Learning Effective Embeddings From Crowdsourced Labels: An Educational Case Study

Guowei Xu, Wenbiao Ding
TAL AI Lab
Beijing, China
{xuguowei, dingwenbiao}@100tal.com

Jiliang Tang
Michigan State University
East Lansing, MI USA
tangjili@msu.edu

Songfan Yang, Gale Yan Huang, Zitao Liu*
TAL AI Lab
Beijing, China
{yangsongfan, galehuang, liuzitao}@100tal.com

Abstract—Learning representation has been proven to be helpful in numerous machine learning tasks. The success of the majority of existing representation learning approaches often requires a large amount of consistent and noise-free labels. However, labels are not accessible in many real-world scenarios and they are usually annotated by the crowds. In practice, the crowdsourced labels are usually inconsistent among crowd workers given their diverse expertise and the number of crowdsourced labels is very limited. Thus, directly adopting crowdsourced labels for existing representation learning algorithms is inappropriate and suboptimal. In this paper, we investigate the above problem and propose a novel framework of Representation Learning with crowdsourced Labels, i.e., "RLL", which learns representation of data with crowdsourced labels by jointly and coherently solving the challenges introduced by limited and inconsistent labels. The proposed representation learning framework is evaluated in two real-world education applications. The experimental results demonstrate the benefits of our approach on learning representation from limited labeled data from the crowds, and show RLL is able to outperform state-of-the-art baselines. Moreover, detailed experiments are conducted on RLL to fully understand its key components and the corresponding performance.

Index Terms—representation learning, education data mining, crowdsourcing

I. INTRODUCTION

It has been suggested that the performance of machine learning applications strongly relies on the representation of the input data [1]. A good data representation provides tremendous flexibilities to choose fast and simple models. However, the raw data representation is not typically amenable to learning [2]. Representation learning aims to automatically learn new data representation from raw features by discovering hidden patterns in the data and has attracted increasing attentions in recent years [1]. Representation learning, especially deep learning [3] has been immensely advanced the field of machine learning [4] and the related areas such as computer vision [5], signal processing [6] and natural language processing [7]. On the other hand, modern successful representation learning approaches are discriminatively trained and often require massive labeled data, which is typically unavailable in many real-world scenarios [8].

To bridge this gap, human efforts are needed to acquire labeled data manually and crowdsourcing provides a flexible solution. Theoretically, we can obtain a labeled dataset as large as we want via crowdsourcing platforms such as Amazon Mechanical Turk, Figure Eight, etc. However, in practice, the amount of crowdsourced labels for a given task can be *limited* due to a variety of reasons. For instance, a limited budget prevents us from affording massive labeled data. Another example is in some domains such as healthcare, privacy concerns or a paucity of data leads to very limited crowdsourced labels. Furthermore, crowd workers are unlikely to be experts and they tend to have different levels of expertise. As a consequence, crowdsourced labels can be very *inconsistent*. In other words, two crowd workers can annotate the same object with distinct labels. Given the aforementioned properties of crowdsourced labels, the majority of existing representation learning techniques cannot work appropriately and optimally with crowdsourced labels in practice.

This problem becomes more critical in building ML models in educational scenarios. The difficulties are two-fold: first, label annotation in educational scenarios usually requires more domain knowledge compared to standard crowdsourcing tasks such as image classification, part-of-speech tagging, etc. It is more ambiguous when labeling a 60-min class (whether the class quality is good or bad) than annotating images. This will lead to very inconsistent labels. Second, labeling each sample in educational scenarios requires much more efforts than standard annotation tasks. For example, it may take a crowd worker less than 1 second to annotate an image while the worker has to watch a 60-min video before determining the class quality.

Recent years have witnessed great efforts on learning with *limited* labeled data [8], [9]. Also inferring true labels from *inconsistent* crowdsourced labels has been studied for decades [10], [11]. However, research on representation learning with limited and inconsistent crowdsourced labels is rather limited. Thus, in this paper, we study the problem of representation learning with crowdsourced labels. In particular, we target on answering two questions: (1) how to take advantage of crowdsourced labels under the limited and inconsistent settings? and (2) how to build a unified representation learning framework with crowdsourced labels?

