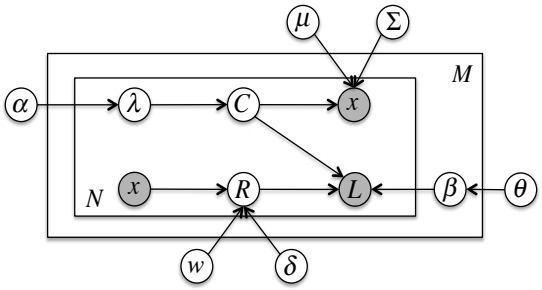


Fig. 3. The graphical model for observed data provided by workers L, features x, multi-domain expertise β, true labels R, domain variables C, and parameter for domain distribution λ. μ, Σ, w, and δ are model parameters.

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 and are shown in the Appendix.

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 [2]. 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.

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:

$$\ln \eta_{iwl} \propto \mathop{\mathbb{E}}_{q} \ln p(z_{iw} = l|\lambda_i) + \mathop{\mathbb{E}}_{q} \ln \phi_{lw} \qquad (11)$$

   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:

$$\tilde{\Upsilon}_{lw} = \Upsilon + \sum_i n_{iw}\eta_{iwl} \qquad (12)$$

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

V. 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, 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 follows:

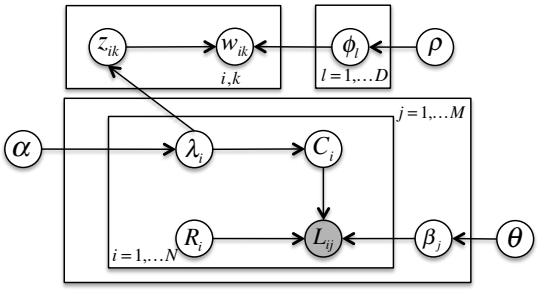


Fig. 4. The graphical model for MDTC. L are observed answers from workers, w_{ik} is word k observed in question i, multi-domain expertise β, true labels R, domain variables C, parameter for domain distribution λ, topic distribution for word k in question i : z_{ik}, word distribution for domain l : ϕ_l.

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 [5]. 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] [8]. 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. Type 1 workers are malicious in both domains, answering questions correctly with probability 0.5, type 2 workers answer questions in domain 0 correctly with probability 0.95 and answer those in domain 1 correctly with probability 0.5 while type 3 workers answer questions in domain 0 correctly with probability 0.5 and answer questions in domain 1 correctly with probability 0.95, and type 4 workers are good at questions in both domains and answer questions correctly with probability 0.95. In order to show that our model MDC and MDFC works with increasing number of workers that are not trustworthy, we simulated several groups of worker settings with increasing number of type 1 workers.

We compare MDC with MV and SDC when no features are included and compare MDFC with MV and SDC when features are incorporated. We use accuracy as the evaluation criterion

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

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

TABLE II. ERROR RATES OF VARIOUS METHODS ON UCI DATASET PIMA INDIANS.

pima dataset	MV	SDC	MDFC	MDC
$(1, 2, 2, 1)$	0.098	0.040	**0.009**	×
$(2, 2, 2, 1)$	0.103	0.042	**0.009**	×
$(3, 2, 2, 1)$	0.150	0.042	**0.008**	×
$(1, 2, 2, 1)$, NF	0.098	0.040	×	**0.039**
$(2, 2, 2, 1)$, NF	0.103	**0.042**	×	0.043
$(3, 2, 2, 1)$, NF	0.150	0.042	×	**0.041**

In the expression $(1, 2, 2, 1)$, the four numbers from left to right mean: there are 1 worker of type 1, 2 workers of type 2, 2 workers type 3, and 1 worker of type 4. NF means omitting the features in pima dataset.

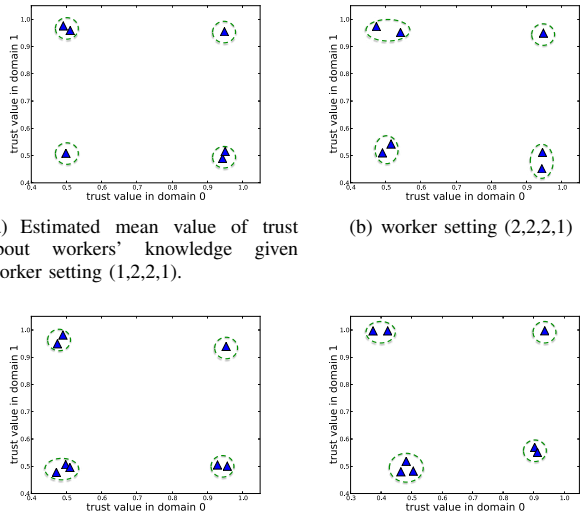
and report results in Table II, where the first column denotes worker settings.

