2015 IEEE 54th Annual Conference on Decision and Control (CDC)
December 15-18, 2015. Osaka, Japan

Trust-Aware Crowdsourcing With Domain Knowledge

Xiangyang Liu and John S. Baras

I. ABSTRACT

The rise of social network and crowdsourcing platforms makes it convenient to take advantage of the collective intelligence to estimate true labels of questions of interest. However, input from workers is often noisy and even malicious. Trust is used to model workers in order to better estimate true labels of questions. We observe that questions are often not independent in real life applications. Instead, there are logical relations between them. Similarly, workers that provide answers are not independent of each other either. Answers given by workers with similar attributes tend to be correlated. Therefore, we propose a novel unified graphical model consisting of two layers. The top layer encodes domain knowledge which allows users to express logical relations using first-order logic rules and the bottom layer encodes a traditional crowdsourcing graphical model. Our model can be seen as a generalized probabilistic soft logic framework that encodes both logical relations and probabilistic dependencies. To solve the collective inference problem efficiently, we have devised a scalable joint inference algorithm based on the alternating direction method of multipliers. Finally, we demonstrate that our model is superior to state-of-the-art by testing it on multiple real-world datasets.

II. INTRODUCTION

In a typical crowdsourcing setting, multiple workers are solicited to provided answers for each of the questions. For example, Facebook users volunteer to perform various annotation tasks on Facebook edit page[1], or workers on *Amazon Mechanical Turk*[2] get paid for solving various tasks uploaded by task requesters. An example task is to determine whether a given plaintext headline expresses one or more of the emotions anger, disgust, fear, joy, sadness, and surprise. So there are six questions associated with a single headline. Our observation is that these questions are not independent. If the system is more confident that a headline exhibits anger emotion, then the headline is not likely to express joy. In addition, workers that provide answers for these questions tend to give similar answers if they share same attributes. These observations motivate us to ultilize these logical constraints, which we call *domain knowledge*, to more accurately estimate true labels of questions as well as trust values of workers.

In this paper, we propose a trust-aware crowdsourcing with domain knowledge framework (TCDK). It is a two-layered probabilistic graphical model, where the top layer encodes the logical relationships using first-order logic rules and the bottom layer encodes the probabilistic dependencies between random variables in traditional crowdsourcing graphical models. We show that the dependency between of the top and bottom layers is equivalent to a special rule, called *cost-function rule* with a fixed weight of 1.0. This two-layered framework can be seen as a generalized probabilistic soft logic framework that contains both logical and probabilistic relations while the probabilistic soft logic in [6] only contains logical relations. TCDK allows users to integrate high level domain knowledge easily into traditional crowdsourcing graphical models without having to derive a whole new model from scratch. More importantly, the leverage of domain knowledge can help the system better estimate true labels of questions and at the same time more accurately estimate the trust values of workers. To jointly infer the true labels of questions and trust values of workers, we develop an inference algorithm based on the alternating direction method of multipliers. More specifically, the algorithm alternates between optimizing variables in the lower layer while fixing variables in the upper layer and optimizing variables in the upper layer while fixing variables in the bottom layer.

Our contributions are the following:

1) We formulate a novel trust-aware crowdsourcing with domain knowledge framework that combines domain knowledge with a traditional crowdsourcing graphical model. Users can express high level domain knowledge without having to re-define the model and the framework can be used to integrate multiple data sources.
2) We develop a scalable joint inference algorithm for estimating true label variables and trust values of workers based on alternating consensus optimization. The inference algorithm can be easily scaled to multiple machines.

III. RELATED WORK

To address the issue of noisy and malicious workers in crowdsourcing systems, many models are developed to jointly estimate true labels of questions and trust of workers [4], [7], [11], [8]. All these works are based on the assumption that questions' true label variables are independent and the trusts of different workers are independent too. However,

Research partially supported by grants AFOSR MURI FA-9550-10-1-0573, NSF CNS-1035655, NSF CNS-1018346 and by Maryland Procurement Office contract H98230-14-C-0127.

