
Dual-Primal Graph VAEs for Noisy Label Aggregation

Patrick Stinson
Zuckerman Institute
Columbia University

Nikolaus Kriegeskorte
Zuckerman Institute
Columbia University

Abstract

Inferring the ground-truth from noisy crowdsourced labels is an important theoretical and practical problem. Neural network-based methods offer an alternative to classical Bayesian models which require specifying a family of generative models used for inference. However, current models either still rely on fairly simple generative models for inference or require pseudo-labels or synthetic data to train the aggregate classifier. We propose a graph VAE architecture in which the decoder and encoder use GAT-based message passing on the adjacency graph of a crowdsourced dataset and its dual, respectively. The ground-truth labels are treated as latent variables, enabling unsupervised representation learning without needing to train a separate classifier. We show our model achieves state of the art performance on crowdsourcing benchmarks. We then demonstrate the generality of our approach by showing how the original crowdsourcing graph can be augmented to incorporate side information such as representations from neural network classifiers trained on the noisy labels to substantially boost their classification performance at test time.

1 Introduction

Aggregating the estimates from imperfect classifiers into an overall “meta-classification” is a topic rooted in both statistical learning theory [43, 7] as well as more colloquial statistics, as illustrated by the “Wisdom of the Crowd” phenomenon famously observed by Galton [8]. Today, crowdsourcing is a common tool used in a range of applications from neural connectome mapping [42] to forecasting world events [49].

If the members of the crowd are elicited multiple times for different estimation tasks, simple estimates from the crowd response, such as the majority vote (MV) can be improved upon by parameterizing individual workers’ response patterns and modifying the overall estimate based on these parameterizations [5]. Increasingly complex hierarchical Bayesian models [18, 25, 48] can be built to model the population response; however, a key limitation is that these generative models must be handcrafted.

Latent variable models parameterized by deep neural networks (DNNs) [20, 38, 33] eliminates the need for handcrafting. However, crowdsourcing methods based on generative modeling [59] have had to use only simple transformations to map from the observed crowd response to latent variables and vice versa.

Crowdsourcing data can be represented as a bipartite graph with edges between worker nodes and task nodes representing noisy labels. Recent work has used representation learning on graphs to parameterize worker-task interactions [26, 57]; however, since they lack explicit generative and inference models, a separate classifier must be learned to map the learned representations to ground-truth label estimates. Since the ground-truth labels are not known, pseudo-labels, proxies to the ground-truth labels such as MV, must be used instead and can be of low quality, giving rise to a sub-par classifier.

We propose overcoming the shortcomings of current latent variable models and representation-based models by integrating representation learning in a graph autoencoder [21] architecture. Specifically,

we use a graph attention network (GAT) [52] to specify a generative model (decoder) for the edges given task latent encodings. The encoder model is a GAT on the dual of the graph in which edge representations are transformed to latent encoding distributions.

In addition, we show how our approach generalizes to integrating disparate sources of information on a graph to do node inference. Specifically, we augment the crowdsourcing graph with a set of nodes corresponding to image representations from DNNs trained on noisy labels and use our method for inference, giving rise to substantially more accurate estimates than those from either source alone.

2 Preliminaries

We are given a set of N labeling tasks, where each task indexed by n has an unobserved ground-truth value $t_n^* \in [L]$. M workers each annotate a subset of these tasks, designated as $\mathcal{N}(m)$ for the m th worker. The full crowdsourcing dataset can be considered as a bipartite graph consisting of a set of worker nodes indexed by $m \in [M]$ and a set of task nodes indexed by $n \in [N]$ with an edge between worker node m and task node n if worker m labeled task n . The value of the edge $E[(m, n)]$ is the (potentially noisy) annotation worker m provided for task n . The goal is to estimate the ground-truth task labels $t^* = \{t_n^*\}_{n=1}^N$ from the worker annotations.

Our strategy is to model the noisy worker annotations as observable projections of the unobserved ground-truth label, which we treat as a latent variable. To model task-worker interactions we use GAT message passing on the crowdsourcing graph to model the observed edges (annotations) given values for the ground-truth labels which we treat as latent variables given by $t = \{t_n\}_{n=1}^N$. Inference of the latent ground-truth labels is done using GAT message passing on the dual of the crowdsourcing graph. The GATs on the dual and primal graphs are used as encoder and decoder models, respectively, to form a VAE.