In this work, we propose a unified framework, i.e., RLL, to jointly solve problems of learning representation from inconsistent and limited labeled data. In our unified framework, we propose a scheme to generate hundreds of thousands of

---

*Corresponding author: Zitao Liu.

training instances from only a limited number of labeled data from crowd workers. Furthermore, instead of isolating true label inference from the representation learning process, we use Bayesian inference to estimate the label confidence and integrate the confidence estimation process into the model learning. Our framework's effectiveness is demonstrated in two real-world scenarios with very limited crowdsourced labels. Further experiments are conducted to fully understand the key model components of RLL.

Our major contributions are two-fold: first, we propose a unified framework, i.e., RLL, to jointly solve problems of learning representation from inconsistent and limited labeled data. Second, we conduct experiments on crowdsourced labels in two real-world education applications to fully understand the effectiveness of RLL.

II. RELATED WORK

A. Learning with limited Labeled Data

Representation learning, especially deep learning has largely advanced the field of machine learning and its applications. Such success typically requires a large amount of labeled data, which is usually unavailable in many domains. Various types of techniques have been developed to enable learning with limited labeled data and next we will review representative techniques. One is few shot learning, which aims to learn new concepts from only a few labeled examples [8], [12]. When large unlabeled datasets are available, techniques have been developed to make use of weak and distant supervision [13] such as higher-level abstractions [14], biased or noisy labels from distant supervision [15] and data augmentation [16]. Another popular technique for learning from limited labels is transfer learning [17], which aims to apply knowledge learned in the source domain to a related target domain.

B. Crowdsourced Labels

Crowdsourcing offers a flexible way to get labeled data for model learning. Due to the fact that crowd workers have different levels of expertise, the crowdsourced labels are often inconsistent, which can compromise practical applications [18]. Therefore, one key problem is to infer true labels from crowdsourced labels. An EM algorithm is proposed to estimate the error rates when patients answer medical questions with repeated but conflicting responses [10]. Inspired by Dawid and Skene [10], Whitehill et al. considered item difficulty for image classification and a score for each annotator is extracted to assess the quality of the annotator [19]. Aforementioned approaches can infer the true labels independently, which can be sub-optimal for the targeted tasks. Hence, there are increasing attention on combining true label inference with the targeted machine learning tasks. Raykar et al. proposed an EM algorithm to jointly learn the levels of annotators and the regression models [20]. Likewise, there are efforts to embed label inference process into other types of models. Rodrigues, Pereira, and Ribeiro generalized Gaussian process classification to consider multiple annotators with diverse expertise [21]. Rodrigues et al. studied supervised topic models

for classification and regression from crowds [22]. Albarqouni et al. introduced an additional crowdsourcing layer to embed the data aggregation process into convolutional neural network learning [23]. Very recently, techniques have been studied, which do not need iterative EM algorithms to estimate weights of the annotators. Guan et al. captured information about the annotators by modeling each annotator individually and then learning combination weights via back propagation [24].

The majority of aforementioned algorithms have been designed to address the problems of noise and inconsistencies in crowdsourced labels and they cannot work as expected when labels are limited, especially for these algorithms developed for deep learning. While in this work, we aim to develop algorithms which can jointly solve the challenges from limited and inconsistent labels.

III. METHODOLOGY

In this section, we will give details about the proposed framework, which aims to jointly address the challenges from limited and inconsistent crowdsourced labels. Before that, we will introduce notations.

In this work, we denote the feature vector of ith example as $\mathbf{x}_i$ and its corresponding crowdsourced labels are $y_{i,1}, y_{i,2}, \cdots, y_{i,d}$, where d is the number of crowd workers to annotate each sample. Without loss of generality, we assume crowdsourced labels are binary, i.e., $y_{i,j} \in \{0, 1\}$. We use $(\cdot)^+$ and $(\cdot)^-$ to indicate positive and negative examples. We represent the entire data $\mathcal{D}$ as a collection of positive ($\mathcal{D}^+$) and negative examples ($\mathcal{D}^-$).

The proposed representation learning framework RLL can learn embeddings from limited training data with crowdsourced labels. It is composed of two key components:

- *a grouping deep architecture* that learns effective representations from very limited training data.
- *a Bayesian confidence estimator* that captures the inconsistency among crowdsourced labels and uses Bayesian inference to integrate the label confidence into the learning process.