The authors are with the Institute for Systems Research and the Department of Electrical and Computer Engineering at the University of Maryland, Collge Park, MD, 20742, USA xyliu@umd.edu, baras@umd.edu

[1] https://www.facebook.com/editor

[2] https://www.mturk.com/mturk/welcome

the assumption is shown to be invalid in the annotation of headline emotion example in Section II.

[10] did consider dependency between workers by revealing the latent group structure among dependent workers and aggregated information at the group level rather than from individual workers. Still, their model did not capture the logical dependencies among questions as in our work. In the natural language processing literature, a framework called Fold·All [1] was proposed to integrate domain knowledge into Latent Dirichlet Allocation (LDA). Their framework can be seen as an extension of the Markov Logic Network while the top layer of our framework can be seen as the generalized probabilistic soft logic [6].

IV. The TCDK framework

We consider a crowdsourcing task with N questions and M workers available in total. Each question is answered by a subset of M workers. Each worker j is modeled by a random variable $\beta_j \in [0,1]$ that has a Dirichlet prior with parameter θ. Higher value of β_j indicates that the worker is more trustworthy. The variable $z_i \in \{0,1\}$ is used to denote question i's true label. The answer to question i given by worker j is denoted by $l_{ij} \in \{0,1\}$.

We first review the graphical model used in [7]:

$$p\left(L,z,\beta|\theta\right) \propto \prod_{i=1}^{N}\prod_{j\in M_i} p\left(\beta_j|\theta\right) p\left(l_{ij}|z_i,\beta_j\right) \tag{1}$$

where M_i is the set of workers that give answers to question i. The task is to infer questions' true labels z_i's and estimate workers' trust values β_j's.

In the above model, the true labels z_i's are assumed to be independent. In TCDK, we incorporate the logical relations between questions using first-order logic rule syntax. Example rules are:

$$\begin{aligned} &\text{ContainsHappiness}(i) \Rightarrow \text{ContainsAnger}(i) \\ &\text{Trust}(j_1) \wedge \text{SimilarBackground}(j_1,j_2) \Rightarrow \text{Trust}(j_2) \end{aligned} \tag{2}$$

The first rule states that if text clip i expresses emotion happiness, then it is unlikely that the text expresses anger and the second rule states that if the worker j_1 is trustworthy and he has similar background with another worker j_2, the worker j_2 tends to be trustworthy too. For each first-order logic rule ℓ as defined in (2), we represent the weight of the rule as λ_ℓ and the set of groundings of rule r as R_ℓ. Higher value of λ_ℓ indicates that the rule ℓ is more important compared to other rules. For each grounded rule r, we associate a non-negative potential function $\phi_r(z,\beta)$. We will discuss the specific definition of $\phi_r(z,\beta)$ later.

Putting together the domain knowledge expressed using first-order logic rules as in (2) and the traditional crowdsourcing model in (1), our proposed model Crowdsourcing with Domain Knowledge (TCDK) defines a generative model

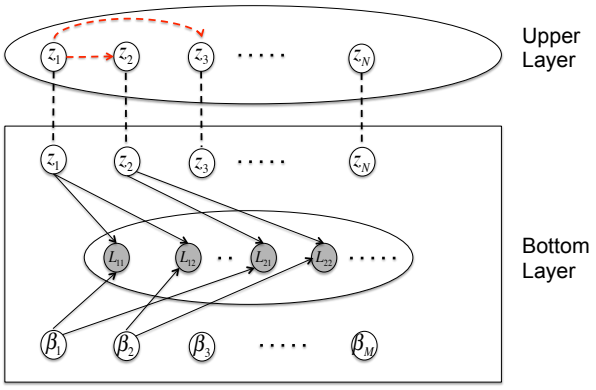

Fig. 1. Graphical Model of Trust-aware Crowdsourcing with Domain Knowledge (TCDK). z_i's are true label variables, β_j's are workers' trust variables, and l_{ij}'s are worker-provided answers. The black-dotted lines in the bottom layer encode probabilistic dependencies between variables and the red-dotted lines in the upper layer encode logical dependencies.

