Aggregating Crowdsourced Labels in Subjective Domains

Supervised learning problems—particularly those involving social data—are often subjective. That is, human readers, looking at the same data, might come to legitimate but completely different conclusions based on their personal experiences. Yet in machine learning settings feedback from multiple human annotators is often reduced to a single "ground truth" label, thus hiding the true, potentially rich and diverse interpretations of the data found across the social spectrum. We explore the rewards and challenges of discovering and learning representative distributions of the labeling opinions of a large human population. A major, critical cost to this approach is the number of humans needed to provide enough labels not only to obtain representative samples, but also to train a machine to predict representative distributions on unlabeled data. We propose aggregating label distributions over, not just individuals, but also data items, in order to maximize the costs of humans in the loop. We test different aggregation approaches on state-of-the-art deep learning models. Our results suggest that careful label aggregation methods can greatly reduce the number of samples needed to obtain representative distributions.

CCS Concepts: • **Human-centered computing** → *Computer supported cooperative work*; • **Computing methodologies** → *Supervised learning*;

Additional Key Words and Phrases: Subjective domains, machine learning, humans in the loop, crowdsourcing

ACM Reference Format:
. 2018. Aggregating Crowdsourced Labels in Subjective Domains. *Proc. ACM Hum.-Comput. Interact.* X, X, Article X (April 2018), 17 pages. https://doi.org/0000001.0000001

1 INTRODUCTION

This paper explores the problem of label aggregation in domains that are highly *subjective*, i.e., where different annotators may disagree for perfectly legitimate reasons. Such settings are common, if underacknowledged. Though increasingly, mass media provides stories about the unintended consequences of ignoring this diversity in machine learning.

For example, Beauty.ai sponsored a worldwide beauty contest, judged by a machine learning algorithm. Though light-skinned entrants made up the majority of entrants, they nonetheless won a disproportionate number of contests.[1] Tay, a Twitter-based learning agent, developed by Microsoft, was taught to tweet that the Holocaust was made up[2] (though the Holocaust factually existed, the same cybersocial dynamics of training bias found in subjective domains led to this outcome). ProPublica discovered that Northpointe risk assessment software—used to help judges determine sentence length for convicts—recommended longer sentences for African-American men than other groups, even when controlled for confounding factors.[3]

[1] https://motherboard.vice.com/en_us/article/78k7de/why-an-ai-judged-beauty-contest-picked-nearly-all-white-winners

[2] https://www.theguardian.com/technology/2017/jan/27/ai-artificial-intelligence-watchdog-needed-to-prevent-discriminatory-automated-decisions

[3] https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Author's address:

2573-0142/2018/4-ARTX $15.00
https://doi.org/0000001.0000001

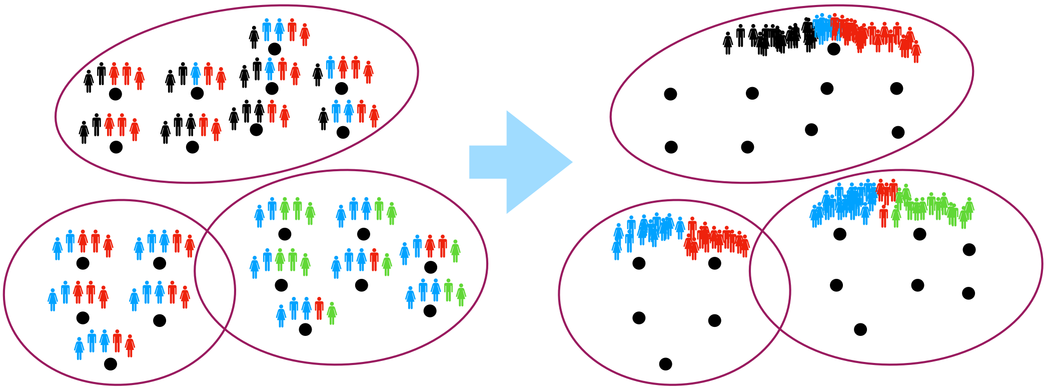

Fig. 1. In this example, data items (black dots) are labeled by five human annotators each (left), where color indicates label choice, yielding an *empirical label distribution* $\mathbf{y}_i$ for each data item i. By clustering similarly labeled objects, we pool together (right) the labels of all data items assigned to the same cluster k into a single, much larger sample θ_k for all items in the cluster. Our research suggests that, in some cases, this larger sample (or a mixture of cluster samples) is a better representation of the true population distribution of beliefs about each data item in the cluster and can lead to better predictive supervised learning.

Learning a distribution of beliefs about a data item, rather than a single "ground truth" label, poses unique challenges. It increases the dimensionality of the learning problem so that more data items may be needed. It also may require more labels per item to get a representative sample of the human populations' beliefs. And for most problems, labels are relatively expensive to obtain. Though crowdsourcing platforms have made this task convenient, they are frequently a resource bottleneck in supervised learning loops.

Our main contribution is a method for minimizing the number of labels needed to learn to predict socially representative label distributions. It is based on the hypothesis that *the sources are subjectivity are limited, and so the number of distinct distributions of beliefs over all data items is likewise limited.* In other words, the label distributions are samples from a relatively small number of true, but hidden, distributions. See Figure 1. These hidden distributions can be seen as latent classes representing population-level beliefs about the labels. According to this hypothesis, we can use unsupervised clustering algorithms to pool together the labels of data items with similar distributions into higher resolution distributions of beliefs shared commonly among all data items in the same cluster.

In particular, we: (1) explore subjectivity as the problem of learning representative distributions from a target population of responses to target questions, (2) propose clustering as a sensible means for pooling together labels from similar data items, to reduce the number of labels needed (3) test what we call our *clustering hypothesis*, that the label distributions of subjective data are clustered around a small number of underlying, true distributions (4) study how different label aggregation strategies and representations affect the performance of state-of-the art deep learning predictors.

It would seem that bias is an inherent part of any information reduction process, such as those found in statistical learning [28]. So it seems naive to expect that machines can learn unbiased models through unsupervised learning alone, or even for any supervised learning that assumes a singular, correct answer to most problems. We hope that this research sparks a broader debate about the best practices for machine learning with humans in the loop.

The rest of this paper is organized as follows. Section 2 describes our experimental workflow, Section 3 presents our results, Section 4 discusses our study, Section 5 presents related work, and Section 6 is the conclusion.

2 METHODS

Figure 2 describes the basic experimental workflow in this study. We discuss each phase below. Note that there are two testing phases, one for determining how well each aggregation method fits the data and another for how well supervised learning algorithms trained by each aggregation strategy perform. Since these test phases share some methods, we discuss them together at the end of the section.