3 Related Work

3.1 Fully Bayesian approaches

Bayesian classifier combination [18] generalizes the classical approach from Dawid-Skene [5] by assuming a generative model of noisy labeling, which is either parameterized by individual workers' confusion matrices (called iBCC for independent) or parameterized by the joint labeling response (called dBCC for dependent). However, the dependent model is generally intractable for datasets involving more than a few workers.

Enhanced Bayesian Classifier Combination (EBCC) [25] makes modeling worker dependencies tractable through the use of tensor rank decompositions [15] to approximate the joint distribution of the population labeling probabilities.

3.2 Latent encoding-based approaches

Label-aware autoencoders (LAA) [59] are perhaps most conceptually similar to our approach as they also use a VAE [20] architecture where (part of) the latent space represents the ground truth labels. However, the encoder consists of direct transformations from the space of the entire crowd's response (with missing data represented with zeros) to the latent space, and the decoder is another direct transformation from latent space to the population response space. Modeling the transformation directly means that the transformation must be relatively simple (e.g., affine) to ensure the interpretability of the latent space. Such simple transformations may lack the expressivity needed to model the underlying structure of the dataset.

A similar technique is used in [39], where in contrast to crowdsourcing benchmarks, task-level information such as the image data from an image classification task is included. A DNN image classifier is used, but the bottleneck layer represents the logits of the ground-truth labels. Like in LAA, an affine transformation maps the latent space to the population response space.

3.3 Representation learning

GOVERN [26] uses data augmentation to create a contrastive loss function [3, 60] using representations formed by a graph convolutional network (GCN) [13]. However, since the ground-truth labels are not explicitly represented, a separate classifier must be trained using pseudo-labels.

TiReMGE [57] uses a multi-view graph representation [10] of workers and items to minimize the worker-reliability weighted normed difference between workers' labelings and the task representations. However, since the model does not generate edge predictions, there is not a clear way to do cross-validation to select the necessary hyperparameters.

CrowdFM [27] takes a separate approach by generating a large amount of synthetic data and using attention-based message passing to learn representations that are then decoded into ground-truth label estimates. The model is trained through supervised learning. However, such an approach may build in potentially false inductive biases into the model. Additionally, unless one wishes to do inference on a large number of crowdsourcing datasets, training such a model can be prohibitively expensive.

3.4 Training neural networks with noisy labels

When task-level information (e.g., images) is available, DNNs can be trained using noisy labels to make task-specific estimates. Importantly, however, these models are unable to make use of noisy labels at test time.

TAIDTM [24] refines the learned representations of confusion matrices of workers using a GCN in conjunction with a task-dependent DNN representation to model instance-dependent confusion matrices and estimate ground-truth labels.

IDNT [9] learns instance-dependent worker confusion matrices by using a Bayesian generalized linear mixed effects model in which the worker-specific and task ground-truth-specific effects are modeled with DNNs and the model is trained using anchor points [29, 36, 58].

4 Dual-Primate Graph VAEs

To overcome the need to use simple transformations to model the relationship between workers and tasks, we use graph attention networks (GATs) [52], which enable us to learn representations of workers and tasks, which are represented by nodes on the crowdsourcing data graph. Given a representation of any given task and worker, a simple expression parameterizes the interaction between task and worker and predicts the distribution over task labelings for that worker, thus specifying the decoder model, which reconstructs the crowdsourcing data from an input of latent variables representing the tasks, including their ground-truth labels.

While node message passing is done by the decoder to specify a distribution over edges, the encoder infers the distribution over task latent variables from learning representations of the edge data, as in [34], using node message passing from the dual of the crowdsourcing graph. We thus call our approach the Dual-Primal Graph VAE (DPGVAE). Once trained, our model's ground-truth label estimates are read off directly from the task label encoding distributions.

4.1 Generative model

Consider an L -ary crowdsourcing dataset with M workers and N tasks. Given ground-truth task labels $t \in \{1, \dots, L\}^N$ and task-specific latent variables $R \in \mathbb{R}^{N \times D_r}$, the generative model outputs a distribution over edges:

$$E|J \sim P(\cdot; \text{GNN}_\theta(J)). \quad (1)$$

The worker and task embeddings are initialized by $h_m^{(0)} = W_{\text{worker}}^{(\text{embed})} \text{one-hot}(m, M)$ and $j_n^{(0)} = [W_{\text{label}}^{(\text{embed})} \text{one-hot}(t) || R]$. Updates are given by

$$h_m^{(c+1)} = f \left(\lambda_{\text{self}} W_{\text{self}} h_m^{(c)} + \lambda_{\text{neigh}} \sum_{n \in \mathcal{N}(m)} \alpha_{mn} W_{\text{neigh}} j_n^{(c)} \right),$$