expressed as follows:

$$\begin{aligned} p\left(L,z,\beta|\theta\right) \propto \exp\left[-\sum_{r=1}^{R}\lambda_r\phi_r(z,\beta)\right] \\ \times \prod_{i=1}^{N}\prod_{j\in M_i} p\left(\beta_j|\theta\right) p\left(l_{ij}|z_i,\beta_j\right) \end{aligned} \tag{3}$$

where R is the number of grounded first-order logic rules. The graphical model in (3) consists of two terms with the first term encoding the logical relations among variables z,β and the second encoding probabilistic dependencies among observed answers L and hidden variables z and β. The logical dependency encoded in $\phi_r(z,\beta)$ is very general and is determined by the specific grounded rule r. For example, it can be defined over true label variables z_i's or over trust variables β_j's or over a mixture of both as in (2). Fig. 1 shows an example causal structure of TCDK when ϕ_r's are defined over z only. In Fig. 1, the statistical layer corresponds to the first term in (3) and the logical layer corresponds to the second term. The red dotted lines represent logical dependencies among z indicated by $\phi_r(z)$.

Note that β_j's are continuously-valued variables and z_i's are discrete variables. If $\phi_r(z,\beta)$'s depend on β_j's only, the first part in (3) is equivalent to a continuous Markov random field [6]. If $\phi_r(z,\beta)$'s also depend on z_i's, the first part combined with the second part in (3) can be viewed as a Hybrid Markov Logic Network (HMLN) [14]. However, HMLN relies solely on first-order logic to express causal structure among variables, therefore it falls short of expressing general dependencies as in the second term in (3).

V. Inference

We are interested in the maximum a posteriori probability (MAP) estimates of true label variables z_i's and trust variables β_j's given answers L in TCDK. The MAP estimates

are the solution to the following optimization problem:

$$\begin{aligned}\underset{z,\beta}{\operatorname{argmin}} \sum_{r=1}^{R} \lambda_r \phi_r(z,\beta) - \\ \sum_{i=1}^{N} \sum_{j \in M_i} \log p\left(l_{ij}|z_i,\beta_j\right) - \sum_{j=1}^{M} \log p\left(\beta_j|\theta\right)\end{aligned} \tag{4}$$

It is challenging to solve the above optimization problem due to the large data size and possibly exponential groundings of first-order logic rules. One can relax the discrete variables to take continuous values and resort to Alternating Optimization with Mirror Descent to avoid fully grounding first-order logic rules [1]. However, their algorithm still can not scale because a single machine is processing all the sampled groundings and the algorithm is not easily scaled to multiple machines. [2] proposed a scalable solution to constrained continuous Markov random fields based on the consensus optimization framework. However, it can not be directly applied to our problem because their optimization objective is based on hinge loss only.

In what follows, we propose a scalable inference algorithm based on the alternating direction method of multipliers (ADMM). First, we relax true label variables z_i's to the interval $[0,1]$ so that the potential functions $\phi_r(z,\beta)$'s are defined over continuous variables taking values from interval $[0,1]$. The algorithm can be seen as *generalized probabilistic soft logic* (GPSL) because it contains special cost-function rules besides first-order logic rules. We briefly review the basics of probabilistic soft logic (PSL) below.