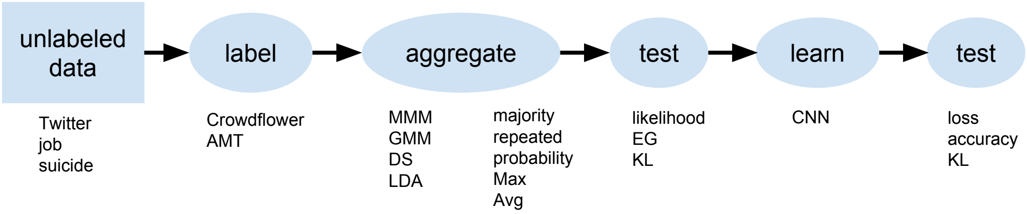

Fig. 2. The basic experimental workflow involves obtaining crowdsourced labels for raw data (yielding empirical label distributions for each data item), trying various strategies for aggregating and pooling those labels (including *no aggregation*), and finally testing how each method affects the accuracy of machine learning prediction. Note there are two testing phases: one for how well each aggregation strategy fits the data and one for machine learning performance. We also list important terms, keywords, and abbreviations associated with each phase of the workflow.

2.1 Data and labeling

We performed extensive experiments, through the aggregation phase, on a number of publicly available, human-labeled datasets [1, 2, 22, 29, 37], including the data described below. However, due to space and time constraints—and because our preliminary studies suggested that these sets covered most of the features of the other sets and exhibited representative performance, we decided to report and focus this study on two of them. Table 1 summarizes the basic properties of these sets, which we now describe in detail.

Before conducting this research, we consulted with our institutional review board, who determined that it did not fall under federal or institutional guidelines as human subjects research. Nonetheless, we took extra precautions to ensure the privacy of all human-generated data. For the twitter data, we replaced all mentions with “@SOMEONE” and URLs with “URL,” paraphrased all examples, and adhered to Twitter’s developer policy. [4]

2.1.1 Job-themed data. We obtained directly from Liu et al. [23] a corpus of machine-filtered *job-related* tweets (i.e., Twitter posts). From this corpus, we randomly selected 2,000 tweets to acquire human annotations from two popular crowdsourcing platforms—Amazon Mechanical Turk[5] (abbreviated as *MT*) and CrowdFlower[6] (*CF*). For each tweet and each platform, we asked five crowdworkers to answer three questions (see Figure 3). We provided contextual information in the form of the three tweets proceeding and succeeding the target tweet made by the target user.

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

ID	Dataset	#Items	#Choices/ item	#Workers	#Labels	Density	MVTD	RMSD
1	jobQ1CF	2000	5	171	10000	5.00	0.37	0.21
2	jobQ1MT	2000	5	1014	12202	6.10	0.17	0.10
3	jobQ1BOTH	2000	5	1185	22202	11.10	0.29	0.16
4	jobQ1MTdeep	50	5	249	2969	59.38	0.43	0.22
5	jobQ2CF	2000	5	171	10000	5.00	0.28	0.16
6	jobQ2MT	2000	5	1014	12202	6.10	0.15	0.09
7	jobQ2BOTH	2000	5	1185	22202	11.10	0.23	0.13
8	jobQ2MTdeep	50	5	249	2969	59.38	0.34	0.19
9	jobQ3CF	2000	12	171	10967	5.48	0.45	0.16
10	jobQ3MT	2000	12	1014	12900	6.45	0.28	0.10
11	jobQ3BOTH	2000	12	1185	23867	11.93	0.40	0.14
12	jobQ3MTdeep	50	12	249	3196	63.92	0.41	0.14
13	Suicide	2000	4	124	13175	6.59	0.27	0.17

Table 1. Basic properties of the label sets we use. Density indicates the average number of labels per data item. “MVTD” (majority-voted-true-class deviation) and “RMSD” (root-mean-square deviation) are two divergence measures for estimating the uncertainty of different label sets, motivated by the literature on scale and outliers [17, 33, 44]. MVTD is the average (over all data items) weight of the weight of the most frequent label: $MVTD = 1 - \sum_{i=1}^{n} \max_j \{y'_{ij}\}/n$. RMSD is the L2 deviation from the average label: $RMSD = \sum_{i=1}^{n} \sqrt{(y_i - \hat{y})^T (y_i - \hat{y})}/n$, where $\hat{y}$ is the average label distribution over all data. After adding labels (ID 4, 8, and 12 in Table 1), we obtained jobQ1MT-new, jobQ2MT-new, and jobQ3MT-new, respectively. We further integrated labels from both platforms to form jobQ1BOTH-new, jobQ2BOTH-new, and jobQ3BOTH-new.

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**)

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

Fig. 3. Our work-related annotation task contains these three questions.

This resulted six distinct *label sets*, one for each choice of platform and question, where each question and each platform, each data item has labels from five crowdworkers. We additionally constructed three additional label sets by combining the labels from both crowdsourcing platforms (denoted **BOTH**), so that each tweet has ten labels.

2.1.2 Suicide-themed data. We obtained another data set of 2000 tweets, filtered for suicide-related discourse [22]. Labels come from 122 CrowdFlower workers and 2 suicide prevention domain experts. For each tweet, five crowdworkers chose the label that described its content from four possible choices: A. *Suicidal thoughts*, B. *Supportive messages or helpful information*, C. *Reaction to suicide news/movie/music* and D. *Others*. Experts were invited to the second stage to annotate the tweets without unanimous labels from five crowdworkers. Thus tweet can have up to 7 labels, from crowdworkers and experts.

2.1.3 Data splits. Due to the expense of obtaining detailed samples from the populations of crowdworkers, we used two different train/dev/test splits.

Broad split. We randomly split each 2,000-tweet dataset into 1000/500/500 train/dev/test sets.

Deep split. We randomly split the job-related dataset only into 1500/540/50 train/dev/test sets. For each item in the 50-item held-out test set, we obtain 50 additional labels from new AMT crowdworkers.

2.2 Aggregation strategies

For a given data set with items $i \in \{1, \ldots, n\}$ and label choice $j \in \{1, \ldots, d\}$, let y_{ij} denote the number of crowdworkers who select label j for data item i. Thus $\mathbf{y}_i$ is a distribution over labels for data item i. We will sometimes abuse notation and also use $\mathbf{y}_i$ to denote the probability distribution obtained by normalizing the label distribution.

Aggregation composes two substages: *clustering* (including *no clustering*) and *reduction*, which depends on whether or not *no clustering* is the strategy used. We discuss this case first.

2.2.1 Reduction strategies when no clustering is used.

Majority. Typically, when annotators disagree on which label is best for a data item x_i, majority voting is used to determine a single gold-standard label: $\hat{y}_i = \underset{j \in \{1,\ldots,d\}}{\arg\max} \{y_{ij}\}$.

Repeated. This strategy assumes each (data item, label) is a separate data item, e.g., if three annotators choose to label 'A.' then we make three identical copies of the data in each training epoch. The model effectively weighs each choice by the number of times it is selected, with the goal of learning a single label, and treats each empirical label distribution as a Bayesian model of the degree of belief in each label choice.

Probability. This is a baseline method for predicting population distributions over label choices. Instead of training on a single label choice for each data item, it uses a d-dimensional vector representing the distribution $\mathbf{y}_i$ of labels for data item i as a probability distribution (which by abuse of notation we also call $\mathbf{y}_i$). It effectively treats each empirical label distribution as a frequentist sample of the true distribution of beliefs (though it crucially does not capture the degree of belief labels, either individually or collective).