A. Grouping Based Deep Architecture

In many real-world crowdsourcing application scenarios, annotated labels from the crowds are very limited due to many reasons such as budget constraints, privacy concerns, etc. This may easily lead to the overfitting problems for many deep representation models and make them inapplicable. To address this issue, instead of directly training the discriminative representation models from the small amount of annotated labels, we develop a grouping based deep architecture to reassemble and transform limited labeled examples into many training groups. We would like to include both positive and negative examples into each group. Within each group, we maximize the conditional likelihood of one positive example given another positive example and at the same time, we minimize the conditional likelihood of one positive example given several negative examples. Different from traditional metric learning approaches that focus on learning distance

between pairs, our approach aim to generate a more difficult scenario that considers not only the distances between positive examples but distances between negative examples.

More specifically, for each positive example $\mathbf{x}_i^+$, we select another positive example $\mathbf{x}_j^+$ from $\mathcal{D}^+$, where $\mathbf{x}_i^+ \neq \mathbf{x}_j^+$. Then, we randomly select k negative examples from $\mathcal{D}^-$, i.e., $\mathbf{x}_1^-, \mathbf{x}_2^-, \cdots, \mathbf{x}_k^-$. After that, we create a group $\mathbf{g}_i$ by combining the positive pair and the k negative examples, i.e., $\mathbf{g}_i = <\mathbf{x}_i^+, \mathbf{x}_j^+, \mathbf{x}_1^-, \mathbf{x}_2^-, \cdots, \mathbf{x}_k^->$. By using the grouping strategy, we can create $O(|\mathcal{D}^+|^2 \cdot |\mathcal{D}^-|^k)$ groups for training theoretically, where $|\mathcal{D}^+|$ and $|\mathcal{D}^-|$ are the number of positive and negative examples in the original labeled data. Let $\mathcal{G}$ be the entire collection of groups, i.e., $\mathcal{G} = \{\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_n\}$ where n is the total number of groups.

After the grouping procedure, we treat each group $\mathbf{g}_i$ as a training example and feed $\mathbf{g}_i$s into a typical deep neural network (DNN) for learning robust embeddings. The input to the DNN is raw features extracted from each example and the output of the DNN is a low-dimensional semantic feature vector. Inside the DNN, we use the multi-layer fully-connected non-linear projections to learn the compact representations as shown in Figure 1.

Model Learning Inspired by the discriminative training approaches in language processing and information retrieval, we propose a supervised training approach to learn our model parameters by maximizing the conditional likelihood of retrieving positive example $\mathbf{x}_j^+$ given positive example $\mathbf{x}_i^+$ from group $\mathbf{g}_i$.

More formally, let $\mathbf{f}_*$ be the learned representation of $\mathbf{x}_*$ from DNN, where $\mathbf{x}_* \in \mathbf{g}_i$. Similarly, $\mathbf{f}_i^+$ represent the embeddings of the positive example $\mathbf{x}_i^+$. Then, the semantic relevance score between two representations in the embedding space within a group is measured as $r(\mathbf{x}_i^+, \mathbf{x}_*) \overset{\text{def}}{=} \text{cosine}(\mathbf{f}_i^+, \mathbf{f}_*)$.

In our representation learning framework, we compute the posterior probability of $\mathbf{x}_j^+$ in group $\mathbf{g}_i$ given $\mathbf{x}_i^+$ from the cosine relevance score between them through a softmax function

$$p(\mathbf{x}_j^+|\mathbf{x}_i^+) = \frac{\exp\left(\eta \cdot r(\mathbf{x}_i^+, \mathbf{x}_j^+)\right)}{\sum_{\mathbf{x}_* \in \mathbf{g}_i, \mathbf{x}_* \neq \mathbf{x}_i^+} \exp\left(\eta \cdot r(\mathbf{x}_i^+, \mathbf{x}_*)\right)}$$

where η is a smoothing hyper parameter in the softmax function, which is set empirically on a held-out dataset in our experiment.

Hence, given a collection of groups $\mathcal{G}$, we optimized the DNN model parameters by maximizing the sum of log conditional likelihood of finding a positive example $\mathbf{x}_j^+$ given the paired positive example $\mathbf{x}_i^+$ from group $\mathbf{g}_i$, i.e., $\mathcal{L}(\Omega) = -\sum_{i=1}^{n} \log p(\mathbf{x}_j^+|\mathbf{x}_i^+)$, where Ω is the parameter set of the DNN. Since $\mathcal{L}(\Omega)$ is differentiable with respect to Ω, we use gradient based optimization approach to train the DNN.

