Disagreement Matters: Preserving Label Diversity by Jointly Modeling Item and Annotator Label Distributions with *DisCo*

Tharindu Cyril Weerasooriya [1*], **Alexander G. Ororbia**[1], **Raj B. Bhensadadia**[1], **Ashiqur R. KhudaBukhsh**[1], **Christopher M. Homan**[1]
[1]Rochester Institute of Technology, USA
[*]cyriltcw@gmail.com

Abstract

This paper contains content that can be offensive or disturbing.

Annotator disagreement is common whenever human judgment is needed for supervised learning. It is conventional to assume that one label per item represents ground truth. However, this obscures minority opinions, if present. We regard "ground truth" as the distribution of all labels that a population of annotators could produce, if asked (and of which we only have a small sample). We next introduce *DisCo* (Distribution from Context), a simple neural model that learns to predict this distribution. The model takes annotator-item pairs, rather than items alone, as input, and performs inference by aggregating over all annotators. Despite its simplicity, our experiments show that, on six benchmark datasets, our model is competitive with, and frequently outperforms, other, more complex models that either do not model specific annotators or were not designed for label distribution learning.

1 Introduction

Human feedback remains a critical component as machine-learning-based systems play an ever larger role in society and our daily lives. Furthermore, when these systems fail, they assume that there is a single, correct answer in every case. Yet, due to differences in perception, context, experiences, attitudes that vary from person to person, and demographic differences, humans often disagree on what the "right response" should be. For instance, Binns et al. (2017) showed that female annotators frequently disagree with males on what constitutes offensive speech.

This prevalence of disagreement in human-labeled data has made *annotator modeling* a popular research problem. In its most elementary form, each item is assigned the majority label. More sophisticated approaches seek to understand annotator behavior (Dawid and Skene, 1979; Rodrigues and Pereira, 2018; Lakkaraju et al., 2015; Gordon et al., 2022) via machine learning. Yet *the vast majority of these approaches require or assume that ground truth is a single (but unknown) label (or collection of labels) and that any deviation from the ground truth label is indicative of poor quality.* Consequently, most models learn to discriminate between "good" and "bad" annotators (Lakkaraju et al., 2015; Rodrigues and Pereira, 2018) and resolve disagreement out of existence.

However, on problems such as hate speech detection, language complexity, or machine translation, disagreement may actually signify the views of vulnerable communities that should be preserved (Gray and Suri, 2019; Klenner et al., 2020; Basile, 2020; Prabhakaran et al., 2021), or even made predictable (Lakkaraju et al., 2015). A major barrier to achieving these goals is *annotator sparseness*. Human annotators are an expensive and often limiting factor in a learning loop. It is usually not feasible to collect enough annotations from each item in order to have confidence in them as a representative sample of the underlying population's response.

In this paper, we explore the idea that, in key settings, ground truth is more plainly seen as a distribution of labels representing the opinions and beliefs of a (partially observed) population of annotators, rather than a single label (or multi-label). In the extreme (as we do here), this approach ignores the near-certainty that some annotators are unreliable. However, the goal of modeling as precisely as possible annotator responses at the population level is a transparent, data-conservative approach to preserving annotator information, as opposed to the conventional approach of resolving disagreement before learning. Later in this work, we will discuss extensions for modeling annotator reliability.

We propose a new neural model, *DisCo* (*Distribution from Context*), designed to address the annotator sparsity problem. At training time, the model takes in as input a training example and an

Findings of the Association for Computational Linguistics: ACL 2023, pages 4679–4695
July 9-14, 2023

annotator id, and outputs three simultaneous predictions: the label the annotator gives the example, the distribution of all labels the example received (from all annotators who responded to it) and the distribution of all labels (over all examples) that the annotator provided. At inference time, it takes an unlabeled item as input and predicts the distribution of labels that it would receive from the population of annotators. It ties together two rather successful prior approaches: label distribution learning (LDL) (Geng, 2016) and item-annotator modeling (Dawid and Skene, 1979). Our models are publicly available.[1]

In this work, we address the following questions:

RQ1 How does the performance of DisCo compare to that of LDL approaches that do not model annotators?

RQ2 How does the performance of DisCo compare to that of non-LDL approaches that model annotators?

To answer these questions, we evaluate DisCo against three competitive models that exemplify, respectively, the conventional ground truth approach, a label distribution approach without annotator modeling, and an approach that models annotators but, unlike DisCo, is not purpose-built for distributional ground truth. We test these models on six benchmark datasets that contain annotator-item assignments and annotator-level labels. We evaluate our models on two different gold standards: the most frequent label and the label distribution.

2 Related Work

Our work is philosophically aligned with well-documented analyses of inherent annotator disagreement (Davani et al., 2021; Prabhakaran et al., 2021; Pavlick and Kwiatkowski, 2019) or annotator bias (Field and Tsvetkov, 2020). However, we are going beyond simply analyzing the difference. Rather, we seek to leverage the distribution of annotator responses as a signal to be learned for its own sake.

The study of annotator disagreement has a long history, coincident with the emergence of data-driven behavioral research (Cohen, 1960). Dawid and Skene (1979) introduced item-annotator tableau models. They use the multiple labels associated with each data item and each annotator to jointly estimate the ground truth label of each item as well as the error rate of each annotator. Their approach uses only the labels, not the data item features associated with them, and so, alone, this method cannot outperform supervised learning. Rather, it is used as the first of a two-step learning process, where the second step can be any supervised learning algorithm.

Later researchers put this model on a fully Bayesian foundation (Raykar et al., 2010; Kim and Ghahramani, 2012) or considered more complex models of annotators, ground truth, or both (Whitehill et al., 2009; Northcutt et al., 2019). Notably, (as spam is a common problem in crowdsourced label sets) several investigators distinguish between honest and dishonest annotators (Raykar and Yu, 2012; Hovy et al., 2013). More recently, investigators have studied clustering as an unsupervised approach to discover annotators with similar behavior (Venanzi et al., 2014; Lakkaraju et al., 2015). Yet all of these approaches are still based on the assumption that each item is associated with a single ground truth label.

Label distribution learning (LDL), in contrast, assumes that the ground truth itself is a distribution. However, this distribution does not necessarily come from a population of annotators (Geng, 2016; Gao et al., 2017; Wang and Geng, 2019; Zhang et al., 2020). Notably, LDL has proven useful in a diverse range of settings (Geng et al., 2014; Geng and Hou, 2015; Ren and Geng, 2017; Ling and Geng, 2018; Shirani et al., 2019; Yang et al., 2020; Liu et al., 2019a; Weerasooriya et al., 2020). Here, the goal is to predict the distribution of labels associated with an item rather than a single ground truth label. It is relatively natural, in this setting, to consider clustering together related data items in order to improve the ground truth estimates of label distributions, as several prior efforts have done, either in the feature space of the items (Zheng et al., 2018; Zeng et al., 2020; Xu et al., 2021) or directly in the label space of the items themselves (Liu et al., 2019b,a; Weerasooriya et al., 2020). Note that models that cluster only in the label space can only be used as the first step in a two-step supervised learning process (for the same reason that the David and Skene model can only be used in this way).

Our work is most closely aligned with others who seek not only to gain understanding of annotator disagreement, but to predict it for it's own sake. CrowdTruth (Aroyo and Welty, 2013; Dumitrache

[1] Experimental code available through https://github.com/Homan-Lab/disco

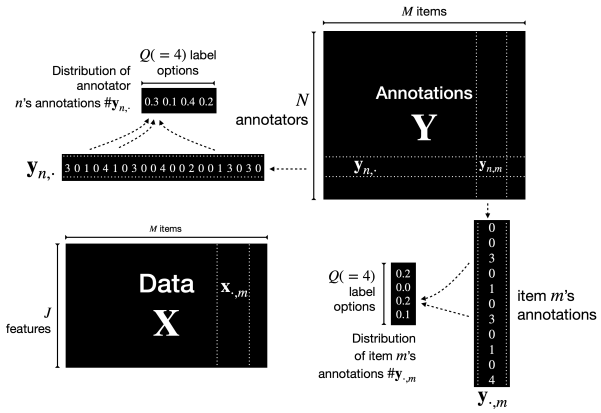

Figure 1: We represent each item $\mathbf{x}_{\cdot,m}$ from each dataset used in this study as a column vector of features. Each data item is associated with a vector of annotations, which, we represent as column of individual annotator responses $\mathbf{y}_{\cdot,m}$ as well as a distribution over the label choices $\#\mathbf{y}_{\cdot,m}$. We are also interested in the distribution of responses that each annotator provides for all the items they annotate $\#\mathbf{y}_{n,\cdot}$.

et al., 2018) views truth in crowdsourcing as a function of the data, response space, and workers who annotate the data. Gordon et al. (2022) study modeling and predicting annotator behavior for specific demographic groups. Their approach is based on a recommender system. Wan et al. (2023) also propsed a model to learn and predict labels using annotator demographics. In contrast to their work, DisCo is able to jointly model and learn from annotator behavior, their annotations, and the content of the data itself.