When features are used, MDFC results in lowest error rates. When features are omitted, MDC and SDC perform nearly equally well. This could be explained by that when features are not utilized to infer domain distributions for questions, the estimated domain distributions by MDC might be inconsistent with the truth. However, MDC is still very attractive because it can still effectively estimate the workers' reliability in different domains as shown in Fig. 5(d). This can be useful for task assignment adaptive to workers' trust in the future. Specifically, upon arrival of a new task, we can use the estimated profile of workers to match the question belonging to a particular domain to a worker that is trustworthy in that domain. For example, consider the case when we need to know the true label of a new question that belongs to a certain domain. Then we can match the question with workers that have the highest reliability in that domain.

In Fig. 5, we show that both MDC and MDFC can effectively estimate workers' trust values in both domains considered. Each triangle stands for a worker's trust profile (a two-dimensional real-valuded trust vector) and the dotted circle is used to cluster workers whose estimated trust values are close to each other. Taking a closer look at Fig. 5(a), we see that one worker is clustered close to $(0.51, 0.51)$, two workes close to $(0.95, 0.5)$, two workers close to $(0.5, 0.96)$, and one worker close to $(0.95, 0.95)$. This estimation of trust vectors is consistent with the worker setting $(1, 2, 2, 1)$. Workers' trust vectors can also be effectively estimated in other worker settings in Fig. 5(b), Fig. 5(c), and Fig. 5(d).

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 [11]. 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 algorithm gives ground truth answers, we need to simulate workers of varying



(a) Estimated mean value of trust about workers' knowledge given worker setting (1,2,2,1).

(b) worker setting (2,2,2,1)

(c) worker setting (3,2,2,1)

(d) worker setting (3,2,2,1) and no features available

Fig. 5. Estimated worker reliability under different simulation settings on pima indians dataset. The estimated trust about workers' knowledge in Fig. 5(a), Fig. 5(b), and Fig. 5(c) are by MDFC and the results in Fig. 5(d) are by MDC.

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.

To the best of our knowledge, there is no comparative model that integrates topics models into a probabilistic crowdsourcing framework in the literature, therefore we compare the performance of MDTC with MDC that ignores topic information and with the baseline majority vote algorithm. The mean error rates are reported in Table III. We can see that in all experiments with the number of topics ranging from 4 to 14, MDTC gives the lowest error rate, outperforming MDC by over 50%. This strongly demonstrates the power of MDTC over other models that do not take into account text information.

To further show that MDTC can effectively recover the reliability of workers in different topics, we plot, for each worker, the mean value of trust for each worker in each of the eight topics (T8) as a heatmap in Fig. 6. The x-axis denotes topic index and the y-axis denotes worker index. The intensity of the color in the j-th row and l-th column denotes the trust value of worker j in l-th dimension. We can see that the diagonal blocks have more intense color than others, which is consistent with the simulation setting where each worker $j \in \{0, 1, ..., 7\}$ is trustworthy in topic j and is not reliable in topics other than j. The estimated trust vectors of workers in all eight topics can be very useful in the following scenario: if for example a new question with concept vector $[0.93, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01]$ is added, we probably want to match this question with a worker whose trust vector has high value in the first dimension. The

TABLE III. ERROR RATES OF VARIOUS METHODS ON TEXT:POLARITY. $T4$ DENOTES THE ASSUMPTION OF 4 TOPICS.

scientific text	MV	MDC	MDTC
$T4$	0.181	0.095	**0.044**
$T6$	0.160	0.089	**0.037**
$T8$	0.141	0.082	**0.034**
$T10$	0.125	0.074	**0.032**
$T12$	0.116	0.069	**0.032**
$T14$	0.100	0.064	**0.032**