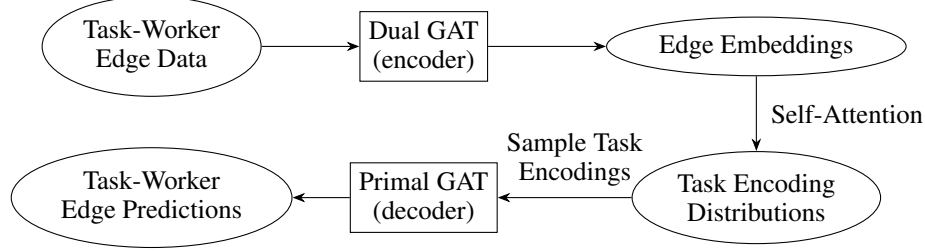


Figure 1: High level description of our proposed method. Crowdsourced data in the form of worker-task edge labels is input to an edge-encoding (dual) GAT. Edge embeddings partitioned by task are then pooled to produce task node encoding distributions from which task encodings are sampled and input to the primal GAT which does node-level message passing and outputs a predictive distribution over each task-worker labeling. Once our autoencoder is trained, the task encoding distributions serve as the ground-truth label estimates.

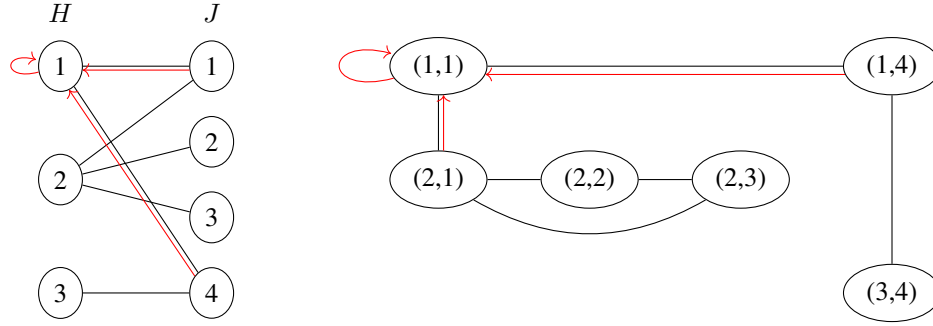


Figure 2: *Left*: Primal representation of crowdsourcing data. Worker nodes and task nodes are labeled above with H and J representing the matrices of worker and task embeddings, respectively. An undirected edge is drawn between worker node m and task node n if worker m labeled task n . Red arrows indicate messages worker node 1 receives (from its neighbors and itself). *Right*: Dual of the crowdsourcing graph. Edges are represented as nodes and nodes are neighbors in the dual graph if the corresponding edges in the primal belong to the same (primal) node. Red arrows indicates messages that edge node $(1, 1)$ receives.

where f is a nonlinearity such as ELU [4] and attention weights are computed by

$$\alpha_{mn} = \frac{\exp(e_{mn})}{\sum_{n \in \mathcal{N}(m)} \exp(e_{mn})},$$

$$e_{mn} = a(h_m^{(c)}, j_n^{(c)}),$$

where a is an attention mechanism. We follow [52] and set

$$a(x, y; W, b) = \text{LeakyReLU}(b^T [Wx || Wy]). \quad (2)$$

Message passing is done over K different attention mechanisms using multi-head attention:

$$h_m^{(c+1)} = \parallel_{k=1}^K f \left(\lambda_{\text{self}}^{(c,k)} W_{\text{self}}^{(c,k)} h_m^{(c)} + \lambda_{\text{neigh}}^{(c,k)} \sum_{n \in \mathcal{N}(m)} \alpha_{mn}^{(c,k)} W_{\text{neigh}}^{(c,k)} j_n^{(c)} \right).$$

Following [52], to learn robust representations we can apply Dropout [46] on the embeddings as well as the adjacency list to learn robust relationships between neighbors.

Updates for task embeddings J are done in an analogous way. After C layers the GAT outputs the final node embeddings which determine the edge predictive distributions by

$$P(E[(m, n)]) = \text{softmax}((h_m^{(C)T} W_l^{(E)} j_n^{(C)})_{l=1}^L), \quad (3)$$

where $\{W_l^{(E)}\}_{l=1}^L$ are learnable weight matrices.

For $C = 0$, we recover a model that resembles a basic form of collaborative filtering: peer and task parameters are learned without paying attention to the graph structure.