3 Data

Figure 1 summarizes the notation that we use to describe our data. Let $\mathbf{x} \in \mathcal{R}^{J\times 1}$ be the Jth dimensional (column) feature vector for a particular data item, where $\mathbf{X} \in \mathcal{R}^{J\times M}$ is the design matrix or entire collection of all M data items of a dataset. $\mathbf{Y} \in \{0, 1, \ldots, Q\}^{N\times M}$ is the dataset's annotator response matrix, where each column $\mathbf{y}_{\cdot,m}$ of $\mathbf{Y}$ corresponds to a data item, each row $\mathbf{y}_{n,\cdot}$ an annotator, and each entry $y_{n,m}$ is one label in $\{1, 2, \ldots, Q\}$ or 0, indicating "no response" for that annotator-item pair. Note that, in practice, each item commonly has ≤ 5 labels, so $\boldsymbol{Y}$ is typically a sparse matrix. However, each annotator *could* label the item if asked.

We are interested in *distributions* over annotator *responses*, for any slice of $\mathbf{Y}$, horizontally (denoted $\mathbf{y}_{n,\cdot}$) or vertically (denoted $\mathbf{y}_{\cdot,m}$), as well as the response of an individual annotation to an individual item (denoted $\mathbf{y}_{m,n}$).

We also viewed the responses in each slice as a probability distribution over the space of possible responses. Let $\#$ denote an operator that converts a (horizontal or vertical) slice $\mathbf{y}$ into a vector $\#\mathbf{y} \in [0, 1]^Q, \sum_i \#\mathbf{y}_i = 1$ representing the frequency of each response $\{1, 2, \ldots, Q\}$ as a probability distribution. So, e.g., if there are three responses of "2" out of 10 responses total in $\mathbf{y}_{\cdot,m}$, then $\#(\mathbf{y}_{\cdot,m})_2 = 0.3$.

We conducted our experiments on the few publicly available human annotated datasets with annotator assignments.[2] See Table 1 for a summary of the datasets.

4 DisCo: A Neural Probabilistic Model for Estimating Label Distributions

DisCo (Figure 2) stands for "Distribution from Context," because it takes two inputs, an item $\boldsymbol{x}_{\cdot,m}$ and an annotator $\mathbf{a}_n$, and then learns to jointly predict the annotator's response to the item $\boldsymbol{y}_{n,m}$, the distribution of *all* responses to the item $\#\boldsymbol{y}_{\cdot,m}$, and the distribution of *all* responses the annotator provides (to all items) $\#\boldsymbol{y}_{n,\cdot}$. Additionally, we intend the name to invoke the inclusive, diversity-celebrating spirit of the early disco movement, as preserving annotator diversity is the primary motivation behind the design.

Note that, because the annotators in of our datasets are completely anonymous (except for their set of responses), we represent $\mathbf{a}_n$ as a one-hot vector $0^{n-1}10^{N-n}$. In future work, we hope to have annotator features associated with key features believe to drive disagreement, such as age, race, gender, ethnicity, political affiliation etc.. Also, because we only deal with vertical slices of $\mathbf{X}$, for clarity we denote $\boldsymbol{x}_{\cdot,m}$ as $\boldsymbol{x}_m$. We denote the machine predictions of $\boldsymbol{y}_{n,m}$, $\#\boldsymbol{y}_{\cdot,m}$, and $\#\boldsymbol{y}_{n,\cdot}$ as $\boldsymbol{z}_y$, $\boldsymbol{z}_{yI}$, and $\boldsymbol{z}_{yA}$, respectively.

Although, strictly speaking, only $\boldsymbol{z}_y$ is needed for prediction, $(\boldsymbol{x}_m, \boldsymbol{a}_n)$ represents the intersection of a column and row of the label matrix $\mathbf{Y}$, and $\boldsymbol{z}_{yI}$ or $\boldsymbol{z}_{yA}$ represent the marginal distribution associated with this column or row, respectively. It also provides the same context during training that many of the established item-annotator models rely on (Dawid and Skene, 1979). Moreover, items and annotators tend to cluster in label distribution space (Lakkaraju et al., 2015; Venanzi et al.,

[2]This number is growing; https://pdai.info keeps a running list of available datasets.

Table 1: Summary of the datasets that we conduct our experiments with. The datasets are: GoEmotion ($\mathcal{D}_{\texttt{GE}}$), LabelMe ($\mathcal{D}_{\texttt{LM}}$), Jobs ($\mathcal{D}_{\texttt{JQ1-3}}$), and SBIC Intent ($\mathcal{D}_{\texttt{SI}}$). All of our datasets contain posts that are in English or are based on image data that is already processed ($\mathcal{D}_{\texttt{LM}}$). **[A]** We calculated the mean entropy per data item (respectively, annotator). **[B]** We calculated the entropy of the mean label distribution over all data items (respectively, annotators). Entropy is calculated via the natural logarithm (the units are in nats). See Section "Data" for more details.

Dataset	No. of ants. (per item)	No. of items (per ant.)	Total data items	No. of label choices	Total anno-tators	Avg. entpy. per item [A]	Entropy of items [B]	Avg. entpy. per ants. [A]	Entropy of ants.[B]
$\mathcal{D}_{\texttt{GE}}$	Avg. 4	662	54,263	28	82	0.866	2.925	2.697	2.940
$\mathcal{D}_{\texttt{JQ1}}$	10	2	2000	5	1185	0.746	1.213	0.671	1.213
$\mathcal{D}_{\texttt{JQ2}}$	10	2	2000	5	1185	0.586	1.093	0.708	1.093
$\mathcal{D}_{\texttt{JQ3}}$	10	2	2000	12	1185	0.993	1.888	1.210	1.888
$\mathcal{D}_{\texttt{LM}}$	Avg. 2.5	169	10,000	8	59	0.277	2.050	1.631	2.049
$\mathcal{D}_{\texttt{MR}}$	Avg. 4.96	11	1500	10	137	1.049	2.150	1.324	2.166
$\mathcal{D}_{\texttt{SI}}$	Avg. 3	289	45,318	4	157	0.343	1.280	0.770	1.256

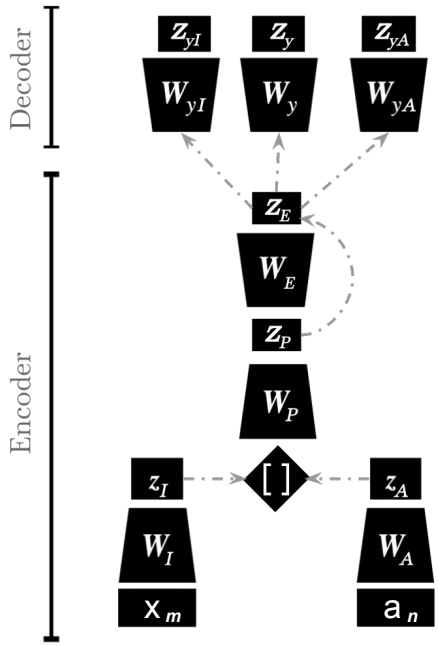

Figure 2: Block diagram showing the main components and parameters of DisCo. The model takes in as input an item $\mathbf{x}_m$ and a one-hot encoding $\mathbf{a}_n$ of an integer identifier n, and is ultimately trained to output a set of three probability distributions, namely, a vector of class probabilities $\mathbf{z}_y$, a distribution of labels from all annotators $\mathbf{z}_{yI}$, and a distribution of labels from all items $\mathbf{z}_{yA}$. Notice that that $\mathbf{x}_m$ and $\mathbf{a}_n$ are first each embedded into their own respective sub-spaces ($\mathbf{z}_I$ and $\mathbf{z}_A$) before they are combined through a vector combination operator (such as concatenation).

2014), and so backpropagating gradients from the Kullback-Leibler (KL) terms placed on these distributions (as we will describe later) acts as a form of regularization tailored to distributional labels. By aggregating labels from related items and annotators, we believe this approach also addresses label sparsity.

In order to facilitate tractable inference and parameter learning, we opted to craft a probabilistic encoder-decoder architecture. DisCo is defined by a set of synaptic weight parameter matrices housed in two constructs $\Theta_e = \{\mathbf{W}_I, \mathbf{W}_A, \mathbf{W}_P, \mathbf{W}_E\}$ and $\Theta_d = \{\mathbf{W}_{yI}, \mathbf{W}_{yA}, \mathbf{W}_y\}$ (bias vectors omitted for clarity), where Θ_e contains the encoder parameters and Θ_d contains the decoder parameters. The model is designed, for each data item feature vector $\mathbf{x}_m$ and annotator $\mathbf{a}_n$ pair, to estimate the values of a set of target label distributions, i.e., the item label distribution $\mathbf{y}_I$ the annotator label distribution $\mathbf{y}_A$, and the (ground-truth) label distribution $\mathbf{y}_{m,n}$.

The output of the encoder is the latent representation of data items and annotators – note that the data item is projected to the space $\mathbf{z}_I$ while the annotator identification integer is embedded into the space $\mathbf{z}_A$. As a result, the encoder, which takes in as input the data item feature vector $\mathbf{x}_m \in \mathcal{R}^{J\times 1}$ (where J is the dimensionality of the item feature space) and the annotator identifier a_n, computes the following output:

$$\mathbf{z}_I = \mathbf{W}_I \cdot \mathbf{x}_m, \;\; \mathbf{z}_A = \mathbf{W}_A \cdot a_n \quad (1)$$

$$\mathbf{z}_P = \phi(\mathbf{W}_P \cdot \phi([\mathbf{z}_I, \mathbf{z}_A])), \text{ and} \quad (2)$$

$$\mathbf{z}_E = \phi\big((\mathbf{W}_E \cdot \mathbf{z}_P) + \mathbf{z}_P\big) \quad (3)$$

where $\cdot$ denotes matrix multiplication and $[\mathbf{a}, \mathbf{b}]$ represents a vector combination operation applied to input vectors $\mathbf{a}$ and $\mathbf{b}$ (such as concatenation or element-wise summation). Notice that a residual connection has been introduced in Equation 3 to improve gradient flow during model training. $\phi(\mathbf{v}) = 1/|1 + \mathbf{v}|$ is the softsign elementwise activation function. $\mathbf{z}_I \in \mathcal{R}^{J_I\times 1}$ and $\mathbf{z}_A \in \mathcal{R}^{J_A\times 1}$ where J_I and J_A are their respective embedding dimensionalities. An additional linear projection is

applied to the combined item and annotator embeddings via matrix $\mathbf{W}_P$ to reduce the dimesionality further to $\mathbf{z}_E \in \mathcal{R}^{J_P \times 1}$ before running the representation through one more non-linear transform (to obtain encoder output $\mathbf{z}_E$).