2.2.2 Clustering strategies. These associate with each data item a probability distribution $\mathbf{z}_i$ over a finite number p of clusters, i.e., a mixture of models, over the space of empirical label distributions $\mathbf{y}_i$. According to our main hypothesis, pooling labels by cluster reveals the true label distributions underlying our empirical distributions, thus amplifying the labeling power of each crowdworker. We can thus associate with each cluster $k \in \{1, \ldots, p\}$ a distribution θ_k over the label choices. This is simply the cluster centroid if the strategy has one (like MMM and GMM below), or the weighted average ($\theta_k = \sum_i z_{ik}\mathbf{y}_i/n$) of the labels (as in DS and LDA below; we call them "centroids" in either case). Our goal is to improve prediction accuracy by replacing each empirical label $\mathbf{y}_i$ with one based on its cluster likelihoods $\mathbf{z}_i$ and cluster-wide label distributions θ_k.

We consider five different clustering strategies: the multinomial mixture model (**MMM**), the Gaussian mixture model[7] (**GMM**), Dawid and Skene's model [8] for selecting labels conditioned on annotator accuracy (**DS**) [9] and latent Dirichlet allocation[9] (**LDA**) [5]. We wrote our own MMM from scratch. We get two distinct strategies from LDA by, in addition to clustering over empirical labels, also clustering on bag-of-word representations of each data item's text, i.e., as LDA is most commonly used.

Though rather elementary, these models collectively provide an informative experimental basis for testing our central hypothesis, i.e., that label distributions in subjective domains are clustered around a finite number of true label distribution. According to this hypothesis, the model that best describes subjective domains should be MMM since it is a generative model where each centroid is defined as a distribution of which each cluster item is a sample. (By contrast, the centroid of GMM has a very different generative interpretation—i.e., as a parameter of a multivariate Gaussian distribution—even though in both models they are the (weighted) means of their respective cluster items.)

Although DS and LDA are not, strictly speaking, clustering models, we can easily obtain cluster-like latent classes—along with likelihood estimates—by integrating over the users (for David and Skene) or the data items (for LDA). Moreover, both models provide useful comparisons to our true clustering models. In particular, DS is widely-used in collaborative filtering settings, of which this can be seen as an example. This model incorporates labeler accuracy and is effective in settings where a labeler provides many examples. In our setting, which uses microtask crowdworkers, anyone labeler only provides ten or so examples (see Table 1), and so we would not expect this model to fit our data especially well.

LDA is very similar to MMM, though it is more commonly used, in part because it tends to be a better fit, both hypothetically and empirically, for more problems, but also because estimating prior distributions is computationally more efficient (in our case, we sidestep and use a maximum likelihood estimator for MMM, but not for LDA). The main difference between these models is in how data is generated. In MMM an empirical label distribution is assumed to come from choosing a cluster, then choosing all samples from that one chosen cluster. In LDA we choose a new cluster for each sample. If each cluster represented the beliefs of an individual, this might make sense, especially if we had a lot of data about the labeling preferences of individual labelers. However, since that is not the case in our setting, and since we are assuming that each cluster represents the distribution of beliefs across society, MMM makes more sense as the best model to fit our hypothesis.

Except for DS, each model requires the number of clusters p as a hyperparameter. We considered all values for p between roughly half and twice the number of label choices for each question. We investigated several model selection strategies, including some of the methods described below, and discovered that numbers they provided were roughly correlated. Furthermore, many of these strategies were designed for specific models or are based on strong prior assumptions. We ultimately chose the native likelihood function of each model, because we felt it provided the most externally consistent strategy for choosing the best p within each clustering strategy, even though it cannot really be used to compare models from different families.

As the estimators for these models are stochastic and/or sensitive to initial conditions, for every model and every choice of hyperparameters, we ran 100 trials on the training data and chose the model with the highest estimated likelihood.

[7] http://scikit-learn.org/

[8] Adapted from https://github.com/dallascard/dawid_skene

[9] Adapted from https://radimrehurek.com/gensim/

2.2.3 Cluster-based reduction strategies. We use these strategies to replace each empirical label distribution $\mathbf{y}_i$ with one based on the cluster centroids $\{\theta_k\}$ and the likelihood of i belonging to each cluster $\mathbf{z}_i$. *Maximum a posteriori* (**Max**) selection replaces $\mathbf{y}_i$ with the most likely cluster centroid $\theta_k : k = \arg\max_k z_{ik}$ and *expected distribution (**Avg**)* replaces $\mathbf{y}_i$ by integrating out the clusters $\sum_k z_{ik}\theta_k$.

Note that the integration step we use to produce aggregate distributions from LDA or DS essentially applies the Avg reduction to each model and that the Max reduction does not have a reasonable interpretation for these models (other than selecting the most likely label, which we can do more directly by simply not clustering). Yet understanding the performance differences between these two reduction strategies stands to yield important insights into the clustering hypothesis. If the clusters can discover the true representative label distribution underlying each empirical distribution, then we would expect predictive models to perform better using the Max strategy for training data, as it commits to a single distribution. Since LDA and DS cannot support both reductions, and thus deny us this important observation, we used only MMM- and GMM- based aggregations (to which either reduction can apply) as inputs to the supervised learning phase.

2.3 Supervised learning methods

We built various text-based supervised classifiers based on a single convolutional neural network (CNN) architecture as illustrated in Figure 4, using Keras with a Tensorflow back end. The differences among the model inputs are rooted in the aggregation strategies used.

CNNs have been used for various sentiment analysis and topic categorization tasks [19] and proved effective across a wide range of corpora. Each takes the text of a tweet as input and outputs a predicted label distribution.

The CNN architecture we use consists of an input layer composing concatenated pre-trained word embeddings, a convolutional layer with numerous filters, a max-pooling layer which captures the most significant feature, and a softmax classifier which outputs the probability distribution over labels/classes. We tested this supervised approach with various label aggregation strategies to obtain the ground truth labels, including clustering approaches, in our text classification experiments.

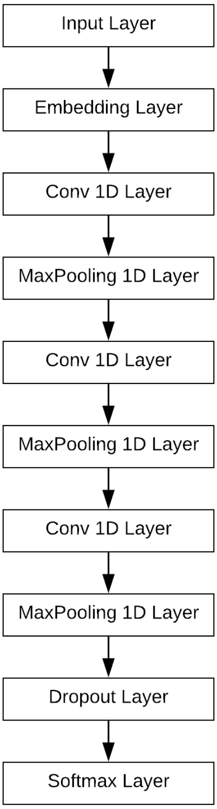

Fig. 4. The convolutional neural networks used in our text-based supervised experiments.

The hyper-parameter settings of the CNN architecture depend on the splits of datasets. We use the GloVe pre-trained word embeddings trained particularly on a Twitter corpus with 2B tweets [32]. We set the vector size of the word embeddings as 100 through our experiments. In our text pre-processing step, we keep the most common 20,000 words and pad the sentence up to 1,000 tokens. We use the *Adam* optimizer to minimize the loss function [20]. We set the batch size as 32 and the number of epochs to train the model as 25.