A. Definitions in PSL

Probabilistic soft logic declares first-order logic rules:

$$\lambda : A(i,j) \wedge B(j,k) \Rightarrow C(i,k) \tag{5}$$

where A, B and C are predicates and i,j,k are variables. Each ground predicate is an instantiation of predicates with instatiated values for i,j,k and takes a soft-truth value from $[0,1]$. The logical connectives (AND, OR, NOT) are relaxed using *Lukasiewicz t-norm* and its corresponding *co-norm*:

$$\begin{aligned} p \wedge q &= \max\left(0, p+q-1\right), \\ p \vee q &= \min\left(1, p+q\right), \\ \neg p &= 1-p \end{aligned} \tag{6}$$

Any grounded first-order logic rule has the form $r_{body} \rightarrow r_{head}$. An interpretation I is defined as an assignment of soft truth values to a set of ground predicates. PSL calculates a potential function for any grounded rule r under interpretation I through the following:

$$\phi_r(I) = \max\{0, I(r_{body}) - I(r_{head})\} \tag{7}$$

B. Scalable ADMM-Based Inference

ADMM is utilized to optimize objectives by iteratively solving local subproblems and finding consensus to the global objective [3]. We observe that z_i's and β_j's are coupled through the term $\log p\left(l_{ij}|z_i,\beta_j\right)$. Therefore we can iteratively optimize (4) while fixing z_i's and vice versa. When β_j's are fixed, (4) becomes:

$$\underset{z}{\operatorname{argmin}} \sum_{r=1}^{R} \lambda_r \phi_r(z,\beta) - \sum_{i=1}^{N} \sum_{j \in M_i} \log p\left(l_{ij}|z_i,\beta_j\right) \tag{8}$$

The first term in (8) corresponds to weighted summation of potential functions of grounded first-order logic rules while the second term is the summation of logarithms of conditional probabilities. We show next that we can put the two parts into a unified framework GPSL.

$$\begin{aligned}\varphi(z_i,\beta_j) &= -\log p\left(l_{ij}|z_i,\beta_j\right) \\ &= -\mathbb{1}\{l_{ij}=z_i\}\log\beta_j - \mathbb{1}\{l_{ij}\neq z_i\}\log(1-\beta_j) \\ &= -\left(z_i l_{ij} + (1-z_i)(1-l_{ij})\right)\log\beta_j \\ &\quad -\left(1-z_i l_{ij} - (1-z_i)(1-l_{ij})\right)\log(1-\beta_j)\end{aligned} \tag{9}$$

Substituting (9) into (8), we have:

$$\underset{z}{\operatorname{argmin}} \sum_{r=1}^{R} \lambda_r \phi_r(z,\beta) + \sum_{i=1}^{N} \sum_{j \in M_i} \varphi(z_i,\beta_j) \tag{10}$$

The second term in (10) is equivalent to the summation of potential functions introduced by N grounded special cost-function rules with weight 1.0. They encode the dependency between upper and bottom layer shown in Fig. 1. Next we present how to optimize β_j's while fixing z_i's. (4) becomes:

$$\underset{\beta}{\operatorname{argmin}} \sum_{r=1}^{R} \lambda_r \phi_r(z,\beta) + \sum_{j=1}^{M} \left(\sum_{i \in N_j} \varphi(z_i,\beta_j) - \log p\left(\beta_j|\theta\right) \right) \tag{11}$$

where the first term corresponds to the summation of potential functions for grounded first-order logic rules that involve β while the second term can be viewed as the summation of potential functions introduced by M grounded special cost-function rules with weight 1.0.

Let $z_r, r=1,\ldots,R$ be a local copy of the variables in Z that are used in potential function $\phi_r(z,\beta)$ and z_{i+R} be a local copy of the variables in Z that are used in potential function $\sum_{j\in M_i}\varphi(z_i,\beta_j)$. Let $Z_i, i=1,\ldots,R+N$ be the global version of the local copies in z_i. Similarly, we define $\mathbf{b}_r, r=1,\ldots,R$ as a local copy of the global variables in $\mathbf{B}$ that are used in $\phi_r(z,\beta)$ and $\mathbf{b}_{j+R}, j=1,\ldots,M$ as a local copy of the variables in $\mathbf{B}$ used in $\sum_{i\in N_j}\varphi(z_i,\beta_j) - \log p\left(\beta_j|\theta\right), j=1,\ldots,M$. The ADMM-based inference algorithm is shown in Algorithm 1. It is scalable in nature because each grounded rule is a subproblem and can be run in parallel over multiple machines.