The decoder, which takes in as input the latent code produced by the encoder $\mathbf{z}_E$, computes its outputs (three different label distribution estimates) as follows:

$$\mathbf{z}_y = \sigma(\mathbf{W}_y \cdot \mathbf{z}_E),\ \mathbf{z}_{yI} = \sigma(\mathbf{W}_{yI} \cdot \mathbf{z}_E),\ \text{and}$$
$$\mathbf{z}_{yA} = \sigma(\mathbf{W}_{yA} \cdot \mathbf{z}_E) \quad (4)$$

where $\sigma(\mathbf{v}) = \exp(\mathbf{v}) / \sum_j \exp(\mathbf{v})[j]$ is the softmax function ($\mathbf{v}[j]$ retrieves the $j$the value/element of the vector $\mathbf{v}$). Note that $\mathbf{z}_y$ is interpreted as $\mathrm{P}(y_{n,m}|\mathbf{x}_m, a_n)$, and can be seen as a Bayesian distribution of annotator n's response to x_m However, $\mathbf{z}_{yI}$ represents $\mathbf{y}_{\cdot,m}$ (the mth column of $\mathbf{Y}$), normalized to sum to one, and $\mathbf{z}_{yA}$ represents $\mathbf{y}_{n,\cdot}$ (also normalized). Thus, these two outputs can be interpreted as frequentist representations of item m's and n's responses, respectively.[3]

4.1 Out-of-Sample Inference

After DisCo has been trained according to the objective function above (Equation 8), it conducts inference over out-of-sample items in a slightly different manner than it does as described in Equations 1-3. Specifically, we present the form that this takes when not only an item $\mathbf{x}_m$ is presented to the model (we later discuss the case when only an item $\mathbf{x}_m$ but also its associated label distribution vector $\mathbf{y}_{\cdot,m}$ are available).

Inference under our proposed model entails using its knowledge of all annotators encountered in the training set to make multiple predictions for a newly encountered data item $\mathbf{x}_m$ and then finally aggregating across this set in order to produce a predicted label or label distribution vector. Concretely, this means that our model will emit N predictions for $\mathbf{x}_m$, i.e., one prediction per annotator embedding stored in its internal memory $\mathbf{W}_A$. Formally, this means that, instead of using Equations 1-3, we conduct inference as follows:

$$\mathbf{z}_I = (\mathbf{W}_I \cdot \mathbf{x}_I) \cdot \mathbf{1}_c,\ \ \mathbf{z}_A = \mathbf{W}_A \quad (5)$$
$$\mathbf{z}_P = \phi(\mathbf{W}_P \cdot \phi([\mathbf{z}_I, \mathbf{z}_A])),\ \text{and} \quad (6)$$
$$\mathbf{z}_E = \phi\big((\mathbf{W}_E \cdot \mathbf{z}_P) + \mathbf{z}_P\big) \quad (7)$$

where $\mathbf{1}_c = \{1\}^{1 \times N}$ is a row vector of ones meant to be multiplied with a column vector to yield a matrix of shape $J \times N$ (meaning the vector result of $(\mathbf{W}_I \cdot \mathbf{x}_m)$ is copied into each column of the output matrix). When using Equation 4 after computing $\mathbf{z}_E$ via Equations 5-7, the resulting outputs $\mathbf{z}_y$, $\mathbf{z}_{yI}$, and $\mathbf{z}_{yA}$ would be matrices with each containing N columns (one distribution vector per annotator). If one desires to use DisCo to produce a final predicted label for item $\mathbf{x}_m$, then $\arg\max$ of each column in $\mathbf{z}_y$ is taken to produce a list of integer labels and the mode is taken over this final set of model-generated class integers. If one desires a single label distribution vector to be produced for item $\mathbf{x}_m$, then the expectation is calculated across columns in $\mathbf{z}_y$.

5 Experimental Setup

Before conducting the research described here, we consulted with our institutional review board(s). They determined it did not constitute human subjects research, primarily because the data was publicly available and secondary. Beyond that, all authors have basic training on conducting human subjects research from CITI.[4] Moreover, we do not reveal any apparent personal identifiers in the data that we use.

In cases when the original data splits are not provided, we use a 50/25/25-percent train/dev/test split. For natural language datasets $\mathcal{D}_{\texttt{GE}}$, $\mathcal{D}_{\texttt{JQ1-3}}$, $\mathcal{D}_{\texttt{MR}}$, and $\mathcal{D}_{\texttt{SI}}$, we used SBERT (Reimers and Gurevych, 2019) with the pretrained `paraphrase-MiniLM-L6-v2` model to generate sentence embeddings as our feature vectors. The model generates embeddings over a 384 dimension space. For $\mathcal{D}_{\texttt{LM}}$, we use features that are distributed with the data set – these are pre-encoded using VGG-16 (Simonyan and Zisserman, 2015).

There are relatively few publicly available datasets that provide label distributions, rather than single labels or label sets. There are even fewer that say which annotators labeled which items (i.e., that provide the annotator label matrix $\mathbf{Y}$). These *annotator assignments* (or *annotator-level* labels (Prabhakaran et al., 2021)) are essential for modeling annotators. Thus, we based our comparison models from those that had been previously tested on data with annotator assignments, even if some of the models in question do not use them. One model

[3] The forms of these KL divergences were derived by exploiting Stirling's approximation (Jaynes, 2003) to deal with the factorial that appears in the definition of the multinomial likelihood. See the supplemental material for details.

[4] https://about.citiprogram.org/en/series/human-subjects-research-hsr/

(CNN) is a baseline that does no pre-processing or modeling of the labels. Another (MM+CNN) is LDL aware, but does not explicitly model annotators. The third (CL) models annotators, but is not explicitly designed for LDL. In the Appendix, we present a short description of our baselines.

5.1 DisCo

Our model is formally described earlier in the paper. The item ($\mathbf{z}_I$) and annotator ($\mathbf{z}_A$) embeddings are combined by setting $[\mathbf{z}_I, \mathbf{z}_A]$ to be vector concatenation. We furthermore regularized model parameters during training by running a Bayesian hyperparameter search (Biewald, 2020) on each dataset across 100 different model parameters. The Adam adaptive learning rate (Kingma and Ba, 2014) was used to optimize parameters by using gradients calculated over mini-batches of 256 samples for 200 epochs. Model parameters that we tuned were: (1) drop-out probabilities, varying from $p = 0$ to $p = 0.99$, to the outputs of $\mathbf{z}_P$ and $\mathbf{z}_E$, (2) random orthogonal matrix (Saxe et al., 2013) versus gaussian versus uniform initialization, (3) the activation function choice between softsign, elu, relu, relu6, and tanh, (4) annotator and item encoder weights varying from 0 to 2, (5) L1 and L2 regularization weights ranging from $1e^{-07}$ to 0.001, and, (6) hidden layer sizes ranging from 64 to 256.

We evaluate these models as a single label learning (SL) problem using accuracy, F1-score, precision, and recall measured over the test set. We further evaluate our models with respect to the label distribution learning (LD) problem using KL-divergence.

6 Results and Discussion

Table 2 presents our main results (additional datasets and results included in the Appendix A.1). Since the main goal of this paper is to learn to predict the distributions of annotator responses, we focus first on KL-divergence. RQ1 asks about the performance of DisCo compared to other LDL approaches that do not model annotators, i.e., CNN, Max Ent, and MM+CNN in our experiments. As per Table 2, we see mixed results, with DisCo performing the best on three and MM+CNN performing best on three. Recall that MM+CNN uses multinomial mixture model clustering and pools together all labels from all items in a given cluster. Compared to DisCo, this tends to result in each item having a much denser set of clusters, and this may explain why it performs so well. We speculate that there is a "sweet spot" between including just enough labels from related items/annotators but not so many that the labels are irrelevant, which varies from dataset to dataset. So when more labels are needed then MM+CNN does best, but when fewer labels are optimal then DisCo wins. Indeed, the datasets on which MM+CNN performs best tend to have more label choices than the datasests where DisCo performs best. It would seem that the more label choices there are, the more labels that one would need to collect in order to get a representative sample of annotator disagreement. In addition, note that DisCo takes about half as much time to train as MM+CNN.

RQ2 asks about the performance of DisCo versus approaches that model annotators, but are not LDL-based, i.e., CL and DS+CNN in our experiments. In contrast to RQ1, DisCo outperforms all of these models across all datasets.