4.2 Inference model

Given the edge information E , the inference model outputs a distribution over ground-truth label values and task parameters:

$$J|E \sim P(\cdot; GNN_\phi(E)). \quad (4)$$

The edge embeddings are initialized by $g_{(m,n)}^{(0)} = W_{\text{edge}}^{(\text{embed})} \text{one-hot}(E[(m,n)], L)$.

Each edge embedding $g_{(m,n)}^{(l)}$ has two sets of neighbors: the set of tasks worker m has labeled: $\{g_{m,n'}^{(l)}\}_{n' \neq n}$ and the set of workers rating task n : $\{g_{m',n}^{(l)}\}_{m' \neq m}$. Specifying a separate attentional mechanism for these two potentially different types of relationships, we have the update

$$g_{(m,n)}^{(c+1)} = \frac{1}{K} \left(\lambda_{\text{self}}^{(c,k)} W_{\text{self}}^{(c,k)} g_{(m,n)}^{(c)} + \lambda_{n' \rightarrow n}^{(c,k)} \sum_{n' \in \mathcal{N}(m) \setminus m} \alpha_{nn'}^{(c,k)} W_{\text{task}}^{(c,k)} g_{(m,n')}^{(c)} + \lambda_{m' \rightarrow m}^{(c,k)} \sum_{m' \in \mathcal{N}(n) \setminus n} \alpha_{mm'}^{(c,k)} W_{\text{worker}}^{(c,k)} g_{(m',n)}^{(c)} \right).$$

The distribution over ground-truth label values is given by

$$t_n \sim \text{Cat} \left(\sigma \left(W^{(t)} \left\langle g_{(m,n)}^{(C)} \right\rangle_{m \in \mathcal{N}(n)} \right) \right),$$

where

$$\left\langle g_{(m,n)}^{(C)} \right\rangle_{m \in \mathcal{N}(n)} \triangleq \sum_{m \in \mathcal{N}(n)} \tilde{\alpha}_{nm} g_{(m,n)}^{(C)},$$

and $\tilde{\alpha}_{nm}$ are attention weights determined through (a separately parameterized) self-attention. The distribution over task parameters is given by

$$R_n \sim \mathcal{N} \left(W^{(r,\mu)} \left\langle g_{(m,n)}^{(C)} \right\rangle_{m \in \mathcal{N}(n)}, W^{(r,\log \sigma)} \left\langle g_{(m,n)}^{(C)} \right\rangle_{m \in \mathcal{N}(n)} \right). \quad (5)$$

4.3 Optimization

Following the standard VAE approach [20], we optimize the evidence lower bound (ELBO) given by

$$\mathcal{L}(\theta, \phi) \triangleq \mathbb{E}_{q_\phi(J|E)} [\log p_\theta(E|J) + \log p_\theta(J) - \log q_\phi(J|E)]. \quad (6)$$

The first term measures the quality of the edge reconstructions from the decoder, the second term penalizes the disparity between the task encodings and the prior distribution, and the last term encourages the complexity (entropy) of the task encodings. Generally, the expectation over the encoder's latent distribution is intractable so we optimize over a Monte Carlo estimate:

$$\hat{\mathcal{L}}(\theta, \phi) = \frac{1}{S} \sum_{s=1}^S \log p_\theta(E|J^{(s)}) + \log p_\theta(J^{(s)}) - \log q_\phi(J^{(s)}|E), \quad (7)$$

where $J^{(s)} \sim q_\phi(\cdot|E)$.

As a prior over the ground-truth labels we use a tempered version of MV:

$$P(t_n = l | \{E[(m,n)]\}_{m \in \mathcal{N}(n)}) \propto \left(\frac{1}{\mathcal{N}(n)} \sum_{m \in \mathcal{N}(n)} \mathbb{I}(E[(m,n)] = l) + \delta \right)^\beta,$$

where δ is an offset ensuring that potential ground-truth labelings without any votes retain some prior probability, and β is an inverse-temperature parameter that determines the strength of the MV estimate as the prior. For the non-label task latent variables, we set $p(R_n) = \mathcal{N}(R_n; 0, I_{D_r})$.

Since the samples $J^{(s)}$ in Equation (7) contain discrete valued components, the chain rule cannot be used to estimate the gradient of the parameters controlling the distribution over t in the recognition model. For gradient estimation through discrete-valued sampling we use Reinmax [28], which utilizes Heun’s method [1] and achieves second-order estimation accuracy with just gradient information.

