Learning to Predict Population-Level Label Distributions

Tong Liu
Rochester Institute of Technology
Rochester, New York
tl8313@rit.edu

Akash Venkatachalam
Rochester Institute of Technology
Rochester, New York
av2833@rit.edu

Pratik Sanjay Bongale
Gleason Corporation
Rochester, New York
pbongale@gleason.com

Christopher M. Homan
Rochester Institute of Technology
Rochester, New York
cmh@cs.rit.edu

ABSTRACT

Machine learning problems are often subjective or ambiguous. That is, humans solving the same problems might come to legitimate but completely different conclusions, based on their personal experiences and beliefs. In supervised learning, particularly when using crowdsourced training data, multiple annotations per data item are usually reduced to a single label representing ground truth. This hides a rich source of diversity and subjectivity of opinions about the labels. Label distribution learning associates for each data item a probability distribution over the labels for that item, thus can preserve the diversity that conventional learning hides or ignores. We introduce a strategy for learning label distributions with only five-to-ten labels per item by aggregating human-annotated labels over multiple, semantically related data items. Our results suggest that specific label aggregation methods can help provide reliable representative semantics at the population level.

CCS CONCEPTS

• **Theory of computation → Unsupervised learning and clustering**; • **Human-centered computing** → *Empirical studies in collaborative and social computing*; • **Computing methodologies** → Classification and regression trees.

KEYWORDS

Subjectivity in crowdsourcing, label distribution learning, clustering and classification, social media.

ACM Reference Format:
Tong Liu, Akash Venkatachalam, Pratik Sanjay Bongale, and Christopher M. Homan. 2019. Learning to Predict Population-Level Label Distributions. In *Companion Proceedings of the 2019 World Wide Web Conference (WWW '19 Companion), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3308560.3317082

1 INTRODUCTION

The goal of many supervised learning problems is to map each given data item to a single (or set of, but in any case, deterministic) label(s) according to some standard of ground truth. However, many real-world problems—such as those related to color, pain, taste, level of danger, or qualitative analysis—have different answers depending on whom is asked, even when the domain of answers is fixed (i.e., *closed domain*) or more than one answer is allowed (i.e., *multilabel*). In such cases, a single (set of) label(s) does not meaningfully solve the problem, or may hide important dissenting beliefs or opinions. Yet the impact of AI agents that fail to recognize diversity in a representative fashion ranges from banal to harmful on a societal level.

For instance, in 2016 contestants from over 100 countries from around the world submitted images of themselves to Beauty.ai's website, and their proprietary deep learning agent, trained on publicly available facial images, chose winners in 44 different beauty pageant categories [28]. Yet the algorithm, perhaps due to biases in the trained data, showed strong signs of racial bias: 37 of the winners had distinctly European facial features [28]. Microsoft built a Twitter bot called Tay that was supposed to learn new language skills, but it had to be shut down soon after launch because it learned to deny the holocaust and demonize feminism [36]. ProPublica reported that Northpointe risk assessment software, used to by judges in Florida to help determine incarceration lengths, systematically assigned higher risk scores to black defendants than to white ones [31].

Label distribution learning (LDL) is a recent approach that replaces the goal of predicting, for each data item, a single (set of) label(s) with the more challenging and complex task of predicting a probability distribution (known as a *label distribution*) over the label choices [13]. A growing body of work has used this approach, e.g., to predict beauty in images [31] and rate movies [14], leading to more nuanced prediction results and insights. There is also evidence that, even in situations where ground truth exists but is difficult to obtain, predicting label distributions is more informative and accurate than aggregating the opinions of multiple labelers into a single (set of) discrete choice(s) [16].

A major resource bottleneck in LDL is the amount of human annotations needed, since for any large population of labelers, the number m of labels needed to estimate (i.e., taken as a sample of) the underlying population's true distribution of beliefs for even one data item is rather large, depending on the size of the label space and the degree of confidence needed. But the number of data items n needed for supervised learning normally runs into the thousands. Thus, taken independently, the total number $m \times n$ of human labels needed for training on label distributions grows quadratically in

WWW '19 Companion, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6675-5/19/05.
https://doi.org/10.1145/3308560.3317082

two variables that are typically rather large, where their product can run into the millions or even billions.

Our main contribution is a new LDL strategy for reducing the total number of human labels needed per data item, by grouping together items determined to be semantically similar and pooling together their labels of all items in the each semantic class into a single label distribution which can be shared by all members of the class. Figure 1 illustrates this strategy.

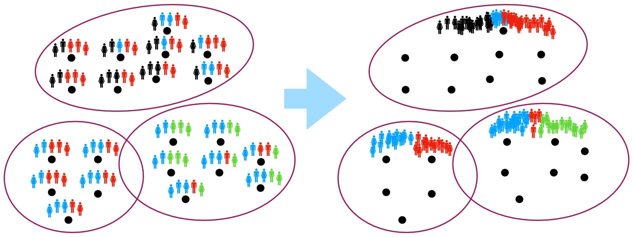

Figure 1: The main strategy this paper explores. The black dots represent data items. (Left:) Five labelers annotate each data item, where the color of the person indicates the label that person chose. If we view these five labels as a sample of the underlying population's beliefs, the sample size is probably too small for there to be much confidence in the sample. (Right:) We cluster together (indicated by the circles) semantic similar data items, and then pool together all the labels in each cluster into a single, larger sample which, according to our strategy, is a good representation of—and thus label distribution for—the population-level beliefs about each item in the cluster.

Specifically, we:

(1) Establish the premise for our proposed approach through a real-world example where there is substantial disagreement over the annotators' interpretations of 50 data items in a common social domain, but where the label distributions appear visually in histogram to cluster into a limited number of distinct classes.
(2) Introduce a novel family of algorithms for label distribution learning on as few as five-to-ten labels per data item that involves an unsupervised learning phase to yield hidden classes of semantically-related data items and assigns to each class an aggregated label distribution, followed by a supervised learning phase based on the labels the unsupervised phase produces.
(3) Show that, for larger label spaces, predictions based on unsupervised learning models that use our clustering strategy outperform those that do not, thus providing supervised learning validation for our approach.
(4) Perform our analysis on natural language data. We believe this is the first exploration of LDL on linguistic data from social media.

2 RELATED WORK

Probability and statistics play such an important role in machine learning that the field is sometimes regarded as a branch of applied statistics. However, in conventional machine learning, probabilistic models are typically used only for *performing* prediction; they are much less seldom the *objects* to be predicted.

Disagreement in human labeling tasks for supervised learning is widely studied as a common problem in its own right [9, 19, 25, 26, 33, 34]. Snow et al. [35], in a study on using multiple crowdsourced annotators to approximate the performance of experts, noted that individuals (including experts) tend to have personal biases (which later research [18] confirmed), but that multiple annotators may contribute to diversity, thus reducing individual annotator bias (see also [7, 11]). However, there is still an underlying assumption that a correct answer exists, even if it can never be directly confirmed.

Recent work has recognized the value of preserving subjectivity and ambiguity in data collection from human annotators. Aroyo and Welty show in a semantic parsing task that crowdworkers, when they agree with each other, can perform at a level comparable to domain experts, and when they disagree it is often for good reason, and in fact usually more desirable than collapsing to a single label [2]. Schaekermann et al. describe a framework for identifying unresolvable annotator disagreement [32].