Learning label distributions can result in better single-label learning (Venanzi et al., 2014; Liu et al., 2019a; Weerasooriya et al., 2020). When evaluated as a single label problem, DisCo beats all of the other models in all but one (MM+CNN beats it in terms of accuracy in $\mathcal{D}_{\texttt{GE}}$). Even then, a common dataset between SL and LD measure which bypasses DisCo is $\mathcal{D}_{\texttt{GE}}$. Notably, this is one of the largest datasets with offensive language content, with a large label selection and high number of items per annotator.

To get a sense of the impact of the $\mathbf{W}_y I$ and $\mathbf{W}_y A$ aggregating layers of DisCo, we included results using the model but with those layers removed ($A = I = 0$). The model shows substantial improvement in nearly all tests.

To gain a qualitative sense of our results, we inspected several of the test splits of the more interesting datasets for examples on which DisCo assigned nearly even weights to the two highest-scoring labels. The SBIC Intent ($\mathcal{D}_{\texttt{SI}}$) dataset is one on which DisCo performed the best. It is also the one for which DisCo would be expected to yield the most interesting results, as annotator disagreement could be quite significant. Many of these results were jokes, such as "What do you get when you mix human DNA and goat DNA? Kicked out of the petting zoo." with (Intended, Not-Intended) $= (0.35, 0.65)$ or "why was the lord of the rings trilogy filmed in new zealand ? cause the us were missing the two towers." with (Intended, Not-

Table 2: This table presents experimental results for the classification tasks. **DisCo** is the new method introduced here. **CNN** is a baseline with no label modeling, which receives for each data item the empirical (unprocessed, i.e. f_{dist}) label distributions provided by annotators. **DS+CNN** uses the same CNN model, but the empirical label distributions are replaced with labels from the (Dawid and Skene, 1979) model. **MM+CNN** uses the same CNN model, but the empirical label distributions are replaced with multinomial mixture model centroids. **CL** is the CrowdLayer baseline. We repeated each experiment 100 times and report the mean and standard deviation. **A=I=0** is our DisCo models without contextual layer with annotator and item encoding. The best performing model is indicated in bold. See Section "Experiments" for further details.

Data	CNN	Max Ent	DS + CNN	MM + CNN	CL	$A = I = 0$	DisCo
				Accuracy ↑			
$\mathcal{D}_{\text{GE}}$	0.942 ± 0.001	0.302 ± 0.002	0.168 ± 0.003	**0.946 ± 0.002**	0.359 ± 0.005	0.300 ± 0.001	0.848 ± 0.004
$\mathcal{D}_{\text{JQ1}}$	0.494 ± 0.001	0.621 ± 0.003	0.684 ± 0.004	0.842 ± 0.001	0.813 ± 0.005	0.705 ± 0.003	**0.850 ± 0.005**
$\mathcal{D}_{\text{JQ2}}$	0.475 ± 0.001	0.569 ± 0.004	0.658 ± 0.003	0.810 ± 0.002	0.873 ± 0.003	0.764 ± 0.001	**0.895 ± 0.001**
$\mathcal{D}_{\text{JQ3}}$	0.284 ± 0.020	0.266 ± 0.031	0.061 ± 0.031	0.456 ± 0.010	0.458 ± 0.005	0.656 ± 0.004	**0.702 ± 0.001**
$\mathcal{D}_{\text{LM}}$	0.750 ± 0.045	0.145 ± 0.002	0.201± 0.006	0.798 ± 0.030	0.827 ± 0.002	0.800 ± 0.005	**0.963 ± 0.005**
$\mathcal{D}_{\text{MR}}$	0.177 ± 0.080	0.193 ± 0.003	0.145 ± 0.007	0.824 ± 0.002	0.179 ± 0.003	0.425 ± 0.002	**0.484 ± 0.003**
$\mathcal{D}_{\text{SI}}$	0.759 ± 0.001	0.648 ± 0.004	0.508 ± 0.067	0.658 ± 0.002	0.661 ± 0.003	0.773 ± 0.003	**0.766 ± 0.003**
				F1-Score (Macro) ↑			
$\mathcal{D}_{\text{GE}}$	0.016 ± 0.001	0.017 ± 0.003	0.024 ± 0.004	0.071 ± 0.003	0.042 ± 0.001	0.203 ± 0.001	**0.767 ± 0.005**
$\mathcal{D}_{\text{JQ1}}$	0.168 ± 0.001	0.194 ± 0.005	0.162 ± 0.005	0.388 ± 0.001	0.398 ± 0.002	0.328 ± 0.003	**0.569 ± 0.003**
$\mathcal{D}_{\text{JQ2}}$	0.202 ± 0.001	0.219 ± 0.004	0.197 ± 0.005	0.409 ± 0.001	0.412 ± 0.001	0.400 ± 0.001	**0.587 ± 0.001**
$\mathcal{D}_{\text{JQ3}}$	0.044 ± 0.002	0.083 ± 0.023	0.014 ± 0.021	0.156 ± 0.001	0.164 ± 0.001	0.344 ± 0.001	**0.362 ± 0.001**
$\mathcal{D}_{\text{LM}}$	0.677 ± 0.004	0.097 ± 0.003	0.211 ± 0.007	0.736 ± 0.003	0.821 ± 0.002	0.847 ± 0.003	**0.883 ± 0.007**
$\mathcal{D}_{\text{MR}}$	0.035 ± 0.003	0.100 ± 0.004	0.026 ± 0.008	0.036 ± 0.004	0.087 ± 0.003	0.111 ± 0.005	**0.561 ± 0.004**
$\mathcal{D}_{\text{SI}}$	0.192 ± 0.001	0.356 ± 0.001	0.169 ± 0.070	0.354 ± 0.002	0.359 ± 0.001	0.389 ± 0.002	**0.852 ± 0.002**
				Precision ↑			
$\mathcal{D}_{\text{GE}}$	0.010 ± 0.005	0.012 ± 0.003	0.028 ± 0.006	0.131 ± 0.001	0.330 ± 0.001	0.256 ± 0.001	**0.780 ± 0.003**
$\mathcal{D}_{\text{JQ1}}$	0.145 ± 0.001	0.206 ± 0.002	0.147 ± 0.021	0.392 ± 0.002	0.146 ± 0.003	0.412 ± 0.003	**0.558 ± 0.005**
$\mathcal{D}_{\text{JQ2}}$	0.170 ± 0.001	0.223± 0.023	0.185 ± 0.003	0.388 ± 0.001	0.171 ± 0.002	0.496 ± 0.001	**0.617 ± 0.001**
$\mathcal{D}_{\text{JQ3}}$	0.028 ± 0.002	0.093± 0.040	0.009 ± 0.004	0.133 ± 0.001	0.026 ± 0.001	0.407 ± 0.002	**0.441 ± 0.002**
$\mathcal{D}_{\text{LM}}$	0.580 ± 0.003	0.107± 0.050	0.201 ± 0.005	0.505 ± 0.003	0.836 ± 0.002	0.846 ± 0.004	**0.847 ± 0.004**
$\mathcal{D}_{\text{MR}}$	0.051 ± 0.005	0.129± 0.049	0.015 ± 0.004	0.039 ± 0.002	0.072 ± 0.003	0.116 ± 0.003	**0.643 ± 0.005**
$\mathcal{D}_{\text{SI}}$	0.295 ± 0.001	0.332± 0.001	0.127 ± 0.006	0.328 ± 0.001	0.330 ± 0.002	0.402 ± 0.002	**0.866 ± 0.002**
				Recall ↑			
$\mathcal{D}_{\text{GE}}$	0.035 ± 0.005	0.038 ± 0.040	0.038 ± 0.005	0.059 ± 0.001	0.333 ± 0.001	0.201 ± 0.001	**0.769 ± 0.003**
$\mathcal{D}_{\text{JQ1}}$	0.199 ± 0.001	0.197 ± 0.003	0.206 ± 0.003	0.391 ± 0.002	0.200 ± 0.001	0.312 ± 0.002	**0.558 ± 0.004**
$\mathcal{D}_{\text{JQ2}}$	0.249 ± 0.0001	0.231± 0.030	0.239 ± 0.002	0.431 ± 0.001	0.250 ± 0.005	0.381 ± 0.002	**0.570 ± 0.002**
$\mathcal{D}_{\text{JQ3}}$	0.100 ± 0.002	0.097± 0.048	0.102 ± 0.012	0.205 ± 0.002	0.100 ± 0.0003	0.329 ± 0.001	**0.334 ± 0.001**
$\mathcal{D}_{\text{LM}}$	0.524 ± 0.003	0.101± 0.032	0.322 ± 0.004	0.423 ± 0.005	0.824 ± 0.004	0.892 ± 0.004	**0.856 ± 0.004**
$\mathcal{D}_{\text{MR}}$	0.100 ± 0.005	0.119± 0.048	0.101 ± 0.032	0.119 ± 0.008	0.112 ± 0.004	0.110 ± 0.004	**0.549 ± 0.005**
$\mathcal{D}_{\text{SI}}$	0.259 ± 0.002	0.401± 0.002	0.251 ± 0.004	0.389 ± 0.001	0.390 ± 0.001	0.401 ± 0.001	**0.841 ± 0.001**
				KL-Divergence ↓			
$\mathcal{D}_{\text{GE}}$	2.011 ± 0.001	2.145 ± 0.030	3.247 ± 0.012	**0.707 ± 0.001**	5.838 ± 0.003	4.119 ± 0.002	1.089 ± 0.005
$\mathcal{D}_{\text{JQ1}}$	1.092 ± 0.004	0.648 ± 0.040	1.042 ± 0.005	0.458 ± 0.001	2.077 ± 0.003	0.425 ± 0.002	**0.420 ± 0.005**
$\mathcal{D}_{\text{JQ2}}$	1.088 ± 0.004	0.686± 0.022	1.035 ± 0.003	**0.515 ± 0.001**	1.695 ± 0.003	0.467 ± 0.002	0.575 ± 0.002
$\mathcal{D}_{\text{JQ3}}$	1.462 ± 0.004	1.003± 0.032	3.197 ± 0.034	**0.887 ± 0.001**	3.862 ± 0.001	0.855 ± 0.001	0.900 ± 0.001
$\mathcal{D}_{\text{LM}}$	1.825 ± 0.009	2.692± 0.040	2.201 ± 0.005	1.638 ± 0.008	0.816 ± 0.002	0.497 ± 0.004	**0.209 ± 0.004**
$\mathcal{D}_{\text{MR}}$	1.101± 0.003	1.043± 0.080	1.325 ± 0.007	**0.593 ± 0.004**	3.777 ± 0.001	1.818 ± 0.009	1.389 ± 0.008
$\mathcal{D}_{\text{SI}}$	0.889 ± 0.003	0.782± 0.001	1.514 ± 0.067	0.991 ± 0.003	2.076 ± 0.002	1.498 ± 0.001	**0.554 ± 0.001**