VI. Experiments

In order to evaluate the performance of our proposed TCDK framework, we performed experiments on two real datasets. In what follows, we describe each of the datasets, define first-order logic rules and special cost-function rules, and present experimental results. For each of the two datasets, we consider the following models for comparison:

Algorithm 1: Consensus optimization for z and β

Input: ϕ, λ, L, $\mathbf{Z}$, $\mathbf{B}, \theta, \varphi, \rho > 0$
Output: MAP estimates for z_i's and β_j's
while *not converged* **do**
  /* Optimize z_i's while fixing β_j's */
  Initialize $\mathbf{z}_i$ as a copy of the variables in $\mathbf{Z}$ that appear in $\phi_r, r = 1, \ldots, R$
  Initialize $\mathbf{z}_{i+R}$ as a copy of the variables in $\mathbf{Z}$ that appear in $\sum_{j \in M_i} \varphi(z_i, \beta_j), i = 1, \ldots, N$
  Initialize dual variable $y_k = 0, k = 1, \ldots, |R| + N$
  while *not converged* **do**
    for $k = 1, 2, \ldots, R, R+1, \ldots, R+N$ **do**
      $\mathbf{y}_k = \mathbf{y}_k + \rho\left(\mathbf{z}_k - \mathbf{Z}_k\right)$
    end
    for $r = 1, 2, \ldots, R$ **do**
      $\mathbf{z}_r \leftarrow \operatorname{argmin}_{\mathbf{z}_r \in [0,1]^{n_r}} \lambda_r \phi_r(z, \beta) + \frac{\rho}{2} \left\| \mathbf{z}_r - \mathbf{Z}_r + \frac{1}{\rho}\mathbf{y}_r \right\|_2^2$
    end
    for $i = 1, 2, \ldots, N$ **do**
      $\mathbf{z}_{i+R} \leftarrow \operatorname{argmin}_{\mathbf{z}_{i+R} \in [0,1]^{n_{i+|R|}}} \sum_{j \in M_i} \varphi(z_i, \beta_j) + \frac{\rho}{2} \left\| \mathbf{z}_{i+R} - \mathbf{Z}_{i+R} + \frac{1}{\rho}\mathbf{y}_{i+R} \right\|_2^2$
    end
    Set each entry z_i in $\mathbf{Z}$ to the average of the all the local copies.
  end
  /* Optimize β_j's while fixing z_i's */
  Initialize $\mathbf{b}_r$ as a copy of the variables in $\mathbf{B}$ that appear in $\phi_r, r = 1, \ldots, R$
  Initialize $\mathbf{b}_{j+R}$ as a copy of the variables in $\mathbf{B}$ that appear in $\sum_{i \in N_j} \varphi(z_i, \beta_j) - \log p\left(\beta_j | \theta\right), j = 1, \ldots, M$
  Initialize dual variables $\mathbf{v}_k = 0, k = 1, \ldots, M, M+1, R+M$
  while *not converged* **do**
    for $k = 1, \ldots, R, R+1, R+M$ **do**
      $\mathbf{v}_k = \mathbf{v}_k + \rho(\mathbf{b}_k - \mathbf{B}_k)$
    end
    for $r = 1, 2, \ldots, R$ **do**
      $\mathbf{b}_r \leftarrow \operatorname{argmin}_{\mathbf{b}_r \in [0,1]^{n_r}} \lambda_r \phi_r(z, \beta) + \frac{\rho}{2} \left\| \mathbf{b}_r - \mathbf{B}_r + \frac{1}{\rho}\mathbf{v}_r \right\|_2^2$
    end
    for $j = 1, 2, \ldots, M$ **do**
      $\mathbf{b}_{j+R} \leftarrow \operatorname{argmin}_{\mathbf{b}_{j+R} \in [0,1]^{n_{j+|R|}}} \sum_{i \in N_j} \varphi(z_i, \beta_j) - \log p\left(\beta_j | \theta\right) + \frac{\rho}{2} \left\| \mathbf{b}_{j+R} - \mathbf{B}_{j+R} + \frac{1}{\rho}\mathbf{v}_{i+R} \right\|_2^2$
    end
    Set each entry b_j in $\mathbf{B}$ to the average of the all the local copies.
  end
end