B. Bayesian Confidence Estimator

When obtaining labels from the crowds, each example is usually labeled by d workers and by nature the crowdsourced labels are not consistent. Moreover, it is unrealistic to hire a large amount of crowd workers to label the same example multiple times to avoid the inconsistency via majority vote. In most cases, even though two examples are both identified as positive, the confidence of their "positiveness" might be different. For example, assuming $\mathbf{x}_1^+$ and $\mathbf{x}_2^+$ are two positive examples whose corresponding 5-person crowdsourced labels are $(1, 1, 1, 1, 1)$ and $(1, 1, 1, 0, 0)$. Apparently, our assurance of their labels should be different and we should consider such crowdsourced labeling inconsistency into our model training process. In this work, we model such inconsistency as the confidence about the crowdsourced labels. Let δ_i be the confidence of each example $\mathbf{x}_i$.

One obvious approach is to treat the confidence (δ_i) as a random variable that follows the Bernoulli distribution and obtain δ_i by using maximum likelihood estimation (MLE) as follows:

$$\delta_i^{\text{MLE}} = \sum_{j=1}^{d} y_{i,j}/d \tag{1}$$

However, in many real-world setting, we are not able to afford too many crowd workers to label the same example simultaneously, i.e, d is relatively small. This leads to the inferior performance in the MLE approach (eq.(1)). To address this problem, we assign a Beta prior to δ_i, i.e., $\delta_i \sim \text{Beta}(\alpha, \beta)$. Therefore, the posterior estimation of the crowdsourced label confidence is

$$\delta_i^{\text{Bayesian}} = \frac{\alpha + \sum_{j=1}^{d} y_{i,j}}{\alpha + \beta + d} \tag{2}$$

After that, we integrate the crowdsourced label confidence into our representation learning. The confidence weighted conditional probability is defined as follows:

$$\hat{p}(\mathbf{x}_j^+|\mathbf{x}_i^+) = \frac{\exp\left(\eta \cdot \delta_j \cdot r(\mathbf{x}_i^+, \mathbf{x}_j^+)\right)}{\sum_{\mathbf{x}_* \in \mathbf{g}_i, \mathbf{x}_* \neq \mathbf{x}_i^+} \exp\left(\eta \cdot \delta_* \cdot r(\mathbf{x}_i^+, \mathbf{x}_*)\right)} \tag{3}$$

where δ_j and δ_* are confidence scores of $\mathbf{x}_j^+$ and $\mathbf{x}_*$. Accordingly, we adjust the objective function by using the confidence weighted conditional probability (eq.(3)).

C. Model Summary

In our RLL framework, given the limited data with crowdsourced labels, we first generate a fair large amount of groups of training examples by including both positive and negative examples. Then, we estimate the label confidence for each crowdsourced data by a Bayesian estimator. After that, we feed all groups into a DNN which maximizes the confidence-weighted conditional likelihood of retrieving the positive examples. The entire RLL framework is illustrated in Figure 1.

IV. EXPERIMENTS

In this section, we evaluate our approach on two real-world crowdsourced datasets. We first introduce the experimental setting, next validate the effectiveness of the proposed framework by comparing with representative baselines and finally study the important parameters of the proposed framework. To encourage the reproducible results, we
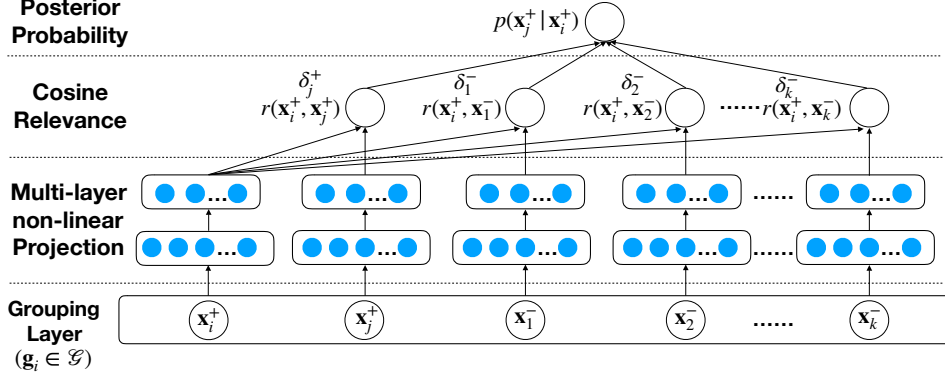
Fig. 1. The overview of the RLL framework.

make our code publicly available at: https://github.com/tal-ai/Representation-Learning-with-crowdsourced-Labels.