Intended) $= (0.30, 0.70)$, which are clearly offensive to some people, but apparently funny to others. There are also politically charged messages such as "we need to bring back monster trucks, guns, heavy metal 1776 MAGA I want trumps next speech to have monster trucks jumping over an ac/dc concert," with (Intended, Not-Intended) $= (0.35, 0.65)$, or the use of racially derogative terms that may not be universally recognized as such.

Wan et al. (2023) also proposed a model using $\mathcal{D}_{\text{SI}}$ dataset for modeling annotators based on their demographic details. We summarize F1-Scores for all of the baselines and DisCo in Figure 3. The contextual learning ability of DisCo does show a significant improvement over the prior models which do not perform in a similar manner.

On the other hand, items on which the prediction assigned all or nearly all of the probability mass to one label tended to be very obviously racist and/or hateful. In the specific research focus of hate or offensive speech and monitoring in real-world settings involving contentious issues (Palakodety et al., 2020), there is a growing consensus that human-in-the-loop systems aided by automated methods can be more robust in handling controversial edge cases. If our automated method assigns nearly even weights to two (or more) highest scoring labels, perhaps those instances merit greater scrutiny and vetting from multiple web moderators. Since real-world human moderation is costly, our model can potentially serve as a guide in prioritizing human moderation resources.

Our involvement in label distribution learning came from a community-based participatory research group that we belonged to on the use of AI technology in vulnerable communities as a means of preserving, in AI pipelines, minority perspectives that would otherwise be erased when annotator disagreement is resolved (usually in favor of the plurality label, as is common practice today). We believe that these methods, coupled with demographic information on annotators and reliable

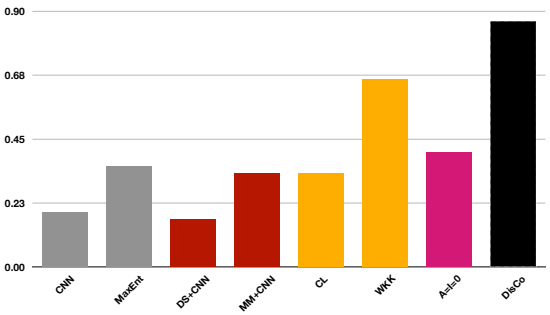

Figure 3: The figure consists a comparison of the models for $\mathcal{D}_{\text{SI}}$ dataset based on F1-Score. Left to right, (grey) **CNN** is a baseline with no label modeling, which receives for each data item the empirical label distributions provided by annotators. (red) **DS+CNN** uses the same CNN model, but the empirical label distributions are replaced with labels from the DS model. **MM+CNN** uses the same CNN model, but the empirical label distributions are replaced with MM centroids. (yellow) **CL** is the CrowdLayer baseline. **WKK** is the baseline from Wan et al. (2023). (pink) **A=I=0** is our DisCo models without output layer with annotator and item embedding. (black) **DisCo** is the new method introduced here.

confidence estimates, can lead to annotated data that is more representative of the true values within a society.

6.1 Future work

Our work currently assumes that each annotator provides at most one label for each item from a fixed set of allowed responses. However, settings in which annotators may provide multiple labels per item, or where the domain of responses is open or highly structured are common, and are often where response diversity can be particularly rich, are rewarding to model. Consider, for instance, machine translation, in which there is clearly no "correct" translation from one language to another. One way to handle multiple responses per item/annotator is to consider each element in the powerset $\mathbf{P}(Q)$ of individual responses Q, so that each subset of Q is treated as an individual response. However, this creates a very large, and usually sparse, response space that is unwieldly. Such simplistic approaches do not address more complex responses such as translations from one language to another.

We hope to fulfill a vision laid out by Lakkaraju et al. (2015), further motivated by Sap et al. (2021), and addressed by Gordon et al. (2022) in which we predict, not just the distribution of responses of the entire population, but that of key vulnerable subgroups. This would allow us to better understand when disagreement is likely to have social or political impacts. Note that if we had demographic information about our annotators, we could infer over any such group by masking out at inference time all annotators who do not not belong to the group of interest.

Given the clustering of responses revealed in Figure 4 and the competitive performance of MM+CNN with respect to KL-divergence, we would like to explore ways to incorporate clustering into the design of DisCo.

7 Conclusion

We proposed a novel neural architecture, DisCo, for modeling the distribution of labels an item receives and the distribution that annotators provide in the presence of item-annotator pairs. Our design was motivated by the desire to break free of the standard assumption (in supervised learning) of single-label ground truth. Experimental results indicate that DisCo performs at a level comparable to state-of-the-art models that were purpose-built for label distribution learning, but with faster training time, and outperforms state-of-the-art annotator-modeling models, even on single-label learning problems. Qualitative inspection of the data shows that the model can predict striking examples of annotator disagreement. Future work will explore ways to more flexibly increase the labels of related items/annotators in order to enhance the sparse label sets.

Limitations

It is highly desirable to test our model on more datasets. However, there are very few multi-class, publicly available datasets that include information about annotator assignments. Often this information is, unfortunately, either discarded or withheld. Without annotator assignments, it is difficult to run experiments related to label distribution learning driven by annotator-item modeling. We hope that this paper encourages more researchers to collect and share more datasets that retain information about annotator-item matchings.

Datasets: We understand that the disagreement between the annotators could arise due to the subjectivity/ambiguity of the content to be annotated, nature of the study, or even worker reliability (Aroyo and Welty, 2013; Inel et al., 2014). These observations cannot be solely utilized to disregard a dataset, since it is not a limitation of the dataset but the nature of the problem domain of annotator

disagreement.

Ethical Considerations

All statistical methods are double-edged swords. Used maliciously, these methods could be used to misrepresent social values and opinions. Moreover, while these methods would be more informative with demographic information on the annotators, this conflicts with the privacy of the annotators, a group of workers who are often treated unfairly (Gray and Suri, 2019).

Acknowledgments

This research was supported by Google Research Award and support through the Google Cloud Research credits. We appreciate the feedback on our work and support for coining the name *DisCo* from Lora Aroyo. The work also utilized resources from Research Computing at the Rochester Institute of Technology (2022). We thank the anonymous reviewers for their helpful feedback and suggestions for our work and the community.

References

Sohail Akhtar, Valerio Basile, and Viviana Patti. 2021. Whose Opinions Matter? Perspective-aware Models to Identify Opinions of Hate Speech Victims in Abusive Language Detection. ArXiv:2106.15896 [cs].

Dina Almanea and Massimo Poesio. 2022. ArMIS - The Arabic Misogyny and Sexism Corpus with Annotator Subjective Disagreements. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2282–2291, Marseille, France. European Language Resources Association.

Lora Aroyo and Chris Welty. 2013. Crowd truth: Harnessing disagreement in crowdsourcing a relation extraction gold standard. *WebSci2013. ACM*, 2013(2013).

Valerio Basile. 2020. It's the end of the gold standard as we know it. On the impact of pre-aggregation on the evaluation of highly subjective tasks. *CEUR Workshop Proceedings*.

Lukas Biewald. 2020. Experiment tracking with weights and biases. Software available from wandb.com.

Reuben Binns, Michael Veale, Max Van Kleek, and Nigel Shadbolt. 2017. Like trainer, like bot? inheritance of bias in algorithmic content moderation. In *International conference on social informatics*, pages 405–415. Springer.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Aida Mostafazadeh Davani, Mark Díaz, and Vinodkumar Prabhakaran. 2021. Dealing with disagreements: Looking beyond the majority vote in subjective annotations. *arXiv preprint arXiv:2110.05719*.

A. P. Dawid and A. M. Skene. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. 28(1):20–28.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. Goemotions: A dataset of fine-grained emotions. *arXiv preprint arXiv:2005.00547*.

Anca Dumitrache, Oana Inel, Lora Aroyo, Benjamin Timmermans, and Chris Welty. 2018. Crowdtruth 2.0: Quality metrics for crowdsourcing with disagreement. *arXiv preprint arXiv:1808.06080*.

Anjalie Field and Yulia Tsvetkov. 2020. Unsupervised discovery of implicit gender bias. *arXiv preprint arXiv:2004.08361*.