2.4 Testing

For each data item i, we now have three associated probability distributions: the empirical distribution of labels $\mathbf{y}_i$, the likelihood distribution over the clusters $\mathbf{z}_i$, and a new label distribution from the aggregation phase $\mathbf{w}_i$.

For the sake of using the clustering strategies to test our main hypothesis, we argue that the set of distributions $\theta_1, \ldots, \theta_p$ is a good fit for the hypothesis if, in addition to maximizing likelihood, the entropy over the cluster likelihoods $H(\mathbf{z}_i)$ is less than that of the empirical distributions $H(\mathbf{y}_i)$. However, these entropies cannot be directly compared because the number d of alternatives in the label set may be different from the number p of clusters. So we normalize by dividing by the logarithm of the number of items in each distribution. We call this the *entropy gap* (**EG**): $H(y_i)/\log d - H(z_i)/\log p$. This score applies to any label aggregation model or clustering approach that has likelihoods associated with each (data point, cluster) pair and where each point can be interpreted as a probability distribution. The danger with this score is that it is easy to "cheat" to get a good score, say, by assigning all data items to the same cluster. Since, however, we select our models based on maximum likelihood, we use this metric honestly here.

Another useful, and standard test is the **Kullback–Leibler divergence**, which measures how one probability distribution diverges from a second one [21]. For discrete probability distributions P (say, $\mathbf{y}_i$) and Q (say, $\mathbf{w}_i$, or, later, the label predicted by the CNN) it is: $D_{KL}(P||Q) = -\sum_j P(j) \log \frac{Q(j)}{P(j)}$.

We also use KL divergence to evaluate the performance of the CNN model (entropy gap does not make sense here). In addition, *KL1* measures the divergence from the CNN predicted probability to the empirical distribution of labels $\mathbf{y}_i$, and, when clustering is used. *KL2* measures the divergence from the CNN predicted probability to the label distribution from the aggregation phase $\mathbf{w}_i$. Additionally, **Score** is the loss (cost) function—*categorical cross entropy*—used to train the CNN. **Accuracy** measures how often the prediction have the maximum probability in the same class as the true value does. Note that KL divergence and cross entropy are standard tools for comparing probability distributions, while accuracy requires us to convert each distribution into a single scalar label.

3 RESULTS

3.1 Testing the clustering hypothesis

Table 2, 3, 4 show the performance results for each X_{Test} of datasets in Table 1 using the best model selected by the likelihood criterion.

Since we only had 50 data items with 50 extra labels, we tried clustering them visually using histograms. Figure 5 shows that the labels do appear to group clearly into seven clusters. We describe the tweets that fall into each cluster.

Group 1 (Red) distributions have most of their mass on label choices *Getting hired/job seeking* and *None of the above, but job-related.* Here, all the tweets in this group were 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. In contrast, Group 2 (cyan) has almost all the mass exclusively on *Getting hired/job seeking* (e.g., *got the job*). The third group (brown) clusters around *Complaining about work* and *Going to work*, suggesting a topic about complaining about having to go to work. Group four (green) are a set of tweets complaining about work while at work. Groups five and six (blue and orange) have most of their labels on *None of the above, but job-related* and *Not job-related.* Group six (where *Not job-related* was more frequent than *None of the above*) were mostly about road work. Group five (where *None of the above* was more frequent and complicated. It seemed to contain cases where work was mentioned, but was central to the other topics (e.g., *TODAY AT WORK I*

Broad split CL	**MMM**	LDA_{label}	**GMM**	LDA_{text}	**FMM**	**DPMM**
jobQ1CF	10	9	4	2	11	6
jobQ1MT	11	11	4	2	2	6
jobQ1BOTH	11	2	2	2	3	6
jobQ2CF	11	10	3	2	10	6
jobQ2MT	2	11	4	2	2	6
jobQ2BOTH	2	11	2	4	2	6
jobQ3CF	19	18	5	5	17	6
jobQ3MT	5	14	5	5	7	6
jobQ3BOTH	5	18	15	5	5	6
RWsuicide	8	7	2	2		

Deep split CL	**MMM**	LDA_{label}	**GMM**	LDA_{text}	**FMM**	**DPMM**
jobQ1CF	11	9	11	3	10	6
jobQ1MT-new	2	11	2	2	2	6
jobQ1BOTH-new	2	11	2	2	2	6
jobQ2CF	11	10	2	2	11	6
jobQ2MT-new	2	11	2	3	4	6
jobQ2BOTH-new	2	8	2	2	3	6
jobQ3CF	19	19	10	6	17	6
jobQ3MT-new	5	15	19	8	11	6
jobQ3BOTH-new	5	11	17	6	7	6

Table 2. Numbers of clusters for the optimal label aggregation model we achieved on each dataset using two splits. "CL": Number of clusters in the best model.

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.*

3.2 Supervised learning

In Table 5-10, we show the score, accuracy, and KL divergence metrics for a series of CNN-based text classifiers for the job (Broad split: 5-7, Deep split: 8-10) and suicide datasets built with different label aggregation approaches.

4 DISCUSSION

4.1 Testing the clustering hypothesis

Among the aggregation methods tested, MMM and LDA had the best KL scores (Table 4). Since this metric is the best honest score for testing fitness across models, and since the MMM and LDA are better fits for the clustering hypothesis, this seems to partially support the hypothesis.

This would seem to suggest that LDA is a better model for the underlying space of label distributions, and one reason for this could be because the labels indeed depend on independent *classes* of labelers (or possibly even individual labelers, as witnessed by the somewhat unexpectedly good performance of DS). Note that another explanation could be that even in the MMM model the clusters have a substantial enough amount of uncertainty (captured by the $\mathbf{w}_i$ distributions) and that averaging over this uncertainty leads to better predictions. This is a common phenomenon, even in situations where the data is known to be generated from a single model; that is, maximum a posteriori estimates often underperform fully Bayesian ones.