1) TCDK: our proposed trust-aware crowdsourcing with domain knowledge.
2) TC (trust-aware crowdsourcing without domain knowledge): same as TCDK except that we omit domain knowledge by setting zero weights to first-order logic rules and special cost-function rules defined for each dataset.
3) MV (majority vote): a true value variable is estimated to be 1 if more than half workers answer 1 and is estimated to be 0 if less than half workers answer 0. Ties are broken randomly.

A. Affective Text Evaluation

This dataset was produced by crowdsourcing task [13] where each worker was given a headline and asked to give a rating (ranging from 0 to 100) about the degree of emotions that the headline expresses. Six emotions were considered: anger, disgust, fear, joy, sadness and surprise. We use the dataset provided by [12], where 100 pieces of headlines were selected and 10 answers were solicited for each of the six emotions from workers on Amazon Mechanical Turk. Note that each headline-emotion pair might be answered by a different group of workers.

We represent a headline Q expressing emotion X as predicate $tl(Q, X)$, where $Q = 1, \ldots, N$ and $X \in \{Anger, Disgust, Fear, Joy, Sadness, Surprise\}$. The grounded predicate $tl(Q, X)$ takes value from $[0, 1]$. Our domain knowledge tells us that among those six emotions, there exists two types of relations between emotions X and

TABLE I
EMOTIONS RELATIONS

Relations	Emotion pairs
Opposite	(Anger, Joy), (Anger, Fear), (Anger, Sadness),(Anger, Surprise) (Disgust, Joy), (Disgust, Sadness), (Fear, Joy), (Sadness, Joy), (Surprise, Joy), (Surprise, Sadness)
Similar	(Fear, Sadness)

TABLE II
PERFORMANCE OF ALGORITHMS ON AFFECTIVE TEXT

Model	precision	recall	F1	accuracy
TCDK	31.91	**75.00**	**44.48**	**93.83**%
TC	**34.04**	51.61	41.03	92.33%
MV	**34.04**	47.06	39.51	91.83%

Y, one is similar relation which we define as predicate $simRel(X, Y)$ and the other is opposite relation which we define as predicate $oppRel(X, Y)$. Y takes values from the six emotions as X does. We define the following first-order logic rules to represent our domain knowledge:

$$\begin{aligned} &tl(Q, X) \wedge oppRel(X, Y) \rightarrow \neg tl(Q, Y), \quad w : 5.0 \\ &tl(Q, X) \wedge simRel(X, Y) \rightarrow tl(Q, Y), \quad w : 1.0 \end{aligned} \tag{12}$$

The first rule states that if a headline expresses emotion X and the two emotions X and Y are opposite, it is unlikely that the headline expresses emotion Y whereas the second rule states that if X and Y are similar, there is a chance that a headline expresses emotion Y if it expresses emotion X. The weights for the two first-order logic rules are assumed to be known and set to 5.0 and 1.0 respectively. Higher weight of the first rule indicates it is a more important rule than the second. The values of grounded predicates $oppRel(X, Y)$ and $simRel(X, Y)$ are assumed to be part of our domain knowledge. The details of these two grounded predicates are shown in Table I. For example, we believe that a headline can not express $Anger$ and Joy at the same time. In addition to the first-order logic rules, we define the cost-function rules:

$$LinearLoss(\beta, tl(Q, X)), w : 1.0 \tag{13}$$

The rule corresponds to the second term in (10). The predicate is called $LinearLoss$ because the potential function associated with this rule is linear in $tl(Q, X)$ as can be observed from (9) and (10).