4.3.1 Invariances and subsampling

The GNN architecture makes the VAE invariant to permuting the task, worker, and/or label indices. The GAT architecture enables subsampling neighbors without needing to rescale the subsampled data since the attention weights already normalize the embeddings. To augment each dataset, we resample tasks with replacement N times. To help resampled tasks contain different information, edges are randomly masked with some probability.

4.3.2 Computational Complexity

Memory for using the full graph and its dual is $\mathcal{O}(D_e \max(M \max_m \mathcal{N}(m)^2, N \max_n \mathcal{N}(n)^2))$. From Equation 2, computational complexity scales in the same way. However, it is easy to prune the graph such that $\max_m \mathcal{N}(m)$ and $\max_n \mathcal{N}(n)$ are bounded, thus enabling linear scaling. Dropout also reduces memory costs, and in our experiments we did not need to employ any pruning or Dropout to the crowdsourcing benchmarks in Section 5.1 for the memory request to fit in our GPU. For our experiments augmenting crowdsourcing graphs with 10,000 tasks in Section 5.2, a Dropout rate of 0.9 on the task-linked neighbors of the dual was sufficient to cut down enough on memory requirements.

Recent work [16] has established theoretical guarantees on GNNs generalizing over subsamples of graphs. Using these techniques to scale our approach to unsupervised learning on larger graphs that can occur in other domains is left for future work.

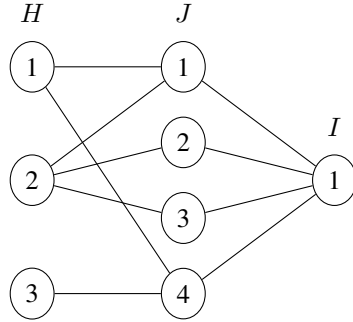


Figure 3: Crowdsourcing graph in Figure 2 augmented with another source of information, e.g., a DNN image classifier’s bottleneck layer representation of each image.

4.4 Posterior Collapse and Disentanglement

In infamous issue with VAEs is posterior collapse [30, 11, 54], in which the latent encodings are ignored by the decoder model, which is powerful enough to still reconstruct the data. In our model, posterior collapse would mean that the task label encodings would revert to their MV prior and the model’s estimate would be identical to MV. From Equation 3, we can put a simple restriction on the embedding dimensionality of the decoder to ensure the decoder cannot reconstruct the input data with arbitrary precision without paying attention to the latent encodings.

Proposition 1. *The conditional likelihood of the decoder in Eqn. 3 cannot be maximized under an arbitrary input $J = [T||R]$ for $D_e < L \max_n |\mathcal{N}(n)|$.*

We prove Proposition 1 in Appendix A.1. Proposition 1 gives a condition under which our model is not susceptible to posterior collapse due to an overexpressive decoder.

However, in practice, we do not find this worst case condition to be a failure mode or even a practical factor in our model. In Figure 4, we see that setting a fixed embedding dimensionality to 100 does not greatly reduce the performance of the model, despite being substantially greater than $\min_{\mathcal{D}} L_{\mathcal{D}} \max_n |\mathcal{N}_{\mathcal{D}}(n)|$, where $\mathcal{D}$ indexes the datasets used in our experiments (though we see that having the encoding dimension as a hyperparameter to select for in cross-validation improves performance, as expected). The reason behind this is that the structure in the data is reflected in the GAT architecture, and so the final worker and tasks embeddings are not treated simply as free parameters, which may be the case in models that are susceptible to overparameterization such as LAA.

A related phenomenon is entanglement of the latent variables [14, 2, 32]. If the label and task-specific latents both partially encode the ground-truth labels, our model’s estimate, which is given by the label encoding latents, may be inaccurate. In Figure A1, we show traces of Monte Carlo estimated mutual information between t and r as a function of training epoch. For only a couple of datasets does the estimated mutual information increase beyond negligible levels, and in those datasets the mutual information eventually reduces back to negligible levels over the course of training. To see if these transient entanglements inhibited the final encodings, we tried training our model with a penalty on the sample mutual information but found that the resulting estimates were slightly worse.

One possible explanation for the phenomenon of automatically disentangled encodings is due to the heterogeneous structure of the joint latent space, where t_n is a categorical random variable whose probability distribution consists of L point masses, and R_n is a Gaussian distribution. Studies demonstrating entanglement in VAEs generally use a joint Gaussian encoding space. It has also been shown that vector-quantized latent encodings can rescue VAEs from posterior collapse [51, 37].