We were surprised by how much worse GMM performed compared to the other methods (except LDA_{text}, which uses a different feature set than the others and so is *a priori* an outlier), as for

Broad split EG	**MMM**	LDA_{label}	**GMM**	**DS**	LDA_{text}	**FMM**	**DPMM**
jobQ1CF	0.07	0.18	0.50	**0.70**	0.09	0.37	0.33
jobQ1MT	0.06	-0.06	**0.24**	0.19	-0.14	0.16	0.08
jobQ1BOTH	0.29	0.23	0.46	**0.68**	0.07	0.38	0.30
jobQ2CF	0.03	0.06	0.37	**0.54**	-0.02	0.25	0.18
jobQ2MT	0.15	-0.09	**0.20**	0.18	-0.19	0.15	-0.01
jobQ2BOTH	0.30	0.12	0.36	**0.54**	-0.03	0.30	0.20
jobQ3CF	0.00	0.00	0.40	**0.97**	-0.01	0.24	0.11
jobQ3MT	0.16	-0.10	0.24	**0.36**	-0.18	0.15	0.14
jobQ3BOTH	0.32	0.12	0.41	**1.00**	-0.02	0.34	0.33
RWsuicide	0.22	0.12	0.41	**0.58**	0.08		

Deep split EG	**MMM**	LDA_{label}	**GMM**	**DS**	LDA_{text}	**FMM**	**DPMM**
jobQ1CF	0.03	0.19	0.50	**0.64**	0.17	0.35	0.27
jobQ1MT-new	0.63	0.19	0.66	**1.05**	0.31	0.66	0.64
jobQ1BOTH-new	0.61	0.35	0.67	**1.08**	0.31	0.67	0.64
jobQ2CF	0.05	0.08	0.41	**0.57**	0.06	0.25	0.13
jobQ2MT-new	0.53	0.13	0.53	**0.85**	0.18	0.52	0.52
jobQ2BOTH-new	0.54	0.25	0.54	**0.87**	0.18	0.54	0.52
jobQ3CF	-0.04	-0.02	0.34	**0.71**	-0.06	0.16	0.02
jobQ3MT-new	0.45	0.09	0.47	**1.02**	0.05	0.46	0.46
jobQ3BOTH-new	0.47	0.21	0.48	**1.12**	0.08	0.47	0.47

Table 3. Entropy gap obtained using the optimal label aggregation model on each dataset using two splits. "EG": Normalized entropy gap (i.e., the average entropy gap per data item). The highest EG for each dataset is highlighted in bold.

large samples GMM and MMM rather similar. However, the sample sizes (number of labels) we use here are normal for many supervised learning tasks, and at this scale, the differences appear to be significant.

Regarding EG (Table 3): that GMM and DS tend to outperform the other models is not too surprising, given that EG is not honest (see discussion in the testing subsection). And we expected LDA to perform poorly on this metric, due to the fact that, under LDA, most empirical distributions are drawn from multiple clusters. Thus we would expect the cluster likelihood distribution to have higher entropy than in the MMM model (which assumes all labels are drawn from a single cluster).

4.2 Supervised learning

Starting again with KL divergence (Tables 7 and 10), CNNs trained and tested on MMM_{AVG} outperform all other models most of the time, with no-clustering, probability-based CNNs a close second. GMM_{Avg} has some very good and very bad results, and the relative dominance of MMM_{AVG} recedes when the deep label distributions are used for evaluation.

Together, these results show that learning over the entire distribution of labels is feasible and that using clustering to aggregate labels sometimes results in better performance.

What was not expected (though consistent with our clustering hypothesis tests, where *LDA* outperformed *MMM*) is that MMM_{Avg} outperforms MMM_{Max}. Better MMM_{Max} performance would seem to be more consistent with the hypothesis that the clustering algorithm can discover the true underlying label distributions. Instead, MMM_{Avg} draws from each of the predicted ground truth label distributions, yielding distributions that very similar in construction to those produced by LDA.

Broad split KL	**MMM**	LDA_{label}	**GMM**	**DS**	LDA_{text}	**FMM**	**DPMM**
jobQ1CF	0.35	**0.23**	0.53	0.38	0.53	0.40	0.39
jobQ1MT	0.19	**0.18**	0.68	0.38	0.68	0.36	0.22
jobQ1BOTH	**0.20**	0.40	0.46	0.27	0.46	0.22	0.20
jobQ2CF	0.26	**0.19**	0.54	0.32	0.54	0.33	0.31
jobQ2MT	0.36	**0.15**	0.74	0.37	0.74	0.36	0.13
jobQ2BOTH	0.28	**0.17**	0.51	0.26	0.50	0.28	0.17
jobQ3CF	**0.51**	0.52	1.00	0.59	0.97	0.63	0.63
jobQ3MT	0.50	**0.33**	1.15	0.69	1.11	0.41	0.44
jobQ3BOTH	0.45	**0.35**	0.82	0.47	0.86	0.45	0.40
RWsuicide	0.22	**0.20**	0.57	0.26	0.67		
Deep split KL	**MMM**	LDA_{label}	**GMM**	**DS**	LDA_{text}	**FMM**	**DPMM**
jobQ1CF	0.30	**0.24**	0.57	0.44	0.63	0.41	0.40
jobQ1MT-new	0.20	**0.07**	0.39	0.09	0.38	0.20	0.10
jobQ1BOTH-new	0.21	**0.06**	0.38	0.09	0.37	0.20	0.07
jobQ2CF	0.24	**0.20**	0.65	0.39	0.65	0.28	0.28
jobQ2MT-new	0.26	**0.09**	0.50	0.10	0.49	0.11	0.11
jobQ2BOTH-new	0.25	**0.09**	0.48	0.10	0.45	0.13	0.08
jobQ3CF	0.29	**0.27**	0.97	0.48	0.86	0.49	0.41
jobQ3MT-new	0.20	**0.17**	0.51	0.27	0.58	0.14	0.23
jobQ3BOTH-new	**0.18**	**0.18**	0.64	0.25	0.57	0.15	0.17

Table 4. KL divergence obtained using the optimal label aggregation model on each dataset using two splits. "KL": Kullback–Leibler divergence. The lowest KL divergence for each dataset is highlighted in bold.

Broad split Score	**majority**	**repeated**	**probability**	MMM_{Max}	MMM_{Avg}	GMM_{Max}	GMM_{Avg}	LDA_{Max}	LDA_{Avg}
jobQ1CF	1.89	1.59	1.72	1.34	**1.33**	1.75	1.71	1.40	1.36
jobQ1MT	1.76	1.22	**1.11**	1.27	1.27	1.57	1.51	0.84	0.98
jobQ1BOTH	1.40	**1.18**	1.22	1.31	1.31	1.31	1.28	1.02	1.14
jobQ2CF	1.68	1.57	1.40	1.19	**1.17**	1.55	1.58	1.07	1.18
jobQ2MT	2.05	1.18	**1.11**	1.30	1.22	1.34	1.46	0.91	0.98
jobQ2BOTH	1.55	1.08	**1.06**	1.23	1.23	1.20	1.22	0.97	1.03
jobQ3CF	4.86	2.19	2.15	2.06	**2.05**	2.25	2.21	2.03	2.01
jobQ3MT	3.17	1.71	**1.66**	1.98	1.92	2.23	2.21	1.76	1.76
jobQ3BOTH	3.42	**1.77**	1.79	1.98	2.00	2.03	2.01	1.70	1.79
RWsuicide	1.37	1.07	**1.05**	1.41	1.37	9.27	9.27	1.00	1.06

Table 5. Scores of CNN-based text classification experiments with different aggregation models, using the Broad split. The lowest score for each dataset is highlighted in bold.

Among the clustering models, as expected, KL2 outperforms KL1, and this supports our hypothesis by showing that principled aggregation processes are effective for training *and* prediction.