Chen et al. [8] argue persuasively that to a wide spectrum of social scientists the volume of unstructured data available for qualitative analysis generated by social media is so great that automated methods like machine learning are needed to keep up. They also argue that preserving annotator disagreement is essential to applying qualitative methodologies like grounded theory at scale.

Zhang et al. [38] study the use of clustering to improve estimates of ground truth in crowdsourced labeling tasks, and show that the latent classes determined by their clustering models are, compared to plurality-based labels, better estimates of the semantics of the data items they study, thus providing support for this approach in the context of supervised learning (which they do not study). Clustering is a key subroutine in our learning algorithms, but here we study supervised learning tasks, where the goal is to predict the label distributions of unlabeled data (based on features separate from the labels themselves, in our case language as features), given a training set of human-labeled examples for learning. There are also key differences as to which clustering approaches we use and how we interpret the clusters they return.

Geng pioneered the systematic study of label distribution learning [13], where the objects to be predicted are probabilities of labels/classes. He and colleagues studied applications of LDL in many settings, some of which are related to predicting population-level distributions [14, 16, 31] while others are not [12, 15]. Several of these studies acknowledge the difficulty of obtaining valid label distributions that represent the underlying beliefs of human annotators; in fact, most of them were based on data and labels originally collected for the purpose of conventional (i.e., non-probabilistic labels) supervised learning problems. However, this line of research has thus far assumed that the label distributions obtained are equal to ground truth, i.e., without questioning the statistical validity of the data, even though the sample size of the labels for each item is small.

A number of research areas are related to LDL. In *multilabel learning* [39], each data item is associated with multiple labels. However, it does not typically distinguish between multiplicity due to disagreement (where different annotators might believe that only

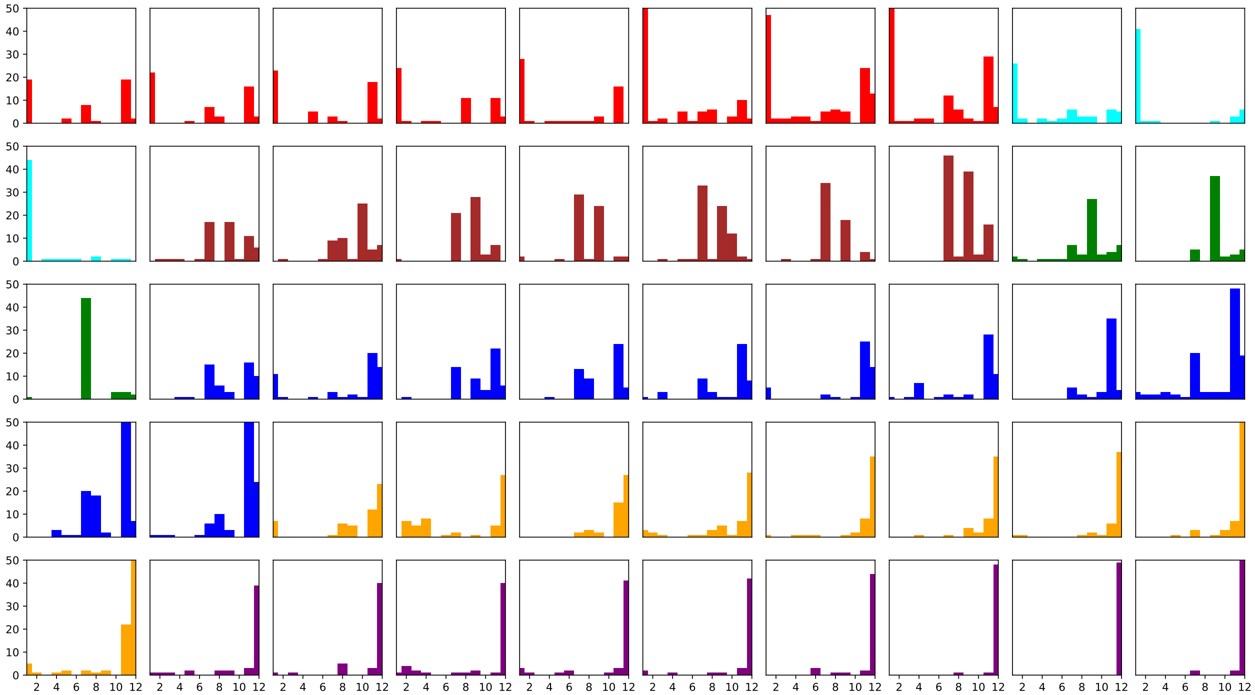

Figure 2: Histograms of the jobQ3MT+ label distributions. The X-axis ranges from 1 to 12, matching the Q3 choices in Figure 4. The Y-axis denotes the label counts. Tweets color: 1-8 red, 9-11 cyan, 12-18 brown, 19-21 green, 22-32 blue, 33-41 orange, and 42-50 purple.

.

one label is correct, but disagree on which one), ambiguity (where an annotator might believe multiple labels are valid), or uncertainty. Such distinctions may have significant social impacts, especially when disagreements fall along crucial demographic boundaries or indicate important but opposing perspectives that should just be preserved in the machine learning predictive models. Moreover, there are settings where label distributions are important but multilabel approaches do not naturally apply, such as when the prediction is ordinal (e.g., Likert-scaled) or real-valued. We are interested in capturing the diversity of beliefs across a population, where each member of the population may only associate a data item with one (set of) label(s), but different people may disagree on which ones.

One of the few areas where large samples of labeler beliefs or opinions are available are on commercial websites such as Amazon or Netflix. Recommender systems [3] seek to use such data to personalize user experiences like shopping, viewing, or playing by matching an individual's past behaviors and habits to similar individuals. Although modeling individual annotators is of interest to us (and a limitation of this paper is that we do not attempt to do so), in many data annotation settings there is, compared to recommender systems, little information available about the annotators to exploit. For instance, datasets labeled via crowdsourcing may not even have plenty of labels per annotator. Also, we are interested in predicting population-level beliefs about data items. Recommender systems, by contrast, predict which items an individual prefers.

There is research in the field of statistics on how to validly combine multiple samples from an underlying population [27]. These papers typically only consider standard statistical settings where a small number of samples are available and each is much larger than ours. We by contrast consider the problem of partitioning a massive number of very small samples into a much smaller number of clusters and then combining together all of the samples in each partition.

3 LABEL DISTRIBUTION LEARNING ON POPULATIONS

The *population label distribution learning problem* is to learn to predict the distribution of labels $\mathbf{y}$ among a population of annotators for each test set data item $\mathbf{x}$, given a collection of training data items $(\mathbf{x}_i)_{i\in\{1,\ldots,n\}}$ and a corresponding collection of label distribution *raw estimates* $(\hat{\mathbf{y}}_i)_{i\in\{1,\ldots,n\}}$, based on the normalized *empirical label distributions*, i.e., the distributions of the annotations received for each data item. Note that, here, we assume these distributions are multinomial samples of the underlying population of annotator's *true label distribution* $(\mathbf{y}_i)_{i\in\{1,\ldots,n\}}$, and that the each raw estimate was obtained by randomly choosing an annotator and then asking that annotator to choose a label, then repeating this process m times, where m is a parameter of the sampling process.

One example of a label set that supports this problem definition came from an effort to model Twitter discourse on life trajectories. When inspecting annotators' answers to a question which identifies employment transition events, we observed that when there was disagreement it was often for good reason.