4.5 Integrating Disparate Sources of Information

In addition to doing inference on standard bipartite crowdsourcing graphs, our method generalizes to a large family of graphs in which additional multiple sources of information can be easily included. For instance, in Section 5.2, we first train neural networks using noisy data and use the learned task-specific representations as an extra set of edges connecting to the task nodes, shown in Figure 3. Our method thus generalizes beyond the standard crowdsourcing context in which the data that can be utilized for inference is restricted to a bipartite worker-task graph.

5 Experiments

After each training epoch, we calculated the reconstruction loss on a held out part of the data (10% of the dataset) as a validation set. After the specified number of training epochs, the model with the lowest validation reconstruction loss was used as the final model. The argmax of the task label latent encoding distributions resulting from performing inference on the full dataset were used as the final ground-truth label estimates. Accuracies are shown as mean $\pm$ standard deviation over five random seeds, using random tie-breaks over MV estimates.

For all experiments, the MV prior tempering hyperparameters δ and β were set to be .01 and 1.0, respectively, and we used $C = 2$ layers in our GATs and set $D_r = 1$ in all non-ablation experiments. We used Adam [19] with a cosine-annealed learning rate starting at 5×10^{-4} . In the crowdsourcing benchmark experiments in Section 5.1, we performed cross-validation over hyperparameters: embedding Dropout {0, .2, .5}, neighbor Dropout: {0, .2, .5}, embedding dimensionality {20, 50, 100}, number of attention heads {5, 10} number of epochs {1000, 5000}. For our experiments in Section 5.2, we did not do cross-validation over the hyperparameters and instead fixed the embedding Dropout to .5, the primal GAT dropout to .5, the dual GAT dropout to .9, the embedding dimensionality to 30, the number of attention heads to 5, and the number of training epochs to 5000.

5.1 Crowdsourcing benchmarks

We first compared our proposed method with competitors on crowdsourcing benchmarks. Datasets include CF [17]: sentiment classification on tweets about the weather, face [35]: facial expression classification, dog [56]: labeling of different dog breeds, zhang14spectral tweet sentiment classification for a company, prod [53]: determining whether two descriptions are for the same product, adult [31]: age-restriction appropriateness of a website given its url, RTE [45]: textual entailment classification, LabelMe [41] image scene classification. LAA-B, LAA-O, and LAA-L refer to variants of LAA detailed in [59] that correspond to models with $D_r = 0, 1, 2$, respectively.

In Table 1, we show our model performs competitively to superior to the competing models. Overall, we find its ranking average over all datasets to be the best.

Table 1: Accuracy of DPGVAE and competing methods on crowdsourcing datasets.

	CF	face	dog	senti	prod	adult	RTE	LabelMe	avg rank
MV	.880 $\pm$.006	.637 $\pm$.004	.822 $\pm$.004	.933 $\pm$.002	.897 $\pm$.000	.757 $\pm$.004	.899 $\pm$.003	.767 $\pm$.005	5.4
iBCC [18]	.883 $\pm$.000	.641 $\pm$.003	.831 $\pm$.001	.960 $\pm$.001	.938 $\pm$.000	.745 $\pm$.000	.921 $\pm$.001	.764 $\pm$.000	4.3
EBCC [25]	.883 $\pm$.000	.638 $\pm$.004	.840 $\pm$.000	.961 $\pm$.000	.935 $\pm$.000	.748 $\pm$.000	.931 $\pm$.000	.786 $\pm$.000	2.9
GOVERN [26]	.860 $\pm$.054	.653 $\pm$.006	.822 $\pm$.001	.957 $\pm$.000	.771 $\pm$.106	.610 $\pm$.094	.931 $\pm$.000	.775 $\pm$.003	4.9
LAA-B [59]	.849 $\pm$.002	.618 $\pm$.006	.836 $\pm$.002	.948 $\pm$.003	.896 $\pm$.000	.756 $\pm$.006	.882 $\pm$.005	.758 $\pm$.001	6.4
LAA-O [59]	.811 $\pm$.007	.648 $\pm$.003	.810 $\pm$.003	.930 $\pm$.006	.862 $\pm$.001	.746 $\pm$.001	.881 $\pm$.008	.766 $\pm$.001	7.0
LAA-L [59]	.837 $\pm$.012	.646 $\pm$.001	.807 $\pm$.006	.923 $\pm$.005	.896 $\pm$.000	.757 $\pm$.002	.872 $\pm$.007	.766 $\pm$.000	6.5
CrowdFM [27]	.881 $\pm$.000	.641 $\pm$.000	.823 $\pm$.000	.882 $\pm$.000	.899 $\pm$.000	.762 $\pm$.000	.915 $\pm$.000	.771 $\pm$.000	4.6
DPGVAE	.898 $\pm$.002	.658 $\pm$.003	.842 $\pm$.002	.955 $\pm$.003	.928 $\pm$.004	.757 $\pm$.003	.925 $\pm$.002	.789 $\pm$.003	2.0