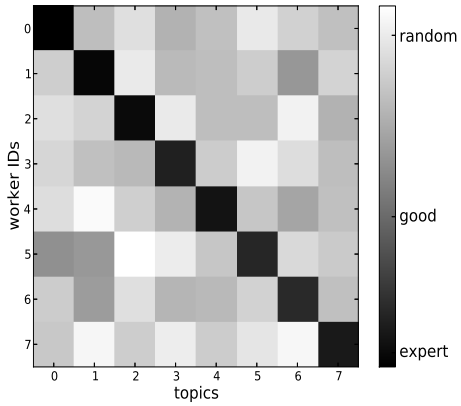


Fig. 6. Trust matrix about workers' knowledge over topics estimated by MDTC model.

representative words (top 10 words with the highest probability in a particular topic) in all eight topics are shown in Table IV.

VI. 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. We show that our proposed models have superior performance compared to state-of-the-art on two real datasets and can effectively recover the trust vectors of workers. This can be very useful in task assignment adaptive to workers' trust values in different dimensions in the future. We assume answers from workers are collected first and are then fed to models for inference. For future work, we will investigate the problem of choosing which question to be labeled next by which worker based on the trust vectors of workers.

TABLE IV. REPRESENTATIVE WORDS IN TOPICS ON SCIENTIFIC TEXT

topic0	ins, protein, cells, express, activity, mutant, resulted, similar, human, rna
topic1	ins, cells, binding, two, presence, day, method, study, acids, reporter
topic2	ins, binding, process, protein, cells, quot, factor, structure, dna, splice
topic3	ins, cells, blotting, protein, using, analysis, western, express, antibodies, demonstrate
topic4	ins, signal, wnt, cells, activity, resulted, using, protein, pathway, regulation
topic5	ins, system, two, sequences, suggest, cloning, data, effects, transcripts, different
topic6	ins, activity, cells, dna, binding, forms, gene, required, phosphorylation, receptor
topic7	ins, min, cells, containing, activity, described, incubated, protein, mms, buffer

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.

VII. ACKNOWLEDGEMENTS

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

VIII. APPENDIX

This appendix derives the approximate posteriors in MDC, MDFC and MDTC using variational inference.

A. Updates in MDC

1) Update each factor $q(\beta_{jl})$: by variational approach, $q(\beta_{jl})$ has the following form:

$$\begin{aligned}\ln q(\beta_{jl}) &\propto \ln p(\beta_{jl}|\theta_{jl0},\theta_{jl1}) + \sum_{i\in N_j} \mathop{\mathbb{E}}_{q} \ln p\left(l_{ij}|r_i, C_i=l, \beta_j\right) \\ &= \left(\theta_{jl1} + \sum_{i\in N_j} q(C_i=l) q(R_i = l_{ij})\right) \ln \beta_{jl} \\ &+ \left(\theta_{jl0} + \sum_{i\in N_j} q(C_i=l) q(R_i \neq l_{ij})\right) \ln\left(1-\beta_{jl}\right)\end{aligned} \tag{13}$$

We can see that the above posterior of $q(\beta_{jl})$ has Beta distribution Beta$(\tilde{\theta}_{jl})$ with parameter $\tilde{\theta}_{jl} = [\tilde{\theta}_{jl0}, \tilde{\theta}_{jl1}]$, 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) Update each factor $q(r_i)$: the optimal approximate posterior of $q(r_i)$ takes the form:

$$\begin{aligned}\ln q(r_i) &\propto \ln p(r_i) + \sum_{j\in M_i} \mathop{\mathbb{E}}_{q} \ln p\left(l_{ij}|r_i, C_i, \beta_j\right) \\ &= \ln p(r_i) + \sum_{j\in M_i} \sum_{l=1}^{D} q_{il} \left[\delta_{ij} \mathop{\mathbb{E}}_{q} \ln \beta_{jl} + (1-\delta_{ij}) \mathop{\mathbb{E}}_{q} \ln\left(1-\beta_{jl}\right)\right]\end{aligned} \tag{14}$$

where $q_{il} = q(C_i = l)$. The expectation of logarithmic beta variables

$$\mathop{\mathbb{E}}_{q} \ln \beta_{jl} = \psi(\tilde{\theta}_{jl1}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0})$$

and

$$\mathop{\mathbb{E}}_{q} \ln\left(1-\beta_{jl}\right) = \psi(\tilde{\theta}_{jl0}) - \psi(\tilde{\theta}_{jl1} + \tilde{\theta}_{jl0})$$

where $\psi(\cdot)$ is digamma function. Then $q(r_i)$ is normalized in order to be a valid probability.