Figure 2 shows the label distributions over the the jobQ3MT+ label set (see more details in Section 4.1). These histograms of labels (one histogram per data item) appear to cluster into approximately eight categories, where the tweets in each seemed to be semantically related. Group 1 (red) distributions have most of their mass on *Getting hired/job seeking* and *None of the above, but job-related*, with tweets talking about plans to get a job (e.g., *really want a job, dont put that on ur resume for a minimum wage job*) or the process of getting a job. Group 2 (cyan) has almost all the mass exclusively on *Getting hired/job seeking* (e.g., *got the job*). Group 3 (brown) clusters around *Complaining about work* and *Going to work*, suggesting a topic about complaining about having to go to work. Group 4 (green) are a set of tweets complaining about work while at work. Groups 5 and 6 (blue and orange) have their peaks on *None of the above, but job-related* and *Not job-related.* Group 6 (where *Not job-related* was more frequent than *None of the above*) were mostly about road work. Group 7 seemed to contain cases where work was mentioned, but not central (e.g., *Today at work I learned about...*) or used "work" or "job" metaphorically, though there exist some clear *None of the above, but job-related* tweets, like *Perks of working overnight: donuts fresh out of the fryer.*

As to why such clustering happens, Zhang et al., on a different dataset, noticed similar clustering patterns [38]. We note that any k-choice annotation task effectively reduces the full breadth of interpretations encoded in each data item x to one of only k choices; We theorize that the act of annotation reduces not only the interpretive domain of the each data item, but also the social, experiential and cognitive factors, such as disparities in experience and knowledge, that drive annotator disagreement. Thus, the number p of *distinct* ground truth label distributions resulting from any annotation task are also limited, and the set of all annotations for any given data item is (assuming annotators are selected i.i.d. from the population of annotators) a sample from one of the p distinct ground truth distributions. We refer to this tentative explanation here as the *clustering theory.*

3.1 Overview of Learning Stages

Our approach to label distribution learning on populations consists of two stages. First, we use unsupervised learning to convert the raw label distribution estimates $(\hat{\mathbf{y}}_i)_{i\in\{1,\dots,n\}}$, into *refined estimates* $(\hat{\mathbf{y}}'_{\mathbf{i}})_{i\in\{1,\dots,n\}}$, by aggregating over semantic-related data items. Next, we perform supervised learning models on the refined label distributions with unstructured text features and conduct comparative experiments. We discuss each stage below.

3.2 Unsupervised Learning for Pooling Label Distributions

The unsupervised learning algorithms we consider here are consistent, to varying degrees, with the clustering theory. The (finite) multinomial mixture model **F**. This model most directly simulates the sampling process according to the cluster theory. It assumes that the empirical label distributions are generated by, (1) drawing a multinomial distribution π according to a Dirichlet prior over p elements (i.e., corresponding to the hypothesized number of true label distributions) $\pi \sim \mathrm{Dir}(p, \gamma = 75)$, where γ is the prior's (symmetric) hyperparameter (and higher numbers tend to produce lower entropy multinomials); (2) drawing multinomial distributions $\phi_1, \dots \phi_p \sim \mathrm{Dir}(d, \gamma = 0.1)$; (3) for each data item, we (3a) choose $i \sim \pi$ and (3b) m labels according to ϕ_i. Thus, according to the clustering theory the most likely cluster distribution ϕ_j for each data item should be a good estimate of the true label distribution: $\phi_j \approx \mathbf{y}_i$. We use a variational Bayes algorithm[1] to learn the model.

Next come two variants of **F**. The Dirichlet process multinomial mixture model (**DP**) is a non-parametric version of **F**. Instead of choosing p multinomial models from a Dirichlet prior before generating the data, it starts with two multinomial models $\phi_1, \phi_2 \sim \mathrm{Dir}(d, 0.1)$. Then, for each new data item it draws from the current set of multinomial models in approximate proportion to the number of times each has been previously drawn OR draws a new multinomial model (with weight proportional to $\gamma = 50$). We use a variational Bayes algorithm to learn this model. The main purpose for including it here is to test whether in this setting non-parametric methods outperform parametric ones using standard model-selection criteria.

M is a multinomial mixture model without Dirichlet priors. This rather simple model can be learned using EM, however it lacks the regularization and adaptability that the Dirichlet priors provide. We expect this model to underperform the others.

In contrast to the previous models, we chose the Gaussian mixture model **G** as a weak alternate hypothesis of sorts. Rather than simulate the sampling process, as the multinomial distributions do, these distributions capture the variance in a population of samples. Additionally, it captures covariance between the labels; these should be close to zero in single label settings (or settings where the vast majority of annotators provide only one label per item). We use EM[2] to learn this model.

Finally, **L** is latent Dirichlet allocation[3] [4]. Though LDA is not a proper clustering model, we can obtain cluster-like latent classes from it. In terms of **F**, rather than choosing a single class selection distribution π for all data items, it chooses a new one π_i for each item i and for each label chooses a new distribution in $\{\phi_1, \dots, \phi_p\}$ according to π_i. Thus, each instance of the labels for each item i from LDA represent a true mixture of all of the generating distributions, and is therefore not a proper clustering model (in contrast to the other models, where each instance of labels comes from one generating distribution only, although different instances may use different generators). Nonetheless, we can "assign" to i the most likely ϕ_j according to π_i.

3.3 Supervised Learning for Predicting Label Distributions

We train supervised-learning-based classifiers using refined label distributions obtained from the various unsupervised learning algorithms described above. We retain the most common 20,000 words in the test set and pad the sentence with up to 1,000 tokens in the text pre-processing step, then embed each word into a 100-dimension vector using the GloVe 2B-tweet corpus [29].

[1] Adapted from https://github.com/bnpy/bnpy.
[2] http://scikit-learn.org/stable/modules/mixture.html
[3] Based on https://radimrehurek.com/gensim/

We consider two neural network models. One ("**CNN**") is based on a 1D convolutional neural network, designed for sentence or tweet classification [21], with three max pool/convolution layers, followed by a dropout and a softmax layer. The other ("**LSTM**") is an encoder-decoder sequence-to-sequence model using recurrent neural networks . The encoder outputs a fixed-length encoding of the input text, and the decoder predicts the output sequence.

For both types of models, we use *softmax*: $\frac{exp(z_i)}{\sum_t exp(z_t)}$, to transform the output of the penultimate layer $\mathbf{z}$ into a probability distribution. We use *Kullback-Leibler (KL) divergence*, a standard measure of the difference between the "true" (in our case the refined estimate) probability distribution $\hat{\mathbf{y}}'$ and a predicted estimator $\tilde{y}$: $\mathrm{D}_{KL}(\hat{y}', \tilde{y}) = \sum_i P(\hat{y}' = i)\frac{\log P(\hat{y}'=i)}{\log P(\tilde{y}=i)}$, as the loss function for backpropagation, with the *Adam* optimizer [22]. We train with a batch size of 32 and 25 epochs.

4 EXPERIMENTS

Figure 3 summarizes our experiment framework which includes data and label collection, unsupervised and supervised modeling phrases and corresponding performance evaluations.