Also of interest are the accuracy tests (Tables 6 and 9). Since this test requires the model to produce a single "best" label, and since clustering is used here for preserving diversity in the label distributions, we expected the clustering methods to underperform the no-clustering methods. Among the no-clustering methods, majority can be seen as a the standard approach of learning a single label for each data item, while probability attempts to learn the frequentist, empirical labels, even though (for the purpose of the accuracy test) it only reveals one label. Repeated is implicitly a Bayesian approach. Except for jobQ3BOTH-new, majority and probability give nearly the same performance, which suggests that modeling the underlying distribution, even without clustering,

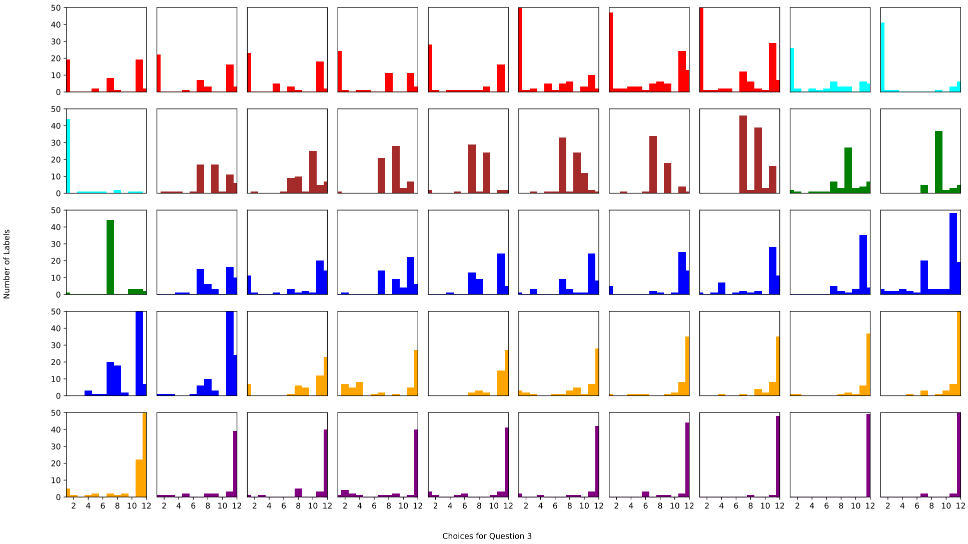

Fig. 5. Histograms of the label distributions for the 50 job-related tweets having 50 extra labels each. The X axis ranges from 1 to 12, representing the Q3 choice indices in Figure 3.

Broad split ACC	majority	repeated	probability	MMM_{Max}	MMM_{Avg}	GMM_{Max}	GMM_{Avg}	LDA_{Max}	LDA_{Avg}
jobQ1CF	0.73	0.53	0.72	0.78	0.81	0.95	**0.98**	0.58	0.73
jobQ1MT	**0.80**	0.72	0.79	0.56	0.57	0.67	0.65	0.76	0.82
jobQ1BOTH	**0.82**	0.64	0.81	0.57	0.56	0.76	0.78	0.76	0.81
jobQ2CF	0.73	0.63	**0.79**	0.71	0.72	0.62	0.64	0.94	0.96
jobQ2MT	**0.73**	0.68	**0.73**	0.48	0.55	0.55	0.58	0.71	0.75
jobQ2BOTH	**0.76**	0.65	**0.76**	0.63	0.60	0.58	0.59	0.71	0.78
jobQ3CF	0.36	0.31	0.41	**0.47**	0.46	0.32	0.32	0.45	0.50
jobQ3MT	**0.53**	0.45	0.51	0.26	0.30	0.28	0.29	0.49	0.49
jobQ3BOTH	0.48	0.42	0.53	0.31	0.29	**0.62**	0.55	0.46	0.52
RWsuicide	0.81	0.65	0.78	0.18	0.27	**1.00**	**1.00**	0.76	0.76

Table 6. Accuracy of CNN-based text classification experiments with different aggregation models, using the Broad split. The highest accuracy for each dataset is highlighted in bold.

generally does not degrade the accuracy of single-label models. That repeated underperforms the other perhaps reflects the reality that each empirical distribution represents a sample of population beliefs, rather than degree of belief.

One important and obvious limitation of this work is that uncertainty in human labeling is caused by many things other than subjectivity, including data encoding errors and communication ambiguities [3, 8, 46], lack of sufficient information [6, 8, 14], and unreliable annotators and their bias [14]. We do not attempt to quantity whether the uncertainty we observe is due to these other causes or to subjectivity (i.e., varying user perspectives). We hope to explore this avenue in future work.

In order to truly understand the social impact of representative learning, we need to know the underlying demographics of the sampling frames in question, in this case AMT and CrowdFlower.

Broad split KL1/2	**majority**	**repeated**	**probability**	MMM_{Max}		MMM_{Avg}		GMM_{Max}		GMM_{Avg}		LDA_{Max}		LDA_{Max}	
	KL1	KL1	KL1	KL1	KL2	KL1	KL2	KL1	KL2	KL1	KL2	KL1	KL2	KL1	KL2
jobQ1CF	2.98	0.79	0.91	0.55	0.12	0.55	**0.07**	0.97	0.74	0.91	0.69	0.59	0.47	0.50	0.24
jobQ1MT	2.03	0.80	0.72	0.87	0.65	0.86	**0.62**	1.13	1.05	1.13	0.97	0.61	0.52	0.48	0.27
jobQ1BOTH	2.38	0.45	0.48	0.55	0.36	0.59	0.35	0.57	0.38	0.56	**0.33**	0.35	0.27	0.35	0.18
jobQ2CF	2.29	0.91	0.79	0.60	0.21	0.58	**0.15**	0.95	0.78	0.94	0.81	0.65	0.13	0.56	0.08
jobQ2MT	2.10	0.80	0.78	1.02	0.81	0.99	**0.70**	1.22	0.98	1.33	1.09	0.69	0.67	0.55	0.35
jobQ2BOTH	2.12	0.49	0.47	0.63	0.48	0.67	**0.46**	0.68	0.48	0.67	0.48	0.42	0.37	0.37	0.20
jobQ3CF	4.20	1.66	1.14	1.09	0.31	1.07	**0.25**	1.31	0.68	1.27	0.64	1.02	0.66	0.90	0.33
jobQ3MT	3.18	2.24	1.05	1.43	1.04	1.47	**0.90**	1.75	1.32	1.78	1.28	1.11	0.54	1.03	0.27
jobQ3BOTH	3.38	1.40	0.77	1.07	0.62	1.05	0.60	1.04	0.49	1.07	**0.47**	0.78	0.62	0.69	0.38
RWsuicide	2.16	1.40	**0.45**	0.83	0.69	0.82	0.61	0.88	13.62	0.88	13.62	0.42	0.33	0.38	0.18

Table 7. Kullback–Leibler divergence of CNN-based text classification experiments with different aggregation models, using the Broad split. The lowest KL divergence for each dataset is highlighted in bold.