A. Experimental Setting

To assess the proposed framework, we conduct several experiments on two real-world datasets.

- **Oral Math Questions**("oral") We collect 880 audio files about oral math questions from students in grade 2. In each wav, a student talks about his or her entire thinking process of solving a math question. Our task is to predict whether the student's entire speech is fluent.
- **Online 1v1 Class Qualities** ("class") We collect 472 videos from public online 1v1 classes where a teacher is teaching classes to a student on a live broadcast platform. The task is to predict the quality of the entire class based on the interactions between the teacher and the student, whether the students take notes, etc. The average length of our class video files is 65 minutes.

In this work, each example in both *oral* and *class* are annotated by five annotators. The annotation tasks for both datasets are binary classification tasks in which the workers only need to assign 1 or 0 to instances. 1 represents fluent and good quality in *oral* and *class* respectively. Please note that data annotation for multimedia data (audio and video data) is very time-consuming. For example, to label an example in *class*, the worker has to watch the entire 65-minute video before giving the label. In addition, experts are also asked to annotate these two datasets and expert labels are considered as the ground truth for evaluation purpose. For expert labels, in *oral* dataset, the positive (students' speeches are fluent) over negative (students' speeches are influent) samples ratio is 1.8. In *class* dataset, the positive (the class is in good quality) over negative (the class is not in good quality) sample ratio is 2.1.

Following the tradition to assess representation learning algorithms, we evaluate the classification performance via accuracy, and F1 score. We choose logistic regression as the basic classifier. For each task, we conduct a 5-fold cross validation on the datasets and report the average performance.

We use label class prior to set the hyper parameters α and β in eq.(2) for both *oral* and *class*.

B. Performance Comparison

To assess the effectiveness of the proposed framework, we carefully choose three groups of state-of-the-art as our baselines.

1) Group 1: True Label Inference from Crowdsourcing: The first group contains methods inferring true labels from crowdsourced labels. For examples in both *oral* and *class* datasets, we extracted a wide range of linguistic features from the raw texts after having automatic speech recognition on the videos. They are listed as follows:

- Logistic regression with every pair (instance, label) provided by each crowd worker as a separate example. Note that this amounts to using a soft probabilistic estimate of the actual ground truth to learn the classifier, i.e., *SoftProb* [20].
- Logistic regression with EM labels, i.e., *EM* [25]. The labels are treated as hidden variables and inferred by expectation-maximization. The EM algorithm iteratively estimated worker's accuracy and exploited the estimated accuracy to compute the aggregated result.
- Logistic regression with GLAD labels, i.e., *GLAD*. GLAD infers the true labels by jointly inferring the true label, worker's expertise and the difficulty of each data instance [19].

2) Group 2: Representation Learning with Limited Labels: The second group includes representation learning methods designed for limited labels. We use the majority vote from the crowdsourced labels to infer the true labels. They are listed as follows:

- Siamese networks, i.e., *SiameseNet* [26]. We train a siamese network that takes a pair of examples and trains the embeddings so that the distance between them is minimized if they're from the same class and is greater than some margin value if they represent different classes.
- Triplet networks, i.e., *TripleNet* [27]. We train a triplet network that takes an anchor, a positive (of same class as

an anchor) and negative (of different class than an anchor) examples. The objective is to learn embeddings such that the anchor is closer to the positive example than it is to the negative example by some margin value.

- Relation network for few-Shot learning, i.e., *RelationNet* [28]. The RelationNet learns a deep distance metric to compare a small number of images within episodes.

3) Group 3: Two-stage Models by Combining Group 1 and Group 2: The third group are methods combining baselines from the first (i.e., inferring true labels) and second groups (i.e., learning embedding with limited labels). They solve the problems of the limited and inconsistent labels in two stages. Due to the page limit, we only combine the best approaches from first and second groups.

4) Group 4: Our Methods: We also create some variants of our RLL framework, i.e., RLL, RLL-MLE and RLL-Bayesian, as follows:

- Learning representation by using RLL without Bayesian confidence score and true labels inferred from majority vote, i.e., RLL.
- Learning representation by using RLL with confidence score estimated by MLE, i.e., RLL-MLE.
- Learning representation by using RLL with confidence score estimated by Bayesian approach, i.e., RLL-Bayesian.