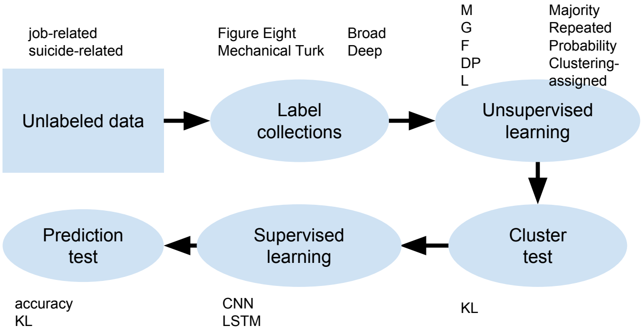

Figure 3: Our experimental workflow involves obtaining crowdsourced labels for raw data (yielding empirical label distributions for each data item), trying various unsupervised strategies for aggregating those labels, and finally testing how each approach affects the efficiency of supervised learning prediction. Note there are two testing phases: one for how well each aggregation strategy fits the data and one for supervised learning performance. We also list key terms, keywords, and abbreviations associated with each phase of the workflow.

4.1 Data and Labels

We consider two corpora: a set of 2,000 job-related tweets (mentioned in Section 3) and another set of 2,000 *suicide-related* tweets. Our institutional review board determined that our work did not fall under federal or institutional guidelines as human subjects type of research. Nonetheless, we took extra precautions to guarantee the privacy of the Twitter data: we replaced all mentions of usernames with "@SOMEONE" and URLs with "http://URL," and adhered to Twitter's developer policy[4]. Table 1 describes the basic properties of labels we collected for these two corpora.

Job-related. We introduced the job dataset in Section 3. It contains 2,000 tweets about work that were extracted by a publicly available library [24]. We asked five crowdworkers each from Figure Eight[5] (FE) and Amazon Mechanical Turk[6] (MT) to answer three questions about each tweet. Figure 4 shows the three questions we asked and their corresponding selections of labels. We denote these label sets *jobQ1/2/3*. To provide some insight into how performance might change with more labels from a more diverse population of labelers and labeling platforms, we first consider FE and MT as two separate label sets, then combine them into a single label set (denoted BOTH).

For each question, we then run experiments on two different train/dev/test splits. We first consider a 1000/500/500 split on each of the label sets: Q1, Q2, and Q3 (which we call the **Broad** split). Next, to get a more accurate ground-truth estimate for testing we randomly selected 50 tweets from our dataset and asked 50 additional MT crowdworkers to label them. We denote these label sets *jobQ1/2/3MT+* and create 1500/450/50 splits (called the **Deep** splits), where the training and development sets are from the BOTH label sets (minus the *jobQ1/Q2/Q3MT+* label set items) and the test sets are from *jobQ1/Q2/Q3MT+*, respectively.

Suicide-related. The *Suicide* tweet label set was obtained directly from [23]. It contains for each data item labels from five Figure Eight crowdworkers and up to two experts in suicide prevention. Each tweet was labeled as one of the following: Ⓐ *Suicidal thoughts*, Ⓑ *Supportive messages or helpful information*, Ⓒ *Reaction to suicide news/movie/music* and Ⓓ *Others*. We use a 1000/500/500 train/dev/test split.

4.2 Unsupervised Learning Experiments

4.2.1 Model Selection. For those clustering models requiring p as a hyperparameter, we test values for $p \in [d/2, 2d]$, where d is the number of label choices. We use the native likelihood function as our model selection criterion, because it is the native optimization goal of each unsupervised clustering algorithm. As the estimators for these models are stochastic and/or sensitive to initial conditions, for every model and every set of hyperparameters we ran 100 trials on the training/dev set and picked the model with the highest estimated likelihood. Table 2 shows the number of clusters selected on each of the two training splits on each label set and for **DP** the number of clusters the algorithm generated.

4.2.2 Evaluation. For the model M produced by each unsupervised learning algorithm and each data item i in the test set, we determine the most likely cluster j for i's empirical label distribution ϕ_j: $\arg\max_j P(\hat{y}_i \sim \phi_j \mid M)$. We then compute the KL divergence between the empirical label distribution $\hat{y}_i$ and the cluster distribution ϕ_j.

Table 3 shows that the multinomial mixture models (**M**/**F**/**DP**) generally outperformed **G**, as we expected. The crowdsourced sample sizes of 5–10 labels we used for each training item are typical

[4] https://developer.twitter.com/en/developer-terms/agreement-and-policy
[5] https://www.figure-eight.com/
[6] https://www.mturk.com/

Label Set	#Items	#Choices /item	#Workers	#Labels	Density	MVTD	RMSD
jobQ1FE	2,000	5	171	10,000	5.00	0.37	0.21
jobQ1MT	2,000	5	1,014	12,202	6.10	0.17	0.10
jobQ1BOTH	2,000	5	1,185	22,202	11.10	0.29	0.16
jobQ1MT+	50	5	249	2,969	59.38	0.43	0.22
jobQ2FE	2,000	5	171	10,000	5.00	0.28	0.16
jobQ2MT	2,000	5	1,014	12,202	6.10	0.15	0.09
jobQ2BOTH	2,000	5	1,185	22,202	11.10	0.23	0.13
jobQ2MT+	50	5	249	2,969	59.38	0.34	0.19
jobQ3FE	2,000	12	171	10,967	5.48	0.45	0.16
jobQ3MT	2,000	12	1,014	12,900	6.45	0.28	0.10
jobQ3BOTH	2,000	12	1,185	23,867	11.93	0.40	0.14
jobQ3MT+	50	12	249	3,196	63.92	0.41	0.14
Suicide	2,000	4	124	13,175	6.59	0.27	0.17

Table 1: Basic properties of our label sets. For the job-related data set with three questions *jobQ1/2/3*, *FE* and *MT* represent the labels from the platforms Figure Eight and Amazon Mechanical Turk respectively. *BOTH* combines both FE and MT labels. Density is the average number of labels per data item. MVTD (majority-voted-true-class deviation) and RMSD (root-mean-square deviation) are two proposed measures for estimating the variety and divergence of different label sets, motivated by the literature on scale and outlier description [20, 30, 37]. MVTD is the average deviation of the majority-voted label over all data items: MVTD = $1 - \sum_{i=1}^{n} \max_j \{\hat{y}_{ij}\}/n$. RMSD is the L2 deviation from the average label distribution: RMSD = $\sum_{i=1}^{n} \sqrt{(\hat{\mathbf{y}}_i - \overline{y})^T (\hat{\mathbf{y}}_i - \overline{y})}/n$, where $\overline{y}$ is the average label distribution over all data.

Q1. Which of the following items could best describe the point of view of job /employment-related information in the target tweet?

○ 1st person ○ 2nd person
○ 3rd person ○ Unclear
○ Not job-related

Q2. Which of the following items could best describe the employment status of the subject in the tweet?

○ Employed ○ Not Employed
○ Not in Labor Force ○ Unclear
○ Not job-related

Q3. Does the subject specifically mention any job/employment transition event in the tweet? (Choose all that apply)

$\square_{1}$ Getting hired/job seeking
$\square_{2}$ Getting Fired
$\square_{3}$ Quitting a job
$\square_{4}$ Losing job some other way
$\square_{5}$ Getting promoted/raised
$\square_{6}$ Getting cut in hours
$\square_{7}$ Complaining about work
$\square_{8}$ Offering support
$\square_{9}$ Going to work
$\square_{10}$ Coming home from work
$\square_{11}$ None of the above, but job-related
$\square_{12}$ Not job-related

Figure 4: The job-related annotation tasks contain these three questions and corresponding choices. The answers for Q3 are the columns in each of the histograms in Figure 2.