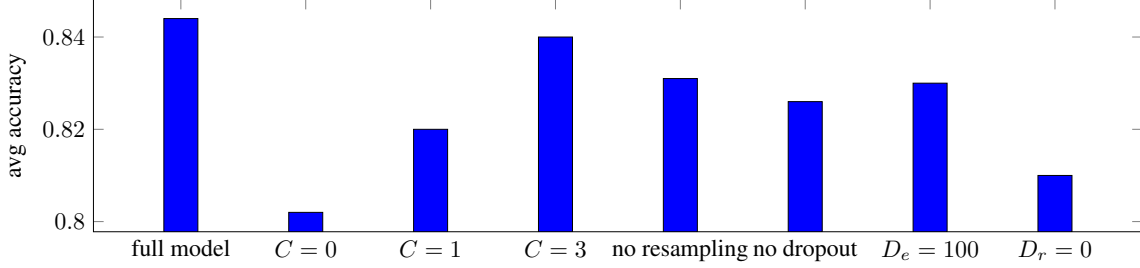


Figure 4: Performance of model ablations on the average performance accuracy on crowdsourcing benchmarks in Table 1. C represents the number of layers in the encoder and decoder GATs, D_e is the fixed embedding dimensionality using in the GATs, and D_r is the encoding dimensionality of the (non-label) task latent variables.

5.1.1 Ablations

We then performed ablations on our model to gain further insights into which parts of our model were important to its success. We show the accuracy of the ablated models averaged over all datasets in Table 1 in Figure 4.

Our ablations show that while cross-validation helps (as expected given the heterogeneity of datasets), there is no catastrophic loss if a given element is fixed. The most notable decrease in performance comes from $C = 0$, which corresponds to not using the GAT architecture to encode or decode the latent variables and instead use affine transformations to map from latent space to embedding space in the encoder and vice versa in the decoder. There is little to no performance drop increasing from $C = 2$ layers to $C = 3$, demonstrating again that the GAT architecture is robust to overparameterization in this domain.

Resampling and Dropout are shown to be beneficial to performance, but removing them does not incur catastrophic loss, showing the robustness of our method. Fixing the embedding dimensionality to the highest among the hyperparameters, $D_e = 100$ also incurs only a slight drop in performance, suggesting that it is fairly difficult to overparameterize the model when the encoding and decoding are done by GATs (additionally, resampling and Dropout can help regularize the model).

Setting $D_r = 0$ decreases the performance of the model more substantially. Without this extra latent encoding information, it may be difficult for the model to distinguish among tasks with the same inferred ground-truth label. For example, in a dataset in which each worker labels each task, embeddings among tasks with the same task label latent variable will be identical.

5.2 Training DNNs with noisy labels and augmenting the crowdsourcing graph with their representations

We then wanted to see if our model could do a more generic form of crowdsourcing by integrating labeling information from workers with representations from neural networks, illustrated in Figure 3, as DNNs themselves cannot make use of noisy labels at test time, nor can previously proposed crowdsourcing methods make use of neural network representations.

Table 2: Classification accuracy using noisy labels: accuracies of baseline methods compared with their augmentations in GraphVAE.