C. Experimental Results

The accuracy and F1 score performance on the *oral* and *class* dataset are demonstrated in Table I. From these results, we make the following observations:

- Methods in **group 3**, which solve the problem of limited and inconsistent labels by combining baselines from **group 1** and **group 2**, tend to obtain better performance than the corresponding individual baselines that address either only inconsistent labels in **group 1** or only limited labels in **group 2**. These results suggest that solving the two problems about crowdsourced labels together is necessary and can benefit the applications which make use of crowdsourced labels.
- The proposed frameworks perform better than methods in **group 3**. Methods in **group 3** are two-stage algorithms; while our representation framework frameworks jointly solve the problems from limited and inconsistent labels in a unified and coherent manner.
- RLL-Bayesian always outperforms RLL-MLE and RLL. Crowdsourcing labels usually are inconsistent due to the different backgrounds and education levels of crowd workers. Meanwhile, it is unrealistic and impractical to solve the inconsistency issue by hiring an "infinite" group of crowd workers. As we can see from the results, both RLL-Bayesian and RLL-MLE show better performance compared with RLL, which indicates the necessity to consider the confidence of crowdsourcing labels when learning from crowdsourced labels. Comparing RLL-MLE and RLL-Bayesian, RLL-Bayesian achieves better

performance. Because of the limited number of crowd-sourced labels for each example, the label confidence estimation cannot purely rely on MLE and we should utilize the prior knowledge to guide the confidence estimation.

- Performance of methods in **group 3** vary a lot and sometimes are much worse than methods in **group 1**. Modern representation learning approaches such as SiameseNet rely heavily on the complex neural network and massive training samples. When the training samples become limited, their performance cannot be guaranteed and may easily run into the overfitting problem and could be inferior compared to classic methods in **group 1**.

TABLE I
PREDICTION RESULTS ORAL AND CLASS DATASETS.

Method	Group	Oral Data		Class Data	
		Accuracy	F1	Accuracy	F1
SoftProb	group 1	0.815	0.869	0.758	0.810
EM	group 1	0.843	0.887	0.606	0.698
GLAD	group 1	0.831	0.881	0.697	0.773
SiameseNet	group 2	0.802	0.859	0.719	0.836
TripletNet	group 2	0.847	0.889	0.750	0.857
RelationNet	group 2	0.843	0.890	0.730	0.842
SiameseNet+EM	group 3	0.798	0.856	0.727	0.842
SiameseNet+GLAD	group 3	0.815	0.871	0.727	0.842
TripletNet+EM	group 3	0.843	0.887	0.727	0.842
TripletNet+GLAD	group 3	0.843	0.890	0.667	0.792
RelationNet+EM	group 3	0.860	0.899	0.727	0.842
RelationNet+GLAD	group 3	0.854	0.889	0.730	0.842
RLL	group 4	0.871	0.901	0.818	0.880
RLL+MLE	group 4	0.871	0.903	0.848	0.902
RLL+Bayesian	group 4	**0.888**	**0.915**	**0.879**	**0.920**

To sum up, the proposed frameworks significantly benefit the classification performance. Next, we design experiments to further understand the working of the proposed frameworks. Since RLL-Bayesian always obtain the the best performance among the three variants, the following investigations are based on RLL-Bayesian.

D. The Impact of Negative Examples

One important parameter of the proposed framework is the number of negative examples k we include in each group g_i. To understand the impact of negative examples, we examine how the performance changes when we vary k as $\{2, 3, 4, 5\}$. The performance variances of the proposed framework w.r.t. k are demonstrated in Table II for the *oral* and *class* datasets respectively.

TABLE II
RLL-BAYESIAN RESULTS WITH DIFFERENT kS.

k	Oral Data		Class Data	
	Accuracy	F1	Accuracy	F1
2	0.809	0.852	0.699	0.813
3	**0.888**	**0.915**	**0.879**	**0.920**
4	0.831	0.875	0.757	0.855
5	0.803	0.851	0.750	0.846

From Table II, we can observe that in general, with the increase of k, the performance tends to first increase and

then decrease. When k is from 2 to 3, the performance increases remarkably. We can have more groups (or training samples) with $k = 3$ compared to $k = 2$. However, when we increase $k = 3$ to $k = 4$ and $k = 5$, the performance decreases dramatically. Though by increasing k, we can get more training samples, we also may introduce noise.