3) Update each factor $q(\lambda_i)$: assume λ_i takes a Dirichlet prior with parameter $\{\alpha_l\}_{l=1}^D$. We have

$$\begin{aligned}\ln q(\lambda_i) &\propto \ln p(\lambda_i) + \mathop{\mathbb{E}}_{q} \ln p(C_i|\lambda_i) \\ &= \ln p(\lambda_i) + \sum_{l=1}^{D} q(C_i = l) \ln \lambda_{il} = \ln \prod_{l=1}^{D} \lambda_{il}^{\tilde{\alpha}_{il}-1}\end{aligned} \tag{15}$$

where $\tilde{\alpha}_{il} = \alpha_l + q(C_i = l)$. It is evident that the posterior $q(\lambda_i)$ also has a Dirichlet distribution with parameters $\{\tilde{\alpha}_{il}\}_{l=1}^D$.

4) Update each factor $q(C_i)$: We have

$$\begin{aligned}&\ln q(C_i = l) \propto \mathop{\mathbb{E}}_{q(\lambda_i)} \ln p(C_i = l|\lambda_i) + \mathop{\mathbb{E}}_{q} \ln p\left(l_{ij}|r_i, C_i = l, \beta_j\right) \\ &= \mathop{\mathbb{E}}_{q(\lambda_i)} \ln \lambda_{il} \\ &+ \sum_{j \in M_i} \left[q(r_i = l_{ij}) \mathop{\mathbb{E}}_{q} \ln \beta_{jl} + q(r_i \neq l_{ij}) \mathop{\mathbb{E}}_{q} \ln (1 - \beta_{jl}) \right]\end{aligned} \tag{16}$$

where

$$\mathop{\mathbb{E}}_{q(\lambda_i)} \ln \lambda_{il} = \psi(\tilde{\alpha}_{il}) - \psi\left(\sum_{k=1}^{D} \tilde{\alpha}_{ik}\right)$$

B. Updates in MDFC

The updates in MDC are divided into two steps: E-step and M-step. In E-step, we obtain the approximate posterior distributions for different random variables in our model given current estimates of model parameters μ_l's, Σ_l's, w, and δ. In M-step, the model parameters are obtained given current posterior approximations. E-step and M-step are iterated until convergence.

1) E-step: Since the updates of posterior distributions of β_j's and λ_i's are the same as those in MDC, we just show the updates of $q(C_i)$'s and $q(r_i)$'s below:

For $q(C_i)$, besides the terms in equation (16), it has an extra term:

$$\begin{aligned}\ln p\left(x_i|C_i = l, \mu_l, \Sigma_l\right) = &-\frac{1}{2}\left(x_i - \mu_l\right)^T \Sigma_l^{-1} \left(x_i - \mu_l\right) - \\ &\frac{1}{2} \ln |\Sigma_l| \left(x_i - \mu_l\right)\end{aligned} \tag{17}$$

where $(*)^T$ denotes the transpose of the term inside the parenthesis. For $q(r_i)$, besides the terms in equation (14), it has an additional term:

$$\begin{aligned}p(r_i|x_i) &= \sigma(w^T x_i + \delta)^{\mathbb{1}\{r_i=1\}} \left(1 - \sigma(w^T x_i + \delta)\right)^{\mathbb{1}\{r_i=0\}} \\ &= \sigma(w^T x_i + \delta)^{\mathbb{1}\{r_i=1\}} \sigma(-w^T x_i - \delta)^{\mathbb{1}\{r_i=0\}}\end{aligned} \tag{18}$$

where $\sigma(*)$ denotes the sigmoid function and $\mathbb{1}\{r_i = 1\}$ denotes the indicator function that equals to 1 if $r_i = 1$ and equals to 0 if not. The second equality in equation (18) holds because for the sigmoid function we have $\sigma(-z) = 1 - \sigma(z)$.