Bin-Bin Gao, Chao Xing, Chen-Wei Xie, Jianxin Wu, and Xin Geng. 2017. Deep label distribution learning with label ambiguity. volume 26, pages 2825–2838. IEEE Press.

Xin Geng. 2016. Label Distribution Learning. In *IEEE Transactions on Knowledge and Data Engineering*, volume 28, pages 1734–1748.

Xin Geng and Peng Hou. 2015. Pre-release prediction of crowd opinion on movies by label distribution learning. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-15*.

Xin Geng, Qin Wang, and Yu Xia. 2014. Facial age estimation by adaptive label distribution learning. In *Proceedings - International Conference on Pattern Recognition*.

Mitchell L. Gordon, Michelle S. Lam, Joon Sung Park, Kayur Patel, Jeffrey T. Hancock, Tatsunori Hashimoto, and Michael S. Bernstein. 2022. Jury Learning: Integrating Dissenting Voices into Machine Learning Models. *arXiv:2202.02950 [cs]*.

Mary L. Gray and Siddharth Suri. 2019. *Ghost work: how to stop Silicon Valley from building a new global underclass*. Houghton Mifflin Harcourt, Boston.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

Oana Inel, Khalid Khamkham, Tatiana Cristea, Anca Dumitrache, Arne Rutjes, Jelle van der Ploeg, Lukasz Romaszko, Lora Aroyo, and Robert Jan Sips. 2014. Crowdtruth: Machine-human computation framework for harnessing disagreement in gathering annotated data. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 8797, pages 486–504. Springer International Publishing, Cham. ISSN: 16113349.

Edwin T Jaynes. 2003. *Probability theory: The logic of science*. Cambridge university press.

Brendan Kennedy, Mohammad Atari, Aida Mostafazadeh Davani, Leigh Yeh, Ali Omrani, Yehsong Kim, Kris Coombs, Shreya Havaldar, Gwenyth Portillo-Wightman, Elaine Gonzalez, Joe Hoover, Aida Azatian, Alyzeh Hussain, Austin Lara, Gabriel Cardenas, Adam Omary, Christina Park, Xin Wang, Clarisa Wijaya, Yong Zhang, Beth Meyerowitz, and Morteza Dehghani. 2022. Introducing the Gab Hate Corpus: defining and applying hate-based rhetoric to social media posts at scale. *Language Resources and Evaluation*, 56(1):79–108.

Hyun-Chul Kim and Zoubin Ghahramani. 2012. Bayesian classifier combination. In *Artificial Intelligence and Statistics*, pages 619–627. PMLR.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Manfred Klenner, Anne Göhring, and Michael Amsler. 2020. Harmonization sometimes harms. *CEUR Workshops Proc.*

Himabindu Lakkaraju, Jure Leskovec, Jon Kleinberg, and Sendhil Mullainathan. 2015. A bayesian framework for modeling human evaluations. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 181–189. SIAM.

Elisa Leonardelli, Stefano Menini, Alessio Palmero Aprosio, Marco Guerini, and Sara Tonelli. 2021. Agreeing to Disagree: Annotating Offensive Language Datasets with Annotators' Disagreement. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10528–10539. ArXiv:2109.13563 [cs].

Elisa Leonardelli, Alexandra Uma, Gavin Abercrombie, Dina Almanea, Valerio Basile, Tommaso Fornaciari, Barbara Plank, Verena Rieser, and Massimo Poesio. 2023. SemEval-2023 Task 11: Learning With Disagreements (LeWiDi). ArXiv:2304.14803 [cs].

Miaogen Ling and Xin Geng. 2018. Soft video parsing by label distribution learning. In *Frontiers of Computer Science*, pages 1331–1337.

Tong Liu, Christopher Homan, Cecilia Ovesdotter Alm, Megan Lytle, Ann Marie White, and Henry Kautz. 2016. Understanding discourse on work and job-related well-being in public social media. In *Proceedings of the 54th Annual Meeting of the ACL*.

Tong Liu, Akash Venkatachalam, Pratik Sanjay Bongale, and Christopher M. Homan. 2019a. Learning to Predict Population-Level Label Distributions. In *Seventh AAAI Conference on Human Computation and Crowdsourcing*, volume 7, pages 68–76. A preliminary version appears in (Liu et al., 2019b).

Tong Liu, Akash Venkatachalam, Pratik Sanjay Bongale, and Christopher Homan. 2019b. Learning to predict population-level label distributions. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, pages 1111–1120. ACM.

Curtis G Northcutt, Lu Jiang, and Isaac L Chuang. 2019. Confident learning: Estimating uncertainty in dataset labels. *arXiv preprint arXiv:1911.00068*.

Shriphani Palakodety, Ashiqur R. KhudaBukhsh, and Jaime G. Carbonell. 2020. Hope speech detection: A computational analysis of the voice of peace. In *ECAI 2020 - 24th European Conference on Artificial Intelligence*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 1881–1889. IOS Press.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Vinodkumar Prabhakaran, Aida Mostafazadeh Davani, and Mark Diaz. 2021. On releasing annotator-level labels and information in datasets. In *Proceedings of The Joint 15th Linguistic Annotation Workshop (LAW) and 3rd Designing Meaning Representations (DMR) Workshop*, pages 133–138, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Vikas C Raykar and Shipeng Yu. 2012. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *The Journal of Machine Learning Research*, 13(1):491–518.

Vikas C Raykar, Shipeng Yu, Linda H Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. 2010. Learning from crowds. *Journal of Machine Learning Research*, 11(4).

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Yi Ren and Xin Geng. 2017. Sense beauty by label distribution learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*.

Rochester Institute of Technology. 2022. Research computing services.

Filipe Rodrigues, Mariana Lourenço, Bernardete Ribeiro, and Francisco C. Pereira. 2017. Learning supervised topic models for classification and regression from crowds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2409–2422.

Filipe Rodrigues and Francisco Pereira. 2018. Deep learning from crowds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman. 2008. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision*, 77(1):157–173.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A Smith, and Yejin Choi. 2019. Social bias frames: Reasoning about social and power implications of language. *arXiv preprint arXiv:1911.03891*.

Maarten Sap, Swabha Swayamdipta, Laura Vianna, Xuhui Zhou, Yejin Choi, and Noah A. Smith. 2021. Annotators with attitudes: How annotator beliefs and identities bias toxic language detection.

Andrew M Saxe, James L McClelland, and Surya Ganguli. 2013. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*.

Amirreza Shirani, Franck Dernoncourt, Paul Asente, Nedim Lipka, Seokhwan Kim, Jose Echevarria, and Thamar Solorio. 2019. Learning Emphasis Selection for Written Text in Visual Media from Crowd-Sourced Label Distributions. pages 1167–1172.

Jonathon Shlens. 2014. Notes on kullback-leibler divergence and likelihood theory. *arXiv preprint arXiv:1404.2000*.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–14.

Matteo Venanzi, John Guiver, Gabriella Kazai, Pushmeet Kohli, and Milad Shokouhi. 2014. Community-based bayesian aggregation models for crowdsourcing. In *Proceedings of the 23rd international conference on World wide web*, pages 155–164.

Ruyuan Wan, Jaehyung Kim, and Dongyeop Kang. 2023. Everyone's Voice Matters: Quantifying Annotation Disagreement Using Demographic Information.

Jing Wang and Xin Geng. 2019. Theoretical analysis of label distribution learning. pages 5256–5263.

Tharindu Cyril Weerasooriya, Tong Liu, and Christopher M. Homan. 2020. Neighborhood-based Pooling for Population-level Label Distribution Learning. In *Twenty Fourth European Conference on Artificial Intelligence*.

Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in neural information processing systems*, 22:2035–2043.

Ning Xu, Yun-Peng Liu, and Xin Geng. 2021. Label enhancement for label distribution learning. *IEEE Trans. on Knowl. and Data Eng.*, 33(4):1632–1643.

Xuebing Yang, Yajing Wu, Wensheng Zhang, and Wei Tang. 2020. Label distribution learning with climate probability for ensemble forecasting. *Intelligent Data Analysis*, 24(1):69–82.

Xue Qiang Zeng, Su Fen Chen, Run Xiang, Guo Zheng Li, and Xue Feng Fu. 2020. Incomplete label distribution learning based on supervised neighborhood information. *International Journal of Machine Learning and Cybernetics*, 11(1):111–121.

Heng Ru Zhang, Yu Ting Huang, Yuan Yuan Xu, and Fan Min. 2020. COS-LDL: Label Distribution Learning by Cosine-Based Distance-Mapping Correlation. *IEEE Access*, 8:63961–63970.

Xiang Zheng, Xiuyi Jia, and Weiwei Li. 2018. Label Distribution Learning by Exploiting Label Correlations. In *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, 1, pages 3310–3317.

A Datasets

All the datasets we use for this research are collected by other researchers. We have included information on how they were collected and the platforms utilized in the descriptions.

GoEmotions ($\mathcal{D}_{\mathsf{GE}}$): is an English language dataset of around 58k Reddit comments collected by Demszky et al. (2020)[5]. These comments are annotated by 82 Amazon Mechanical Turkers with 27 emotions, *admiration, amusement, anger, annoyance, approval, caring, confusion, curiosity, desire, disappointment, disapproval, disgust, embarrassment, excitement, fear, gratitude, grief, joy, love,*

[5]Available to download at https://github.com/google-research/google-research/tree/master/goemotions

nervousness, optimism, pride, realization, relief, remorse, sadness, surprise, or *neutral*. The number of annotations per item ranges from one to sixteen.