Deep split Score	**majority**	**repeated**	**probability**	MMM_{Max}	MMM_{Avg}	GMM_{Max}	GMM_{Avg}	LDA_{Max}	LDA_{Avg}
jobQ1CF	2.85	1.58	1.72	1.36	**1.33**	1.74	1.69	1.41	1.41
jobQ1MT-new	2.47	1.57	1.63	**1.25**	1.29	1.59	2.12	1.02	1.30
jobQ1BOTH-new	2.54	1.42	**1.32**	**1.32**	1.35	1.49	1.39	1.31	1.31
jobQ2CF	3.65	1.23	1.32	1.17	**1.16**	1.35	1.40	1.34	1.24
jobQ2MT-new	1.48	1.38	1.45	1.25	1.24	**1.11**	1.27	1.14	1.17
jobQ2BOTH-new	2.54	1.14	1.15	1.26	1.24	1.09	**1.07**	1.12	1.13
jobQ3CF	4.07	1.86	**1.84**	2.05	2.02	2.29	2.30	1.73	1.91
jobQ3MT-new	4.90	**1.94**	**1.94**	1.98	2.02	2.24	2.06	1.67	1.66
jobQ3BOTH-new	3.62	**1.65**	1.82	2.06	1.95	2.26	2.25	1.49	1.65

Table 8. Scores of CNN-based text classification experiments with different aggregation models, using the Deep split. The lowest score for each dataset is highlighted in bold.

Deep split ACC	**majority**	**repeated**	**probability**	MMM_{Max}	MMM_{Avg}	GMM_{Max}	GMM_{Avg}	LDA_{Max}	LDA_{Avg}
jobQ1CF	0.62	0.47	0.58	0.78	**0.86**	0.80	0.80	0.54	0.56
jobQ1MT-new	**0.72**	0.53	0.70	0.58	0.58	0.66	0.56	0.72	0.74
jobQ1BOTH-new	0.72	0.51	0.70	0.62	0.58	**0.90**	**0.90**	0.60	0.70
jobQ2CF	0.60	0.53	0.52	0.76	0.74	**0.82**	**0.82**	0.48	0.62
jobQ2MT-new	**0.72**	0.57	0.70	0.58	0.48	0.64	0.56	0.66	0.74
jobQ2BOTH-new	0.72	0.54	**0.76**	0.54	0.60	0.56	0.62	0.68	0.68
jobQ3CF	0.46	0.40	**0.48**	0.16	0.46	0.30	0.30	0.50	0.46
jobQ3MT-new	**0.54**	0.43	**0.54**	0.14	0.24	0.24	0.26	0.48	0.48
jobQ3BOTH-new	**0.62**	0.46	0.48	0.20	0.22	0.40	0.38	0.56	0.56

Table 9. Accuracy of CNN-based text classification experiments with different aggregation models, using the Deep split. The highest accuracy for each dataset is highlighted in bold.

Several studies have investigated these demographics [10, 11, 18, 38]. Among the findings: Mechanical Turk pulls most of its workforce from the United States, whereas CrowdFlower's workforce has proportionally higher levels of participation from smaller countries, like Venezuela. The male to female ratio is similar on both platforms with more female workers than male. A majority of contributors have some college education, of which most have a bachelor's degree. The worker population on both platforms is dynamic and changes frequently, but the number of workers available is steady, so every year some new workers join and balance the workers who quit contributing. The majority of workers fall in the legal working age in the US, most of which are young workers of age group 20-35. These workers earn below the median salary range in the US. The American

Deep split KL1/2	**majority**	**repeated**	**probability**	MMM_{Max}		MMM_{Avg}		GMM_{Max}		GMM_{Avg}		LDA_{Max}		LDA_{Avg}	
	KL1	KL1	KL1	KL1	KL2	KL1	KL2	KL1	KL2	KL1	KL2	KL1	KL2	KL1	KL2
jobQ1CF	3.09	0.77	0.90	0.75	0.13	0.69	**0.07**	1.10	0.69	1.03	0.63	0.58	0.39	0.60	0.23
jobQ1MT-new	2.94	**0.47**	0.54	0.76	0.64	0.73	0.67	1.83	1.08	1.80	1.57	0.50	0.47	0.28	0.25
jobQ1BOTH-new	2.90	0.34	**0.24**	0.56	0.39	0.51	0.41	0.94	0.43	0.69	0.32	0.24	0.38	0.21	0.17
jobQ2CF	3.07	0.57	0.65	0.82	0.18	0.76	**0.13**	1.08	0.56	1.28	0.58	0.57	0.49	0.50	0.24
jobQ2MT-new	1.90	**0.50**	0.58	0.84	0.77	0.82	0.75	1.62	0.68	1.65	0.84	0.76	0.76	0.31	0.30
jobQ2BOTH-new	2.90	**0.27**	0.28	0.77	0.52	0.77	0.49	0.92	0.37	0.94	0.34	0.32	0.35	0.26	0.20
jobQ3CF	3.71	1.45	1.00	1.22	0.34	1.21	**0.21**	1.63	0.63	1.25	0.64	0.92	0.65	0.91	0.36
jobQ3MT-new	3.95	1.98	**0.77**	1.13	1.13	1.07	1.06	1.85	1.21	1.36	1.03	0.97	1.20	0.51	0.42
jobQ3BOTH-new	3.33	1.13	0.63	0.91	0.76	0.98	**0.60**	0.95	0.67	1.01	0.64	0.51	0.49	0.45	0.33

Table 10. Kullback–Leibler divergence of CNN-based text classification experiments with different aggregation models, using the Deep split. The lowest KL divergence for each dataset is highlighted in bold.

racial composition is mostly white. According to Ellie et al. [31] workers speak a diverse set of languages. According to Huff and Tingley [15] those working as office and administrative support are major contributors to AMT.

5 RELATED WORK

It is common in supervised learning settings to model data labels as probability distributions, as we do here, though the similarities are somewhat superficial. In most machine learning problems these probabilities are *Bayesian*, meaning that the distributions represent uncertainty or degree of belief. In sharp contrast, our label probabilities are *frequentist* (though the some of the model probabilities used for clustering are Bayesian), i.e., they literally represent an estimate of the frequency of events (i.e., labels chosen) in a population sample.

As mentioned in the discussion, there are many sources for uncertainty when humans in the loop are concerned [16, 25, 26, 39, 40]. However, most such studies into this matter assume that there is an underlying, if unknown, true label for each data item and do not account for the subjective nature of human comprehensions and beliefs, i.e., more than one answer is reasonably correct and acceptable. Two broad research areas overlapping with our subjective domain research question include *recommender systems* and *multi-label learning problems* [12].

Recommender systems [4] study the tastes and preferences of individuals, typically in online commercial settings. The goal of such systems is to personalize the shopping, viewing, or playing experience of the users of such system, and they rely on copius amounts of data on the users and in grouping users into groups with similar tastes. Here we are interested in how populations beliefs, not tastes, vary, and although modeling users and group of users is of interest to us (particularly to distinguish between different sorts of expertise on the annotation domain), in many annotation setting, such as in crowdsourcing, little information on the annotators may be available.