	Broad split					Deep split				
Dataset	**M**	**G**	**L**	**F**	**DP**	**M**	**G**	**L**	**F**	**DP**
jobQ1FE	10	4	9	3	4	11	11	9	3	4
jobQ1MT	11	4	11	8	10	2	2	11	9	11
jobQ1BOTH	11	2	2	6	8	2	2	11	7	8
jobQ2FE	11	3	10	3	4	11	2	10	3	4
jobQ2MT	2	4	11	7	9	2	2	11	7	10
jobQ2BOTH	2	2	11	5	7	2	2	8	5	7
jobQ3FE	19	5	18	6	7	19	10	19	7	7
jobQ3MT	5	5	14	17	20	5	19	15	17	26
jobQ3BOTH	5	15	18	13	16	5	17	11	17	17
Suicide	8	2	7	4	5	-	-	-	-	-

Table 2: The optimal label aggregation models on each label set using two splits (*Broad* and *Deep*) are achieved with the presented number of clusters (p).

of crowdsourced supervised learning label sets, and the differences between **G** and the other cluster models appear to be substantial at this scale. The success of **L** on a number of label sets surprised us, considering that we only use the mostly likely cluster for each data item which was trained on a mixture of clusters. Finally, **F** outperforms the other models on all of the sets having at least ten annotations per item, and shows the most improvement from the FE/MT (which had five annotations per item) to the BOTH (with ten annotations per item) label sets.

Table 3 also shows the average and standard deviation of the KL divergence scores on the four independent label sets (i.e., BOTH comprises FE and MT) jobQ1BOTH, jobQ2BOTH, jobQ3BOTH and Suicide (highlighted in gray). These statistics indicate that **F** outperforms the other models across different thematic label sets in

	Broad split					Deep split				
KL	**M**	**G**	**L**	**F**	**DP**	**M**	**G**	**L**	**F**	**DP**
jobQ1FE	0.35	0.53	0.23	0.39	0.39	0.30	0.57	0.24	0.37	0.39
jobQ1MT	0.19	0.68	0.18	0.13	0.15	0.20	0.39	0.07	0.09	0.10
jobQ1BOTH	0.20	0.46	0.40	0.19	0.19	0.21	0.38	0.06	0.06	0.07
jobQ2FE	0.26	0.54	0.19	0.32	0.32	0.24	0.65	0.20	0.28	0.28
jobQ2MT	0.36	0.74	0.15	0.10	0.10	0.26	0.50	0.09	0.11	0.13
jobQ2BOTH	0.28	0.51	0.17	0.16	0.16	0.25	0.48	0.09	0.08	0.08
jobQ3FE	0.51	1.00	0.52	0.59	0.64	0.29	0.97	0.27	0.41	0.41
jobQ3MT	0.50	1.15	0.33	0.26	0.29	0.20	0.51	0.17	0.28	0.21
jobQ3BOTH	0.45	0.82	0.35	0.32	0.33	0.18	0.64	0.18	0.12	0.13
Suicide	0.22	0.57	0.20	0.22	0.22	-	-	-	-	-
Average	0.29	0.59	0.28	0.22	0.23	0.21	0.50	0.11	0.09	0.09
Std dev	0.10	0.14	0.10	0.06	0.06	0.03	0.11	0.05	0.02	0.03

Table 3: KL divergence based on the chosen label clustering models in Table 2. Average and standard deviation are based on the KL divergence scores of the gray-highlighted rows (jobQ1BOTH, jobQ2BOTH, jobQ3BOTH and Suicide). The *lowest* KL is highlighted in yellow for each split.

its capability and stability, **DP** is second, and, as we expected, **G** comes last.

Readers may have notice in Figure 4 that Q3 differs Q1 and Q2 in allowing annotators to choose multiple labels. Theoretically, the ideal representation for the annotations (where each *annotation* is the set of labels provided by one annotator for one data item) of Q3 would be over the *power set* of possible labels. However, Table 4 shows that fewer than 10% of the annotations we received had selected more than one label. Thus, we simplify matters by representing these distributions as over just the base label set (i.e., *not* the power set), and treat multiple labels from the same annotator as if each came from its own, independent annotator (for example, an annotation with three labels provided is treated as three separate annotations.).

	Number of workers with labels they submitted				
Label Set	**1**	**2**	**3**	**4**	**5+**
jobQ3FE	10,000	722	176	53	16
jobQ3MT	12,202	628	58	11	1
jobQ3MT+	2,969	193	32	2	0

Table 4: Stats for workers annotating jobQ3 with numbers of labels they submitted individually.

4.3 Supervised Learning Experiments

We then trained the two supervised learning algorithms described in Section 3.3 on our training datasets' texts, using in turn each of the unsupervised learning methods described previously to provide *refined label distribution estimates* $(\hat{\mathbf{y}}'_i)$ as the learning goal. We compared their performances to those of three common baseline strategies for resolving (or not) label disagreement.

- Majority (**Maj**) takes the final label to be $\hat{y}'_i = \underset{j\in\{1,\ldots,d\}}{\arg\max}\ \{\hat{y}_{ij}\}$.
- Repeated (**Rept**) duplicates each data instance once for every annotation it receives and pairs the replicated instance with that label.
- Probability (**Prob**) is the raw label distribution estimates $(\hat{\mathbf{y}}'_i) = (\hat{\mathbf{y}}_i)$. (This is the baseline LDL approach.)

4.3.1 Evaluation. We measure the **KL divergence** between the classifier $(\tilde{\mathbf{y}}_i)$ and cluster-or-baseline-method $(\hat{\mathbf{y}}'_i)$ -based label distributions. (Note that Maj and Rept both associate each data item, by eliminating labels or creating copies of the data items, exactly one label. For the purpose of computing KL divergence we regard this as a distribution where the entire probability mass is on one label.) We also measure accuracy (**ACC**), i.e., the percentage of times $\arg\max_j \tilde{y}_{ij}$ matches $\arg\max_j \hat{y}'_{ij}$ in the test set. Accuracy is often used in nondistributional classification problems. We use it here to shed further light into the differences between distributional and nondistributional problems. In particular, we might expect that nondistributional models might outperform label distribution models with respect to accuracy, even as they underperform with respect to KL divergence.

4.3.2 Results. Tables 5-8 show the KL divergence and accuracy metrics for CNN/LSTM text classifiers built with different label aggregation strategies in two split modes (Broad split: Table 5 and 7, Deep split: Table 6 and 8) .

Starting with the KL divergence results, on the Broad split tests, CNNs trained and tested on **L** outperform other clustering and non-clustering approaches most of the time for both job and suicide discourse themes. For LSTMs, we can also observe that clustering approaches achieved better results more often on different label sets than non-clustering methods. Almost none of CNNs or LSTMs trained on any baseline label reduction strategy can compete.

By contrast, the results of the Deep split KL divergence tests (Table 6) are not as conclusive, and this could be due to there being fewer data items in the Deep split test set. But even so, clustering strategies again perform better in more cases than the baselines.

Tables 7 and 8 show that, for both the CNN and LSTM classifiers and both split modes, the highest accuracies often come from the clustering methods. They outperform non-clustering methods by more than 10% on average, which appears substantial. For those label sets whose accuracy based on clustering strategies do not rank 1st, non-clustering methods win only by a slim or zero margins.