Job-related ($\mathcal{D}_{\texttt{JQ1}}$, $\mathcal{D}_{\texttt{JQ2}}$, and $\mathcal{D}_{\texttt{JQ3}}$): On a dataset of 2000 tweets, Liu et al. (2016) asked five annotators each from MTurk and FigureEight to label work-related tweets according to three questions with multiple choice responses: point of view of the tweet ($\mathcal{D}_{\texttt{JQ1}}$: *1st person*, *2nd person*, *3rd person*, *unclear*, or *not job related*), subject's employment status ($\mathcal{D}_{\texttt{JQ2}}$: *employed*, *not in labor force*, *not employed*, *unclear*, and *not job-related*), and employment transition event ($\mathcal{D}_{\texttt{JQ3}}$: *getting hired/job seeking*, *getting fired*, *quitting a job*, *losing job some other way*, *getting promoted/raised*, *getting cut in hours*, *complaining about work*, *offering support*, *going to work*, *coming home from work*, *none of the above but job related*, and *not job-related*).

LabelMe ($\mathcal{D}_{\texttt{LM}}$): was originally released as part of a data challenge for computer vision research. The label categories were: *highway, inside city, tall building, street, forest, coast, mountain or open country*. There are a total of $2,688$ images in the dataset, out of which $1,000$ were annotated by an average of 2.547 MTurkers (Rodrigues et al., 2017). The authors use data augmentation to create a larger sample of 10,000 items for training CL (Russell et al., 2008). In order to compare DisCo against this previous benchmark, we ran our experiments on this larger dataset.

Movie Reviews ($\mathcal{D}_{\texttt{MR}}$): Rodrigues and Pereira (2018) culled 1,500 items from a dataset of 5,006 movie reviews in English with a single rating a scale of 1-10 (Pang and Lee, 2005). They asked multiple AMT workers (4.96 per item, on average) provide their own ratings as test data for a Crowd-Layer regression task.

The Social Bias Inference Corpus ($\mathcal{D}_{\texttt{SI}}$): The $\mathcal{D}_{\texttt{SI}}$ dataset contains 45k posts from Reddit, Twitter, and hate sites collected by Sap et al. (2019)[6]. The dataset was annotated with respect to seven questions: offensiveness, intent to offend, lewdness, group implications, targeted group, implied statement, in-group language. Out of these, we consider only the "intent to offend" question, as it had the richest label distribution patterns. It has the label options: *Intended*, *Probably Intended*, *Probably Not Intended*, *Not Intended*. The items in this dataset are in English. The number of annotations per data item varies here between one and twenty annotations per item.

[6]Available to download at https://homes.cs.washington.edu/~msap/social-bias-frames/index.html

Dataset	CNN	MaxEnt	Others	DisCo
		Accuracy ↑		
$\mathcal{D}_{\texttt{HSB}}$	**0.915±0.009**	0.900±0.005		0.895±0.007
$\mathcal{D}_{\texttt{AMS}}$	0.589±0.016	0.611±0.008		**0.787±0.049**
$\mathcal{D}_{\texttt{MDA}}$	0.700±0.008	0.768±0.002		**0.793±0.020**
$\mathcal{D}_{\texttt{GAB}}$	0.901±0.010	**0.914±0.001**		0.903±0.007
		F1-Score (Macro) ↑		
$\mathcal{D}_{\texttt{HSB}}$	0.685±0.019	0.570±0.005	0.91	**0.9681±0.04**
$\mathcal{D}_{\texttt{AMS}}$	0.466±0.081	0.513±0.017	**0.82**	0.780±0.053
$\mathcal{D}_{\texttt{MDA}}$	0.638±0.013	0.724±0.003	0.83	**0.933±0.057**
$\mathcal{D}_{\texttt{GAB}}$	0.478±0.000	0.478±0.000		**0.959±0.601**
		Recall ↑		
$\mathcal{D}_{\texttt{HSB}}$	0.725±0.032	0.572±0.003		**0.960±0.051**
$\mathcal{D}_{\texttt{AMS}}$	0.534±0.030	0.558±0.010		**0.787±0.049**
$\mathcal{D}_{\texttt{MDA}}$	0.635±0.013	0.714±0.004		**0.932±0.058**
$\mathcal{D}_{\texttt{GAB}}$	0.500±0.000	0.500±0.000		**0.953±0.774**
		Precision ↑		
$\mathcal{D}_{\texttt{HSB}}$	0.661±0.018	0.568±0.007		**0.979±0.027**
$\mathcal{D}_{\texttt{AMS}}$	0.526±0.124	0.630±0.018		**0.787±0.048**
$\mathcal{D}_{\texttt{MDA}}$	0.658±0.009	0.743±0.003		**0.936±0.057**
$\mathcal{D}_{\texttt{GAB}}$	0.457±0.000	0.457±0.000		**0.971±0.035**
		KL-Divergence ↓		
$\mathcal{D}_{\texttt{HSB}}$	0.192±0.013	0.192±0.013	0.235	**0.183±0.070**
$\mathcal{D}_{\texttt{AMS}}$	0.448±0.002	**0.445±0.000**	0.469	0.448±0.076
$\mathcal{D}_{\texttt{MDA}}$	0.296±0.006	**0.235±0.001**	0.472	0.401±0.027
$\mathcal{D}_{\texttt{GAB}}$	0.249±0.006	0.252±0.003		**0.207±0.175**

Table 3: Results from running DisCo on additional datases that was introduced through the SemEval contest. We have also included the results from the best model through the SemEval contest in the others column to show how they compare against DisCo.

A.1 Additional Datasets

We conduct our experiments on five additional datasets. The SemEval 2023 task "Learning with Disagreements" (LeWiDi) introduced four datasets to repeat our methods and to compare to a large pool of different models (Leonardelli et al., 2023). And the dataset "Gab" ($\mathcal{D}_{\texttt{GAB}}$) introduced by Kennedy et al. (2022). The results are included in the Table 3.

Arabic Misogyny and Sexism($\mathcal{D}_{\texttt{AMS}}$): The dataset introduced by Almanea and Poesio (2022) is a binary annotation study of 943 Arabic tweets, created to study the effect on sexism judgments of bias. The dataset was annotated by three annotators who self identified themselves as; a conservative male, a moderate female, and a liberal female.

Hate Speech on Brexit ($\mathcal{D}_{\texttt{HSB}}$): The dataset introduced by Akhtar et al. (2021) consists of 1,120 English tweets that are related to immigration and Brexit. The tweets were identified based on keyword filtering. It was annotated by six annotators: a target group of three Muslim immigrants in England and three other annotators for the control group. They annotated looking at hate speech on xenophobia and Islamophobia, aggressiveness, offensiveness, and stereotypes. It was a binary annotation study.

The Multi-Domain Agreement($\mathcal{D}_{\texttt{MDA}}$): The dataset created by Leonardelli et al. (2021) consists of 10,753 English tweets from three domains (Black Lives Matter movement, Election 2020, and COVID-19 pandemic). Each tweet was annotated for offensiveness by five annotators through Amazon Mechanical Turk. The annotator pool consisted of > 800 annotators.

The Gab Dataset ($\mathcal{D}_{\texttt{GAB}}$): This dataset collected from the social network "Gab" introduced by Kennedy et al. (2022) consists of 27,665 posts that are annotated by a minimum of three annotators. The original dataset annotated hate and offensive content. We work with the labels associated for vulgar and/or offensive language classification.

B Experiments

B.1 Computational Setup

Our experiments were conducted on: #1 - A desktop computer with an Intel i6-7600k (4 cores) at 4.20GHz, 32GB RAM, and nVidia GeForce RTX 2070 Super 8GB VRAM, and, #2 - A shared server in our institution with an Intel(R) Xeon E7 v4, 264GB RAM, and GPU Tesla P4 8GB. Our worst case computation was using machine #1 in our setup and with the dataset $\mathcal{D}_{\texttt{GE}}$. The runtime for a single pass of experiments on a single dataset for the CNN took 2 minutes, MM + CNN took 2 hours, CL took 30 minutes, and DisCo took 1 hour. We repeated each experiment 100 times in order to report standard error.

B.2 Approaches that predict label distributions, but do not model annotators

Convolutional Neural Network (CNN) is a baseline supervised model that outputs label distributions with no label modeling. It is a 1D convolutional neural network (Kim, 2014), with three convolution/max pool layers (of dimension 128) followed by a dropout (0.5) and softmax layer, implemented in TensorFlow. It uses KL divergence for the cost function.

Multinomial CNN (MM+CNN) is the best-performing model from Weerasooriya et al. (2020); Liu et al. (2019a). It is an LDL-aware, two-step process that, in order to improve the estimates of each item's given label distribution, applies (as the first step) an unsupervised clustering step to the label distributions before they are passed (as the second step) to an unsupervised learner, which is the same as the CNN model described above. We performed parameter search on the number of item clusters $K \in \{4, \ldots, 40\}$ and report the results on the best performing model. Specifically, Table 4 presents the model selection parameters for the MM model of Liu et al. (2019a). MM + CNN model is clustered only on item classes. K is the number of item classes, L is the number of annotator classes, and KL is the KL divergence when evaluated against empirical ground truth.

Table 4: Experimental parameters of the MM+CNN model.