Multilabel classification [12, 13, 24, 27, 30, 34, 35, 35, 36, 41–43, 45] allows for each data item to simultaneously belong to multiple classes [7, 24]. However, it is possible for there to be multiple valid labels, even when there is no disagreement among labelers. It is often important to know when multiplicity is due to disagreement, especially when such disagreements fall along key demographic boundaries, and indicate important but opposing perspectives that should be equally preserved in the predictive model. Multilabel models are not designed to detect such disagreement. Rather, they are designed to detect a rich collection of labels, individualized to each data item, and with no frequentist representation of the diversity of underlying population beliefs. By contrast, we seek to throw disagreement into high relief by assuming that label sets fall into a small number of stereotypical classes, which can be discovered through clustering in the space of label distributions.

6 CONCLUSION

We study the problem of learning to predict the underlying diversity of beliefs present in supervised learning domains. We compare the performance of predictive models that are trained on the empirical distribution of labels produced by crowdworkers to those that collapse those labels to a single ground truth value. Our results show that it is feasible to predict such distributions over labels. Doing so is an important first step in producing intelligent agents that understand the diversity of beliefs in society.

We also studied the use of clustering to pool and aggregate labels in order to reduce the costs of labeling in this richer domain. Our results suggest that such methods are effective, and though the reason may have to do with the underlying sources of subjectivity being limited, more research is needed to understand why. This paper provides a substantial framework of models and tests to further explore this question and others and advance though rigorous testing and evaluation socially-aware intelligent systems.

Indeed, our results suggest a number of next steps. For one, we regret not using LDA-based distributions in the supervised learning phase, since they seemed to perform so well in the aggregation phase. We also need to explore more powerful variants of MMM, including the standard fully Bayesian variant, Dirichlet-multinomial mixtures, and the standard nonparametric variant, Dirichlet process multinomial models.

This project was motivated by the need for active learning methods that are socially aware, and recognizing that the there was very little research in this area to build on. We hope to incorporate the lessons learned here into new active learning query strategies that make learning socially representative labels even more efficient.

ACKNOWLEDGEMENTS

Blind for review.

REFERENCES

[1] Crowdsourcing at Scale. 2013. Fact Evaluation Judgment Dataset. https://sites.google.com/site/crowdscale2013/shared-task/task-fact-eval

[2] Crowdsourcing at Scale. 2013. Sentiment Analysis Judgment Dataset. https://sites.google.com/site/crowdscale2013/shared-task/sentiment-analysis-judgment-data

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

[4] 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.

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

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

[7] Klaus Brinker, Johannes Fürnkranz, and Eyke Hüllermeier. 2006. A unified model for multilabel classification and ranking. In *Proceedings of the 2006 conference on ECAI 2006: 17th European Conference on Artificial Intelligence August 29–September 1, 2006, Riva del Garda, Italy*. IOS Press, 489–493.

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

[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] Djellel Difallah, Elena Filatova, and Panos Ipeirotis. 2018. Demographics and Dynamics of Mechanical Turk Workers. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*. ACM, New York, NY, USA, 135–143. https://doi.org/10.1145/3159652.3159661

[11] Djellel Eddine Difallah, Michele Catasta, Gianluca Demartini, Panagiotis G Ipeirotis, and Philippe Cudré-Mauroux. 2015. The dynamics of micro-task crowdsourcing: The case of amazon mturk. In *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 238–247.

[12] Johannes Fürnkranz. 2002. Round robin classification. *Journal of Machine Learning Research* 2, Mar (2002), 721–747.

[13] Shantanu Godbole and Sunita Sarawagi. 2004. Discriminative methods for multi-labeled classification. *Advances in knowledge discovery and data mining* (2004), 22–30.

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

[15] Connor Huff and Dustin Tingley. 2015. "Who are these people?" Evaluating the demographic characteristics and political preferences of MTurk survey respondents. *Research & Politics* 2, 3 (2015), 2053168015604648.

[16] 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.

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

[18] Panagiotis G Ipeirotis. 2010. Demographics of Mechanical Turk (Tech. Rep. No. CeDER-10-01). New York: New York University. (2010).

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

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

[21] Solomon Kullback. 1959. Statistics and Information theory. *J Wiley Sons, New York* (1959).

[22] 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

[23] 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

[24] Gjorgji Madjarov, Dragi Kocev, Dejan Gjorgjevikj, and Sašo Džeroski. 2012. An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition* 45, 9 (2012), 3084–3104.

[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] Eneldo Loza Mencía, Sang-Hyeun Park, and Johannes Fürnkranz. 2010. Efficient voting prediction for pairwise multilabel classification. *Neurocomputing* 73, 7 (2010), 1164–1176.

[28] Tom M Mitchell et al. 1997. Machine learning. WCB.

[29] Panos Ipeirotis. 2013. Get Another Label. https://github.com/ipeirotis/Get-Another-Label/tree/master/data

[30] Sang-Hyeun Park and Johannes Fürnkranz. 2007. Efficient pairwise classification. *Machine Learning: ECML 2007* (2007), 658–665.

[31] Ellie Pavlick, Matt Post, Ann Irvine, Dmitry Kachaev, and Chris Callison-Burch. 2014. The language demographics of amazon mechanical turk. *Transactions of the Association for Computational Linguistics* 2 (2014), 79–92.

[32] 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

[33] 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.

[34] Jesse Read. 2008. A pruned problem transformation method for multi-label classification. In *Proc. 2008 New Zealand Computer Science Research Student Conference (NZCSRS 2008)*, Vol. 143150.

[35] Jesse Read, Bernhard Pfahringer, and Geoff Holmes. 2008. Multi-label classification using ensembles of pruned sets. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 995–1000.

[36] Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. 2009. Classifier chains for multi-label classification. *Machine Learning and Knowledge Discovery in Databases* (2009), 254–269.

[37] Michael Riegler, Martha Larson, Concetto Spampinato, Pål Halvorsen, Mathias Lux, Jonas Markussen, Konstantin Pogorelov, Carsten Griwodz, and Håkon Stensland. 2016. Right inflight?: A dataset for exploring the automatic prediction of movies suitable for a watching situation. In *Proceedings of the 7th International Conference on Multimedia Systems*. ACM, 45.

[38] Joel Ross, Lilly Irani, M Silberman, Andrew Zaldivar, and Bill Tomlinson. 2010. Who are the crowdworkers?: shifting demographics in mechanical turk. In *CHI'10 extended abstracts on Human factors in computing systems*. ACM, 2863–2872.

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

[40] 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.

[41] Grigorios Tsoumakas and Ioannis Katakis. 2006. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining* 3, 3 (2006).

[42] Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. 2008. Effective and efficient multilabel classification in domains with large number of labels. In *Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD'08)*. 30–44.

[43] Grigorios Tsoumakas and Ioannis Vlahavas. 2007. Random k-labelsets: An ensemble method for multilabel classification. *Machine learning: ECML 2007* (2007), 406–417.

[44] 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.

[45] Ting-Fan Wu, Chih-Jen Lin, and Ruby C Weng. 2004. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research* 5, Aug (2004), 975–1005.

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

Received April 2018