Together, the results for different label sets and split modes reveal several interesting patterns. First, the cluster-based models tend to outperform the baseline methods in terms of either KL divergence or accuracy as reported. This supports the feasibility of our clustering strategy for label distribution learning on subjective problems with annotator disagreement. On the other hand, for conventional (i.e., non-distributional) classification problems, baseline methods can be sufficient, as shown in our experiment results. The advantages of clustering, in terms of KL divergence, is less stark in the Deep compared to the Broad splits, but clustering still seems to outperform baselines on the jobQ3 label set, which has the largest label space and is where pooling and other label conservation methods are most needed.

Broad - KL	CNN							
	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	2.98	0.79	0.91	0.12	0.74	0.47	0.18	0.19
jobQ1MT	2.03	0.80	0.72	0.65	1.05	0.52	1.02	1.00
jobQ1BOTH	2.38	0.45	0.48	0.36	0.38	0.27	0.40	0.38
jobQ2FE	2.29	0.91	0.79	0.21	0.78	0.13	0.31	0.28
jobQ2MT	2.10	0.80	0.78	0.81	0.98	0.67	1.04	0.96
jobQ2BOTH	2.12	0.49	0.47	0.48	0.48	0.37	0.51	0.52
jobQ3FE	4.20	1.66	1.14	0.31	0.68	0.66	0.42	0.36
jobQ3MT	3.18	2.24	1.05	1.04	1.32	0.54	1.12	1.12
jobQ3BOTH	3.38	1.40	0.77	0.62	0.49	0.62	0.71	0.70
Suicide	2.16	1.40	0.45	0.69	13.62	0.33	0.53	0.49
Average	2.51	0.94	0.54	0.54	3.74	0.40	0.54	0.52
Std dev	0.51	0.47	0.13	0.13	5.70	0.13	0.11	0.11

Broad - KL	LSTM							
	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	1.81	0.98	1.34	0.60	1.22	1.02	0.54	0.75
jobQ1MT	1.42	1.17	1.96	1.55	1.67	2.08	1.54	2.07
jobQ1BOTH	1.23	1.10	1.32	1.19	0.69	1.50	1.06	1.03
jobQ2FE	1.41	1.86	1.69	1.57	1.49	1.03	1.44	1.02
jobQ2MT	2.04	1.96	2.20	1.30	1.53	2.39	2.37	1.62
jobQ2BOTH	1.52	1.65	1.54	1.06	1.28	1.33	1.75	1.93
jobQ3FE	1.65	1.92	1.59	0.98	0.99	1.09	1.05	1.09
jobQ3MT	1.80	2.29	2.07	1.82	1.80	1.46	2.01	1.72
jobQ3BOTH	1.67	1.71	1.68	1.30	1.11	1.53	1.36	1.16
Suicide	1.50	1.27	1.34	1.32	16.40	1.28	0.90	1.07
Average	1.48	1.43	1.47	1.22	4.87	1.41	1.27	1.30
Std dev	0.16	0.26	0.15	0.10	6.66	0.11	0.32	0.37

Table 5: Kullback–Leibler divergence of the Broad split. Average and standard deviation are based on the gray-highlighted rows (jobQ1BOTH, jobQ2BOTH, jobQ3BOTH and Suicide). The *lowest* KL is highlighted in yellow.

Deep - KL	CNN							
	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	3.09	0.77	0.90	0.13	0.69	0.39	0.09	0.16
jobQ1MT	2.94	0.47	0.54	0.64	1.08	0.47	1.22	1.05
jobQ1BOTH	2.90	0.34	0.24	0.39	0.43	0.38	0.33	0.35
jobQ2FE	3.07	0.57	0.65	0.18	0.56	0.49	0.21	0.31
jobQ2MT	1.90	0.50	0.58	0.77	0.68	0.76	0.74	1.07
jobQ2BOTH	2.90	0.27	0.28	0.52	0.37	0.35	0.50	0.58
jobQ3FE	3.71	1.45	1.00	0.34	0.63	0.65	0.53	0.43
jobQ3MT	3.95	1.98	0.77	1.13	1.21	1.20	1.26	1.24
jobQ3BOTH	3.33	1.13	0.63	0.76	0.67	0.49	0.71	0.73
Average	3.04	0.58	0.38	0.56	0.49	0.41	0.51	0.55
Std dev	0.20	0.39	0.18	0.15	0.13	0.06	0.16	0.16

Deep - KL	LSTM							
	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	1.16	1.06	1.01	0.92	0.95	0.80	1.16	1.17
jobQ1MT	0.93	0.85	1.09	1.38	2.04	1.54	1.61	2.52
jobQ1BOTH	1.30	0.91	1.20	1.20	1.49	1.13	0.82	1.09
jobQ2FE	1.59	1.11	1.58	1.23	1.38	0.96	1.17	1.09
jobQ2MT	1.16	1.00	0.76	1.69	1.50	1.51	1.42	2.47
jobQ2BOTH	1.02	1.10	1.17	0.97	1.19	1.63	1.69	1.26
jobQ3FE	1.86	1.90	1.67	0.76	1.02	1.73	0.93	1.20
jobQ3MT	1.38	1.39	1.69	2.22	1.65	2.25	2.13	1.59
jobQ3BOTH	1.64	1.49	1.46	1.69	0.99	1.94	1.37	1.44
Average	1.32	1.17	1.28	1.29	1.22	1.57	1.29	1.26
Std dev	0.25	0.24	0.13	0.30	0.21	0.33	0.36	0.14

Table 6: Kullback–Leibler divergence of the Deep split. Average and standard deviation are based on the gray-highlighted rows (jobQ1BOTH, jobQ2BOTH, and jobQ3BOTH). The *lowest* KL is highlighted in yellow.

5 DISCUSSION

Our results provide evidence that clustering is a feasible strategy to improve performance of label distribution learning in certain settings, such as when each label distribution represents a population estimate based on a (micro) sample, and the data falls into a small number of semantic equivalence classes (relative to the learning task). Yet, why this is so is still not clear; our results shed little light on the validity of the clustering theory.

They also raise methodological issues. We expect that the methods introduced here for testing performance will provide helpful baselines for the development of newer methods tailored specifically toward settings where ground truth depends on a small number of samples per data item.

One methodological issue we grappled with was whether to measure the performances of the supervised models against the empirical label ($\hat{y}$) or the refined label ($\hat{y}'$) distributions. We felt that it was standard practice to test supervised learning on the patterns they are fed (i.e., the refined labels in our case), even though in our case the conventional machine learning algorithms are only the last half of a larger, novel supervised pipeline that happens to have an unsupervised front end, and which takes the empirical labels as input. We tried both approaches, but here, for space purposes and because we found our earlier results more interesting in this direction, we report on (and examined in much greater depth) only the predictions against $\hat{y}'$. The biggest worry in doing so is that, because pooling labels via a small number of clusters greatly reduces diversity in the label distributions, there is less likelihood of error, which would seem to make predictions easier when measure against $\hat{y}'$ than against the empirical distributions $\hat{y}$. (Certainly, if by chance there was only one cluster, the prediction task would be trivial.) While that may be so, in label distribution learning settings the more important question is, "What is the expected *degree* of error?" For this question, the relationship between label distribution diversity and performance is less clear.

Moreover, even when there is less diversity in the label distribution space, if the relationship between input data features and refined label distributions is less consistent than with the empirical label distributions (recall that the cluster algorithms were trained on the empirical label distribution, but the supervised models were trained on the natural language input data (using the output of the labels from clustering phase as the label distributions to learn), then the predictions could be less accurate.