We conducted coarse-grained experiments on Affective Text dataset, i.e. each rating is mapped to 0 if the original value is smaller than 50 and 1 if larger or equal to 50. We calculate the precision, recall, F-measure and accuracy of all emotions for the TCDK model. The results are reported in Table II. The highest scores in all the measures are in bold format. We observe that our model TCDK obtained best results with respect to recall, F1 score, and accuracy. This demonstrates the advantage of taking into consideration domain knowledge compared to TC that ignores it.

B. Fashion Social Dataset Evaluation

The dataset [9] contains 4711 images crawled from Flickr and along with each image, metadata are available such as the fashion topic used to query the image, title of the image, tags and comments made by Flickr users, etc. In this annotation task, a worker is presented with two questions for each image: Is the image fashion related? Is the image showing a specialty clothing item? Therefore we have in total 9422 questions. For each image, a number of workers from Amazon Mechanical Turk (AMT) provide their answers. Each answer takes values from $\{Yes, No, NotSure\}$. If a worker answers $NotSure$, we treat it as if the worker does not provide an answer for this question. We filter out questions that receive less than three answers and we are left with $N = 8538$ questions, each of which receives equal to or more than three answers from workers. We have in total $M = 201$ workers available for this annotation task. To generate ground truth, three trusted experts were recruited to give high-quality annotations. We take the majority vote from the three trusted experts as the ground truth and use it for evaluation of our models.

The question "Is the image related to fashion?" for image Q is denoted by predicate $fashion(Q)$, where $Q \in \{1, \ldots, N\}$ and the question "Is the image related to cloth?" for image Q by predicate $cloth(Q)$. One piece of the domain knowledge we have is that if an image is related to cloth, the image is more likely to be related to fashion. This is illustrated in Fig. 2. Knowing that the probability of the image being cloth-related is conducive to estimating whether the image is fashion-related. This observation is captured in the following rule:

$$cloth(Q) \rightarrow fashion(Q), w : 5.0 \tag{14}$$

Another observation, as shown in Fig. 3, is that if two questions are similar in terms of the metadata, then the true labels of the two questions are likely to be the same. The following rules capture the observation:

$$\begin{aligned} &sim(Q1, Q2) \wedge fashion(Q1) \rightarrow fashion(Q2), \quad w : 1.0 \\ &sim(Q1, Q2) \wedge cloth(Q1) \rightarrow cloth(Q2), \quad w : 1.0 \end{aligned} \tag{15}$$

where Q denotes an image and the predicate $sim(Q1, Q2)$ represents a question-question similarity metric. Each specific similarity metric creates an instance of the two rules in (15).

We propose to use a context-based similarity metric. An image context refers to a group photo pool or a photoset. One of the example contexts is Artistic Photography. An image can be associated with one or more contexts. The intuition is that if two pictures are more likely to be in the same context, then they tend to have the same label as well. We denote C_1 as the context set for $Q1$ and C_2 as the context set for $Q2$. The context-based similarity score $sim(Q1, Q2)$ is defined as:

$$sim(Q1, Q2) = \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|} \tag{16}$$

TABLE III

PERFORMANCE OF ALGORITHMS ON FASHION DATASET

Model	precision	recall	F1	accuracy
TCDK	**87.83**	**89.84**	**88.82**	**89.27%**
TC	86.79	83.6	85.20	85.39%
MV	86.55	83.73	85.11	85.32%

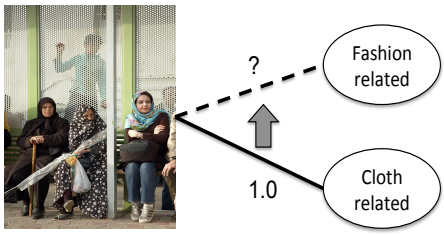

Fig. 2. Estimated true labels for "cloth related" questions can be used for prediction of "fashion related" questions.