E. Impact of the Number of Crowd Workers

Another important parameter of the proposed framework is the number of crowd workers d to annotate each data sample. Similar to the analysis on k, we check the performance changes by choosing d from $\{1, 3, 5\}$. The results are demonstrated in Table III for *oral* and *class* datasets. As shown in Table III, with the increase of d, the performance consistently increases. With more crowd workers to label each data instance, our model is more likely to estimate the trustworthiness of crowdsourced labels. The observation can help us determine the reasonable number of crowd workers for crowdsourcing in practice.

TABLE III
RLL-BAYESIAN RESULTS WITH DIFFERENT dS.

d	Oral Data		Class Data	
	Accuracy	F1	Accuracy	F1
1	0.826	0.873	0.727	0.842
3	0.876	0.922	0.758	0.840
5	**0.888**	**0.915**	**0.879**	**0.920**

V. CONCLUSION

In this work, we study the problem of representation learning with crowdsourced labels. We design a novel representation learning framework RLL for crowdsourced labels under the limited and inconsistent settings. Experimental results on two real-world applications demonstrate (1) the proposed framework outperforms the representative baselines; and (2) it is necessary to address the limited and inconsistent label problems simultaneously. Our current model does not make use of any information about individual crowd worker and we want to extend the proposed framework to incorporate such information in the future.

REFERENCES

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *PAMI*, vol. 35, no. 8, pp. 1798–1828, 2013.
[2] P. Domingos, "A few useful things to know about machine learning," *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, 2012.
[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.
[4] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1.
[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105.
[6] D. Yu and L. Deng, "Deep learning and its applications to signal and information processing [exploratory dsp]," *IEEE Signal Processing Magazine*, vol. 28, no. 1, pp. 145–154, 2011.
[7] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *ICML*. ACM, 2008, pp. 160–167.
[8] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *PAMI*, vol. 28, no. 4, pp. 594–611, 2006.
[9] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*. Association for Computational Linguistics, 2009, pp. 1003–1011.
[10] A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the em algorithm," *Applied statistics*, pp. 20–28, 1979.
[11] P. G. Ipeirotis, F. Provost, and J. Wang, "Quality management on amazon mechanical turk," in *Proceedings of the ACM SIGKDD workshop on human computation*. ACM, 2010, pp. 64–67.
[12] S. Jadon and A. A. Srinivasan, "Siamese networks for one shot learning using kernel based activation functions."
[13] A. J. Ratner, C. M. De Sa, S. Wu, D. Selsam, and C. Ré, "Data programming: Creating large training sets, quickly," in *NIPS*, 2016, pp. 3567–3575.
[14] S. Takamatsu, I. Sato, and H. Nakagawa, "Reducing wrong labels in distant supervision for relation extraction," in *ACL*. Association for Computational Linguistics, 2012, pp. 721–729.
[15] T. Liu, K. Wang, B. Chang, and Z. Sui, "A soft-label method for noise-tolerant distantly supervised relation extraction," in *EMNLP*, 2017, pp. 1790–1795.
[16] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *PAMI*, vol. 38, no. 9, pp. 1734–1747, 2016.
[17] S. J. Pan, Q. Yang *et al.*, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
[18] V. S. Sheng, F. Provost, and P. G. Ipeirotis, "Get another label? improving data quality and data mining using multiple, noisy labelers," in *SIGKDD*. ACM, 2008, pp. 614–622.
[19] J. Whitehill, T.-f. Wu, J. Bergsma, J. R. Movellan, and P. L. Ruvolo, "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise," in *NIPS*, 2009, pp. 2035–2043.
[20] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *JMLR*, vol. 11, no. Apr, pp. 1297–1322, 2010.
[21] F. Rodrigues, F. Pereira, and B. Ribeiro, "Gaussian process classification and active learning with multiple annotators," in *ICML*, 2014, pp. 433–441.
[22] F. Rodrigues, M. Lourenco, B. Ribeiro, and F. C. Pereira, "Learning supervised topic models for classification and regression from crowds," *PAMI*, vol. 39, no. 12, pp. 2409–2422, 2017.
[23] S. Albarqouni, C. Baur, F. Achilles, V. Belagiannis, S. Demirci, and N. Navab, "Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1313–1321, 2016.
[24] M. Y. Guan, V. Gulshan, A. M. Dai, and G. E. Hinton, "Who said what: Modeling individual labelers improves classification," *arXiv preprint arXiv:1703.08774*, 2017.
[25] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.
[26] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Deep Learning Workshop*, vol. 2, 2015.
[27] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015, pp. 815–823.
[28] F. S. Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," 2018.