We have been deliberately vague about what "population of labelers" means. This study was motivated by our work with microtask

	CNN							
Broad - ACC	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	0.73	0.53	0.72	0.78	0.95	0.58	0.64	0.58
jobQ1MT	0.80	0.72	0.79	0.56	0.67	0.76	0.54	0.56
jobQ1BOTH	0.82	0.64	0.81	0.57	0.76	0.76	0.65	0.64
jobQ2FE	0.73	0.63	0.79	0.71	0.62	0.94	0.59	0.64
jobQ2MT	0.73	0.68	0.73	0.48	0.55	0.71	0.53	0.52
jobQ2BOTH	0.76	0.65	0.76	0.63	0.58	0.71	0.54	0.56
jobQ3FE	0.36	0.31	0.41	0.47	0.32	0.45	0.42	0.46
jobQ3MT	0.53	0.45	0.51	0.26	0.28	0.49	0.28	0.28
jobQ3BOTH	0.48	0.42	0.53	0.31	0.62	0.46	0.25	0.21
Suicide	0.81	0.65	0.78	0.18	1.00	0.76	0.37	0.39
Average	0.72	0.59	0.72	0.42	0.74	0.67	0.45	0.45
Std dev	0.14	0.10	0.11	0.18	0.16	0.12	0.15	0.17
	LSTM							
Broad - ACC	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	0.85	0.74	0.86	0.89	0.98	0.76	0.83	0.79
jobQ1MT	0.87	0.83	0.86	0.79	0.80	0.86	0.78	0.77
jobQ1BOTH	0.87	0.79	0.88	0.79	0.88	0.84	0.79	0.82
jobQ2FE	0.87	0.80	0.88	0.79	0.79	0.96	0.83	0.80
jobQ2MT	0.84	0.82	0.85	0.76	0.77	0.85	0.76	0.75
jobQ2BOTH	0.84	0.81	0.86	0.81	0.80	0.83	0.76	0.76
jobQ3FE	0.67	0.65	0.67	0.72	0.67	0.68	0.67	0.71
jobQ3MT	0.69	0.69	0.71	0.63	0.65	0.64	0.65	0.63
jobQ3BOTH	0.70	0.68	0.67	0.63	0.77	0.70	0.64	0.63
Suicide	0.86	0.81	0.87	0.64	0.57	0.85	0.67	0.68
Average	0.82	0.77	0.82	0.72	0.76	0.81	0.72	0.72
Std dev	0.07	0.05	0.09	0.08	0.11	0.06	0.06	0.07

Table 7: Accuracy of the Broad split. Average and standard deviation are based on the gray-highlighted rows (jobQ1BOTH, jobQ2BOTH, jobQ3BOTH and Suicide). The *highest* accuracy for each dataset is highlighted in yellow.

	CNN							
Deep - ACC	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	0.62	0.47	0.58	0.78	0.80	0.54	0.82	0.72
jobQ1MT	0.72	0.53	0.70	0.58	0.66	0.72	0.56	0.58
jobQ1BOTH	0.72	0.51	0.70	0.62	0.90	0.60	0.62	0.60
jobQ2FE	0.60	0.53	0.52	0.76	0.82	0.48	0.50	0.60
jobQ2MT	0.72	0.57	0.70	0.58	0.64	0.66	0.64	0.44
jobQ2BOTH	0.72	0.54	0.76	0.54	0.56	0.68	0.64	0.54
jobQ3FE	0.46	0.40	0.48	0.16	0.30	0.50	0.14	0.40
jobQ3MT	0.54	0.43	0.54	0.14	0.24	0.48	0.30	0.18
jobQ3BOTH	0.62	0.46	0.48	0.20	0.40	0.56	0.16	0.24
Average	0.69	0.50	0.65	0.45	0.62	0.61	0.47	0.46
Std dev	0.05	0.03	0.12	0.18	0.21	0.05	0.22	0.16
	LSTM							
Deep - ACC	Maj	Rept	Prob	M	G	L	F	DP
jobQ1FE	0.77	0.74	0.76	0.87	0.89	0.68	0.84	0.81
jobQ1MT	0.84	0.73	0.85	0.80	0.78	0.83	0.81	0.83
jobQ1BOTH	0.82	0.73	0.79	0.82	0.95	0.76	0.85	0.84
jobQ2FE	0.77	0.77	0.72	0.86	0.87	0.74	0.75	0.80
jobQ2MT	0.80	0.74	0.78	0.75	0.76	0.81	0.83	0.77
jobQ2BOTH	0.78	0.74	0.78	0.79	0.78	0.81	0.84	0.84
jobQ3FE	0.65	0.71	0.67	0.62	0.68	0.71	0.59	0.63
jobQ3MT	0.76	0.67	0.73	0.65	0.63	0.72	0.68	0.69
jobQ3BOTH	0.75	0.70	0.72	0.63	0.75	0.75	0.62	0.67
Average	0.78	0.72	0.76	0.75	0.83	0.77	0.77	0.78
Std dev	0.03	0.02	0.03	0.08	0.09	0.03	0.11	0.08

Table 8: Accuracy of the Deep split. Average and standard deviation are based on the gray-highlighted rows (jobQ1BOTH, jobQ2BOTH, and jobQ3BOTH). The *highest* accuracy is highlighted in yellow.

crowdsourcing sites like AMT and Figure Eight, in which case our labels can be taken as collection of (micro) samples of the population of workers on whichever sites are used for whatever interval of time the requested labeling task is posted. Studies exist on the demographics of these sites. Some sites (like Figure Eight in our study) provide some demographic information on the responders to each microtask request.

We have not yet modeled user behavior, though this is a well-established approach for aggregating labels from multiple annotators. We did, in fact, run experiments using Dawid and Skene's class annotator-based model [10], which is largely based on using behavior. However, as it is designed for conventional, non-distributional supervised learning and did not perform well, we did not report those results here. Another complication is that most of our annotators labeled only ten data items each, so we would be tempted to used clustering to group users in much the same way we used it here to group data items.

Another limitation was that we did not investigate in-depth the causes of inter-annotator disagreement, such as data encoding errors and communication ambiguities [1, 6, 40], lack of sufficient information [5, 6, 17], and unreliable annotators and their bias [17], nor did we attempt to resolve disagreement through follow-up discussions with the annotators, as is common in many grounded theory studies. We suspect in our experiment label sets, there exists some statistical correlations between the subjectivity and ambiguity and the degree of inter-rater disagreement across different questions. We hope to explore these directions in the future.

Another potential future direction could be to explore more highly structured label spaces, such as ordinal ones, or ones based on Bernoulli distributions or "single-peaked-ness" that are common in practice and sometimes yield to high-performance algorithms.

6 CONCLUSION

We study the important problem of predicting the distributions of population beliefs using both unsupervised and supervised learning methods. We test different strategies for clustering data items to obtain aggregated label distributions. We then build supervised CNN/LSTM classifiers using the predicted distributions and compared the performance with common baseline label reduction strategies. Our results from both unsupervised and supervised experiments show that it is feasible to predict probability distributions over labels at the population level. Clustering labels, in general, boosts the label distribution learning by aggregating data items with similar semantics and population-beliefs. We believe our study is an pioneering exploration of disagreement on linguistic data from social media and further helps future intelligent agents understand the diversity of beliefs in society.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their insightful feedback and comments. We thank Cyril Weerasooriya, Lingjia Deng, Chen Chen for helpful conversations.

REFERENCES

[1] Dana Angluin and Philip Laird. 1988. Learning from noisy examples. *Machine Learning* 2, 4 (1988), 343–370.