2) M-step: In order to estimate the model parameters μ_l's, Σ_l's, w, and δ, we adopt alternating optimization by optimizing one set of the parameters while fixing the others. The objective function is the expectation of the logarithm of the likelihood function $Q = \mathbb{E}_q \ln p\left(L, R, \beta, C, \lambda|\mu, \Sigma, w, \delta\right)$ given current approximate posteriors q. Then we have:

$$\begin{aligned}\mu_l^{new} &= \frac{\sum_{i=1}^N q(C_i = l) x_i}{\sum_{i=1}^N q(C_i = l)} \\ \Sigma_l^{new} &= \frac{\sum_i q(C_i = l)\left(x_i - \mu_l^{new}\right)\left(x_i - \mu_l^{new}\right)^T}{\sum_{i=1}^N q(C_i = l)} \\ \frac{\partial Q}{\partial w} &= \sum_{i=1}^{N} \left[q(r_i = 1)\sigma(w^T x_i + \delta) - q(r_i = 0)\sigma(-w^T x_i - \delta)\right] x_i \\ \frac{\partial Q}{\partial \delta} &= \sum_{i=1}^{N} \left[q(r_i = 1)\sigma(w^T x_i + \delta) - q(r_i = 0)\sigma(-w^T x_i - \delta)\right]\end{aligned} \tag{19}$$

To obtain the optimal values of w and δ, we derive the first order derivatives $\frac{\partial Q}{\partial w}$ and $\frac{\partial Q}{\partial \delta}$ and use the L-BFGS quasi-Newton method [16].

C. Updates in MDTC

The updates for the parameters of the variational posterior distribution for C_i, β_{jl}, and r_i remain the same since no additional dependencies for those variables are introduced as shown in Fig. 4. We derive the posteriors for λ_i, z_{iw}, and ϕ_l.

1) Update each factor $q(\lambda_i)$: We have

$$\begin{aligned}\ln q(\lambda_i) &\propto \exp\left\{\ln p(\lambda_i) + \mathop{\mathbb{E}}_{q} \ln p(C_i|\lambda_i) + \sum_{w} \mathop{\mathbb{E}}_{q} \ln p(z_{iw}|\lambda_i)\right\} \\ &\propto \prod_{l=1}^{D} {\lambda_{il}}^{\tilde{\alpha}_{il}-1}\end{aligned} \tag{20}$$

where $\tilde{\alpha}_{il} = \alpha_l + q(C_i = l) + \sum_w n_{iw}\eta_{iwl}$. α_l is the parameter of the prior Dirichlet distribution of λ.

2) Update each factor $q(\phi_l)$: We have

$$\begin{aligned}\ln q(\phi_l) &\propto \ln p(\phi_l) + \sum_{i,k} \mathop{\mathbb{E}}_{q} \ln p\left(w_{ik}|\phi_l, z_{ik}\right) \\ &= \ln p(\phi_l) + \sum_{i,k} q\left(z_{ik} = l\right) \ln \phi_{l w_{ik}}\end{aligned} \tag{21}$$

It is evident that ϕ_l has a Dirichlet posterior distribution with parameter:

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

3) Update each factor $q(z_{iw})$: We have

$$\begin{aligned}\ln \eta_{iwl} &\propto \mathop{\mathbb{E}}_{q} \ln p(z_{iw} = l|\lambda_i) + \mathop{\mathbb{E}}_{q} \ln \phi_{lw} \\ &= \mathop{\mathbb{E}}_{q} \ln \lambda_{il} + \mathop{\mathbb{E}}_{q} \ln \phi_{lw}\end{aligned} \tag{22}$$

where $\mathbb{E}_q \ln \phi_{lw} = \psi(\tilde{\Upsilon}_{lw}) - \psi(\sum_{w'} \tilde{\Upsilon}_{lw'})$. Then we need to normalize $\eta_{iwl}, l = 1, \ldots, D$ to form valid probabilities.

REFERENCES

[1] 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)*, pages 1183–1190, 2012.

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

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

[4] A. Jsang and R. Ismail. The beta reputation system. In *Proceedings of the 15th bled electronic commerce conference*, pages 41–55, 2002.

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

[6] 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*, pages 1–7. IEEE, 2014.

[7] 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*, pages 1950–1957. IEEE, 2009.

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

[9] 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)*, pages 889–896, 2009.

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

[11] A. Rzhetsky, H. Shatkay, and W. J. Wilbur. How to get the most out of your curation effort. *PLoS computational biology*, 5(5):e1000391, 2009.

[12] 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*, pages 914–921. IEEE, 2013.

[13] 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)*, pages 254–263, 2008.

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

[15] 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*, pages 1207–1216, 2009.

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