To avoid quadratic groundings of $sim(Q1, Q2)$, for each question $Q1$ we only keep pairs $(Q1, Q2)$'s whose similarity scores in (16) rank at the top 10 as in [5]. Similar to the model for the Affective Text dataset, the special cost-function rules that bridge the gap between the top and the bottom layers are:

$$\begin{aligned} &LinearLoss(\beta, fashion(Q)), \quad w : 1.0 \\ &LinearLoss(\beta, cloth(Q)), \quad w : 1.0 \end{aligned} \tag{17}$$

We perform ten-fold cross validation with each fold leaving out 10% of data. We estimate values of $fashion(Q)$ and $cloth(Q)$ on the held-out fold and then map them to 0 or 1 using threshold 0.5. The results are shown in Table III. Again, results show that TCDK achieves better performance in all criteria with integrated domain knowledge than TC alone (without the leverage of domain knowledge).

The weights of first-order logic rules defined for both datasets in Section VI-A and Section VI-B are assumed to be known and given as part of domain knowledge in this paper. Though weights can be auto-tuned using maximum-likelihood estimation [6], we leave this problem for future work and aim to demonstrate the power of our model with fixed yet not fine-tuned values set by users of our model.

VII. CONCLUSION

We presented trust-aware crowdsourcing with domain knowledge (TCDK), a unifying framework that combines the power of domain knowledge and traditional crowdsourcing graphical model. It allows users to express domain knowledge using first-order logic rules without redefining the model. To estimate questions' true labels and workers' trust values, we develop a scalable inference algorithm based on alternating consensus optimization. We demonstrate that our model is superior to the state-of-the-art by testing it on two real datasets.

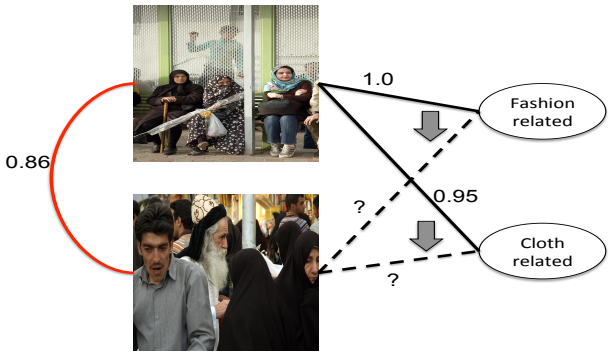

Fig. 3. Estimated true labels for questions can be used for prediction of other questions using image similarity.

REFERENCES

[1] D. Andrzejewski, X. Zhu, M. Craven, and B. Recht. A framework for incorporating general domain knowledge into latent dirichlet allocation using first-order logic. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1171, 2011.

[2] S. Bach, M. Broecheler, L. Getoor, and D. O'leary. Scaling mpe inference for constrained continuous markov random fields with consensus optimization. In *Advances in Neural Information Processing Systems*, pages 2654–2662, 2012.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

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

[5] S. Fakhraei, B. Huang, L. Raschid, and L. Getoor. Network-based drug-target interaction prediction with probabilistic soft logic. 2014.

[6] A. Kimmig, S. Bach, M. Broecheler, B. Huang, and L. Getoor. A short introduction to probabilistic soft logic. In *Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications*, pages 1–4, 2012.

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

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

[9] B. Loni, M. Menendez, M. Georgescu, L. Galli, C. Massari, I. S. Altingovde, D. Martinenghi, M. Melenhorst, R. Vliegendhart, and M. Larson. Fashion-focused creative commons social dataset. In *Proceedings of the 4th ACM Multimedia Systems Conference*, pages 72–77. ACM, 2013.

[10] G.-J. Qi, C. C. Aggarwal, J. Han, and T. Huang. Mining collective intelligence in diverse groups. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1041–1052. International World Wide Web Conferences Steering Committee, 2013.

[11] 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.

[12] 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.

[13] C. Strapparava and R. Mihalcea. Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 70–74. Association for Computational Linguistics, 2007.

[14] J. Wang and P. Domingos. Hybrid markov logic networks. In *AAAI*, volume 8, pages 1106–1111, 2008.