[2] Lora Aroyo and Chris Welty. 2014. The three sides of CrowdTruth. *Journal of Human Computation* 1 (2014), 31–34.

[3] Joeran Beel, Corinna Breitinger, Stefan Langer, Andreas Lommatzsch, and Bela Gipp. 2016. Towards reproducibility in recommender-systems research. *User modeling and user-adapted interaction* 26, 1 (2016), 69–101.

[4] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[5] Pavel Brazdil and Peter Clark. 1990. Learning from imperfect data. In *Machine Learning, Meta-Reasoning and Logics*. Springer, 207–232.

[6] Carla E Brodley and Mark A Friedl. 1999. Identifying mislabeled training data. *Journal of artificial intelligence research* 11 (1999), 131–167.

[7] Chris Callison-Burch. 2009. Fast, Cheap, and Creative: Evaluating Translation Quality Using Amazon's Mechanical Turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1 (EMNLP '09)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 286–295. http://dl.acm.org/citation.cfm?id=1699510.1699548

[8] Nan-Chen Chen, Margaret Drouhard, Rafal Kocielnik, Jina Suh, and Cecilia R. Aragon. 2018. Using Machine Learning to Support Qualitative Coding in Social Science: Shifting The Focus to Ambiguity. *ACM Transactions on Interactive Intelligent Systems* 8, 2 (June 2018).

[9] Alexander Philip Dawid and Allan M Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied statistics* (1979), 20–28.

[10] Alexander Philip Dawid and Allan M Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied statistics* (1979), 20–28.

[11] Michael Denkowski and Alon Lavie. 2010. Exploring Normalization Techniques for Human Judgments of Machine Translation Adequacy Collected Using Amazon Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (CSLDAMT '10)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 57–61. http://dl.acm.org/citation.cfm?id=1866696.1866705

[12] Bin-Bin Gao, Chao Xing, Chen-Wei Xie, Jianxin Wu, and Xin Geng. 2017. Deep label distribution learning with label ambiguity. *IEEE Transactions on Image Processing* 26, 6 (2017), 2825–2838.

[13] Xin Geng. 2016. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering* 28, 7 (2016), 1734–1748.

[14] Xin Geng and Peng Hou. 2015. Pre-release Prediction of Crowd Opinion on Movies by Label Distribution Learning.. In *IJCAI*. 3511–3517.

[15] Xin Geng and Miaogen Ling. 2017. Soft Video Parsing by Label Distribution Learning.. In *AAAI*. 1331–1337.

[16] Xin Geng, Qin Wang, and Yu Xia. 2014. Facial age estimation by adaptive label distribution learning. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 4465–4470.

[17] Ray J Hickey. 1996. Noise modelling and evaluating learning from examples. *Artificial Intelligence* 82, 1 (1996), 157–179.

[18] Christopher Homan, Ravdeep Johar, Tong Liu, Megan Lytle, Vincent Silenzio, and Cecilia Ovesdotter Alm. 2014. Toward Macro-Insights for Suicide Prevention: Analyzing Fine-Grained Distress at Scale. In *Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*. Association for Computational Linguistics, Baltimore, Maryland, USA, 107–117. http://www.aclweb.org/anthology/W14-3213

[19] Nicholas P Hughes, Stephen J Roberts, and Lionel Tarassenko. 2004. Semi-supervised learning of probabilistic models for ECG segmentation. In *Engineering in Medicine and Biology Society, 2004. IEMBS'04. 26th Annual International Conference of the IEEE*, Vol. 1. IEEE, 434–437.

[20] Rob J Hyndman and Anne B Koehler. 2006. Another look at measures of forecast accuracy. *International journal of forecasting* 22, 4 (2006), 679–688.

[21] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[22] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[23] Tong Liu, Qijin Cheng, Christopher Homan, and Vincent Silenzio. 2017. Learning from Various Labeling Strategies for Suicide-Related Messages on Social Media: An Experimental Study.. In *The workshop on Mining Online Health Reports of the 10th ACM Conference on Web Search and Data Mining*. Cambridge, UK. https://arxiv.org/pdf/1701.08796.pdf

[24] Tong Liu, Christopher M Homan, Cecilia Ovesdotter Alm, Ann Marie White, Megan C Lytle, and Henry A Kautz. 2016. Understanding Discourse on Work and Job-Related Well-Being in Public Social Media. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Berlin, Germany, 1044–1053. https://www.aclweb.org/anthology/P/P16/P16-1099.pdf

[25] Andrea Malossini, Enrico Blanzieri, and Raymond T Ng. 2006. Detecting potential labeling errors in microarrays by data perturbation. *Bioinformatics* 22, 17 (2006), 2114–2121.

[26] Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics* 19, 2 (1993), 313–330.

[27] Engineering, National Academies of Sciences, Medicine, et al. 2017. *Federal Statistics, Multiple Data Sources, and Privacy Protection: Next Steps*. National Academies Press.

[28] Jordan Pearson. 2016 (accessed November 11, 2018). *Why An AI-Judged Beauty Contest Picked Nearly All White Winners*. https://motherboard.vice.com/en_us/article/78k7de/why-an-ai-judged-beauty-contest-picked-nearly-all-white-winners

[29] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543. http://www.aclweb.org/anthology/D14-1162

[30] Robert Gilmore Pontius, Olufunmilayo Thontteh, and Hao Chen. 2008. Components of information for multiple resolution comparison between maps that share a real variable. *Environmental and Ecological Statistics* 15, 2 (2008), 111–142.

[31] Yi Ren and Xin Geng. 2017. Sense beauty by label distribution learning. In *Proc, IJCAI*. 2648–2654.

[32] Mike Schaekermann, Edith Law, Alex C Williams, and William Callaghan. 2016. Resolvable vs. irresolvable ambiguity: A new hybrid framework for dealing with uncertain ground truth. In *Proceedings of the 1st Workshop on Human-Centered Machine Learning at SIGCHI*.

[33] Padhraic Smyth. 1996. Bounds on the mean classification error rate of multiple experts. *Pattern Recognition Letters* 17, 12 (1996), 1253–1257.

[34] Padhraic Smyth, Usama Fayyad, Michael Burl, Pietro Perona, and Pierre Baldi. 1995. Inferring ground truth from subjective labelling of venus images. *Advances in neural information processing systems* 7 (1995), 1085–1092.

[35] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, 254–263.

[36] Staff and agencies. 2016 (accessed November 11, 2018). *Microsoft 'deeply sorry' for racist and sexist tweets by AI chatbot*. https://www.theguardian.com/technology/2016/mar/26/microsoft-deeply-sorry-for-offensive-tweets-by-ai-chatbot

[37] Cort J Willmott and Kenji Matsuura. 2006. On the use of dimensioned measures of error to evaluate the performance of spatial interpolators. *International Journal of Geographical Information Science* 20, 1 (2006), 89–102.

[38] Jing Zhang, Victor S Sheng, Jian Wu, and Xindong Wu. 2016. Multi-Class Ground Truth Inference in Crowdsourcing with Clustering. *IEEE Trans. Knowl. Data Eng.* 28, 4 (2016), 1080–1085.

[39] Min-Ling Zhang and Zhi-Hua Zhou. 2014. A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering* 26, 8 (2014), 1819–1837.

[40] Xingquan Zhu and Xindong Wu. 2004. Class noise vs. attribute noise: A quantitative study. *Artificial intelligence review* 22, 3 (2004), 177–210.