Data	$\mathcal{D}_{\texttt{GE}}$	$\mathcal{D}_{\texttt{LM}}$	$\mathcal{D}_{\texttt{JQ1}}$	$\mathcal{D}_{\texttt{JQ2}}$	$\mathcal{D}_{\texttt{JQ3}}$	$\mathcal{D}_{\texttt{SI}}$
KL	2.053	0.643	0.193	0.170	0.269	0.942
K	20	13	14	7	35	21

The rationale behind this design (Liu et al., 2019a) is that if a group of data items have similar label distributions, then the annotators believe that this group of items is related and can be clustered together and regarded as having the same distribution, namely the cluster centroid. In this way, the clustering helps with label sparsity. This approach, however, does not model the annotators (nor does it need to be aware of which annotators labeled which items).

Maximum Entropy (Max Ent) is a barebones maximum entropy linear classifier, a single dense layer classification model with a softmax activation. We use this model as an alternative for the **CNN** classification model.

B.3 Approaches that model annotator behavior, but are not designed to predict label distributions

Dawid and Skene (1979) (DS + CNN) uses the label aggregation method introduced by Dawid and Skene (1979) (see Section "Related Work"), paired with the CNN classification baseline model.

CrowdLayer (CL) Rodrigues and Pereira (2018) attach to the output of any neural network with a Q-dimensional output layer (recall that Q is the size of the label space) a *crowd-layer*, which has multiple, parallel, Q-dimensional, new output layers, one for each annotator, and takes as input the old output layer. This extended model is trained as a single, monolithic neural network. It then learns to simultaneously predict the labels of each annotator. The old output layer (now an inner layer) thus becomes a bottleneck through which each of these independent annotator predictions must pass, and the overall model effectively learns a collective

ground truth distribution for the entire population of annotators. During inference the crowd-layer is discarded and the old output layer is used instead. However, during learning the weights from the bottleneck layer to each individual annotator layer learn to discount unreliable annotators and favor reliable ones. This model effectively can learn a label distribution ground truth (that is, there is nothing in their model to bias the bottleneck layer toward a single label output). However, the authors did not anticipate LDL or evaluate its ability to learn label distributions.

Compared to DisCo, CL takes a single data item as input, while DisCo takes an annotator-item pair. CL also has parallel, independent, output dimensions for each annotator, while DisCo has an output layer whose size is independent of the number of annotators and items. Consequently, DisCo takes in more information at input (an item-annotator pair versus just an item) and has to solve a simpler prediction task (namely, to output one label distribution per input versus one label distribution for each annotator per input). We believe that our design offers a more scalable and more tractable learning problem (especially if there are many annotators, as is commonly the case, e.g., when crowdsourcing is used). We also believe DisCo is the superior design for sparse labels, because each input to the model uses all of its layers. By contrast, CL has a large number of parallel layers that are only active when the corresponding annotators are present. So when annotators are sparse, a relatively large number of these annotator layers are not used.

Furthermore, while both our model and CL have a bottleneck layer, the dimension of the bottleneck in the CL model must have the same dimension as the label space (because it is used for inference) while ours can have an arbitrary dimension. This gives our model a bit more flexibility but also requires us to consider this dimension as a hyperparameter that must be tuned. The implementation is based on the code released for the CrowdLayer classification task.

C Derivation of KL-Divergence for DisCo

To train DisCo's parameters $\Theta = \{\Theta_e, \Theta_d\}$, we propose the following multi-objective function:

$$\begin{aligned}&\mathcal{L}(\Theta)\\ &= -\sum_{m,n} \#\mathbf{y}_{n,m} \cdot \log(\mathbf{z}_y) + \sum_m \mathrm{KL}(\#\mathbf{y}_{\cdot,m}||\mathbf{z}_{yI})\\ &\quad + \sum_n \mathrm{KL}(\#\mathbf{y}_{n,\cdot}||\mathbf{z}_{yA}) \qquad (8)\end{aligned}$$

where the first term is the negative categorical log likelihood of the target one-hot encoded label $\mathbf{y}$ and the second and third terms measure the Kullback-Leibler (KL) divergence of between the decoder's estimate and the actual item label distribution $\mathbf{y}_i$ and the actual annotator label distribution $\mathbf{y}_a$, respectively. Specifically, the form of the KL divergence that we use compares two multinomial/multinoulli distributions:

$$\begin{aligned}&\mathrm{KL}(\mathbf{y}_{\cdot,m}||\mathbf{z}_{yI};\Theta)\\ &\quad = \sum_m \#\mathbf{y}_{\cdot,m} \cdot \log \#\mathbf{y}_{\cdot,m}\\ &\quad - \sum_m \#\mathbf{y}_{\cdot,m} \cdot \log \mathbf{z}_{yI} \qquad (9)\\ &\mathrm{KL}(\mathbf{y}_{n,\cdot}||\mathbf{z}_{yA};\Theta)\\ &\quad = \sum_j \#\mathbf{y}_{n,\cdot} \cdot \log \#\mathbf{y}_{n,\cdot}\\ &\quad - \sum_n \#\mathbf{y}_{n,\cdot} \cdot \log \mathbf{z}_{yA}, \qquad (10)\end{aligned}$$

where here and above $\log$ is applied to each scalar value independently and is base e. DisCo's parameters are adjusted to minimize the function defined in Equation 8 by calculating the gradients with respect to both the encoder and decoder weights, i.e., $\frac{\partial \mathcal{L}(\Theta_e,\Theta_d)}{\partial \Theta_e}$ and $\frac{\partial \mathcal{L}(\Theta_e,\Theta_d)}{\partial \Theta_d}$. The resultant partial derivatives are then used to change the current values in Θ_e and Θ_d via stochastic gradient descent or with a more advanced adaptive learning rate rule such as Adam (Kingma and Ba, 2014).

Our use of KL divergence here as a loss function and in our results as an evaluation instrument is particularly relevant to us because it has a very important connection to the likelihood of multinomial samples. Suppose we wished to estimate $\#\mathbf{y}_{\cdot,m}$ by drawing a sample $\#\hat{\mathbf{y}}_{\cdot,m}$ of size S from the distribution defined by $\mathbf{z}_{yI}$. Let $\mathcal{L}(\#\hat{\mathbf{y}}_{\cdot,m}|\mathbf{z}_{yI})$ denote the log-likelihood of this sample. Then (Shlens, 2014),

$$\lim_{N\to\infty} \mathcal{L}(|\#\hat{\mathbf{y}}_{\cdot,m}|\mathbf{z}_{yI})/S = -\mathrm{KL}(\hat{p}|\#\hat{\mathbf{y}}_{\cdot,m}\|\mathbf{z}_{yI}).$$

D Crowd Analysis

We generated t-SNE visualizations on the outputs of **DisCo** models trained on $\mathcal{D}_{\texttt{JQ1}}$, $\mathcal{D}_{\texttt{JQ2}}$, and $\mathcal{D}_{\texttt{JQ3}}$. See Figure 4. The visualization reveal clustering in the output space of these models and is reminiscent of the clustering in the label distribution space that the MM+CNN model is designed to exploit, but which is not explicitly modeled by DisCo.

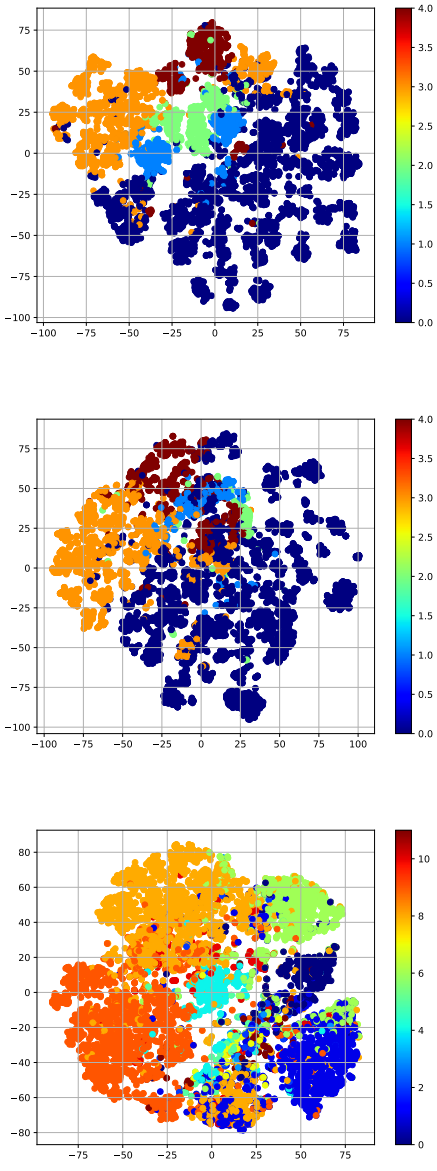

Figure 4: t-SNE plot for training set from DisCo models. Each color represents a label class. (Top) Plot for $\mathcal{D}_{\texttt{JQ1}}$ dataset, which has five label classes (C). (Middle) Plot for $\mathcal{D}_{\texttt{JQ2}}$ dataset, which also has five label classes (C). (Bottom) Plot for $\mathcal{D}_{\texttt{JQ3}}$ dataset, which has 12 label classes (C).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations (after conclusion)

☑ A2. Did you discuss any potential risks of your work?
Ethical Considerations (after conclusion)

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Appendix A

☑ B1. Did you cite the creators of artifacts you used?
Appendix A

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
We used publicly available datasets intended for academic research (Appendix A)

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
We used publicly available datasets intended for academic research (Appendix A)

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Section 6

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Appendix A

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Appendix A and Section 3

C ☑ Did you run computational experiments?

Section 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 5

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 5

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 5

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.