label noise	MNIST			CIFAR10			CIFAR10-N	LabelMe
	low	mid	high	low	mid	high		
MV	.924 $\pm$.001	.766 $\pm$.001	.596 $\pm$.002	.933 $\pm$.001	.771 $\pm$.000	.605 $\pm$.002	.905 $\pm$.003	.767 $\pm$.005
DPGVAE	.927 $\pm$.002	.776 $\pm$.003	.614 $\pm$.004	.939 $\pm$.002	.797 $\pm$.002	.644 $\pm$.003	.912 $\pm$.004	.789 $\pm$.003
MV + $\hat{y}_{\text{IDNT}}$	.967 $\pm$.002	.868 $\pm$.001	.751 $\pm$.002	.970 $\pm$.001	.869 $\pm$.001	.743 $\pm$.000	.942 $\pm$.001	.820 $\pm$.002
DPGVAE + $\hat{y}_{\text{IDNT}}$	.972 $\pm$.002	.894 $\pm$.002	.785 $\pm$.003	.972 $\pm$.001	.885 $\pm$.002	.783 $\pm$.002	.946 $\pm$.003	.851 $\pm$.002
IDNT [9]	.990 $\pm$.001	.988 $\pm$.001	.979 $\pm$.002	.880 $\pm$.001	.874 $\pm$.001	.870 $\pm$.001	.898 $\pm$.003	.850 $\pm$.010
DPGVAE + $\hat{x}_{\text{IDNT}}$	.995 $\pm$.001	.993 $\pm$.003	.987 $\pm$.002	.962 $\pm$.002	.927 $\pm$.002	.881 $\pm$.003	.948 $\pm$.003	.872 $\pm$.013
MV + $\hat{y}_{\text{TAIDTM}}$	.967 $\pm$.000	.873 $\pm$.001	.739 $\pm$.001	.968 $\pm$.001	.874 $\pm$.000	.748 $\pm$.000	.820 $\pm$.001	.814 $\pm$.002
DPGVAE + $\hat{y}_{\text{TAIDTM}}$	.971 $\pm$.001	.896 $\pm$.002	.773 $\pm$.002	.971 $\pm$.002	.893 $\pm$.001	.790 $\pm$.001	.831 $\pm$.002	.822 $\pm$.004
TAIDTM [24]	.993 $\pm$.001	.993 $\pm$.002	.962 $\pm$.003	.912 $\pm$.002	.909 $\pm$.002	.899 $\pm$.001	.900 $\pm$.004	.854 $\pm$.011
DPGVAE + $\hat{x}_{\text{TAIDTM}}$	.998 $\pm$.001	.995 $\pm$.002	.970 $\pm$.002	.976 $\pm$.002	.946 $\pm$.003	.910 $\pm$.002	.943 $\pm$.003	.874 $\pm$.010

5.2.1 Simulated labelings

We first generated simulated labelings of three different noise levels: low, mid, and high with an average labeling error rate of 20%, 35%, and 50%, respectively, for MNIST [6] and CIFAR10 [22] following the procedure outlined in [9]. Each task received 5 labelings out of the pool of 10 simulated workers. We then trained IDNT [9] and TAIDTM [24] classifiers from the noisy labels using their official implementations. For MNIST, we used Lenet-5 [23] and ResNet-18 [12] for CIFAR10.

Once the models are trained, we take the representations before the last fully connected and softmax layers (with dimensionalities of 84 and 512 for Lenet-5 and ResNet-18, respectively) and use them as the initial embeddings for the edges between the task nodes and the I node in Figure 3. We otherwise train our model in the same way using a dual GAT as an encoder and a primal GAT as the decoder with the primal graph as represented in Figure 3.

In Table 2, we show that by using the augmented graph and combining the information in the noisy labelings with the image information encoded by the DNNs (referred to as $\hat{x}$), we achieve a much higher performance over just using the trained DNN (IDNT or TAIDTM in Table 2) or just using our method with the noisy labelings (DPGVAE in Table 2). As a baseline, we also compare the performance gain by simply including the DNN’s label estimate ($\hat{y}$ in Table 2) and find that this alone does not account for the substantial performance boost.

5.2.2 Real world datasets

We used the same training and evaluation procedure for two datasets with real world labelings. CIFAR10-N [55] consists of three worker labelings for each of the 50000 images in CIFAR10’s training set. LabelMe [40, 50] consists of 1000 $256 \times 265 \times 3$ images labeled by an average of 2.5 workers per image. We randomly split each dataset into an 80/10/10 train/validation/test partition. For CIFAR10-N, the architecture was ResNet-18, and for LabelMe, as in [9], each image was first passed through a pretrained VGG-16 [44] network followed by two trainable fully-connected layers (the first with 50% Dropout) followed by a softmax classification layer. For the VGG-16 network, the representation given to DPGVAE was 128 dimensional.

In Table 2 we see the same qualitative trends for the real world noisily labeled datasets as the simulated datasets, where DPGVAE done over the crowdsourcing graph augmented the DNN representations performs substantially better than the neural network models on their own or models using the crowdsourced labels on their own, with the more accurate DNN estimates $\hat{y}$ not sufficient to account for the performance boost.

6 Discussion

In the first half of this paper, we have proposed a crowdsourcing model that combines unsupervised learning and representation learning to overcome shortcomings in previous crowdsourcing models using either only unsupervised learning or representation learning. In the second half of this paper, we have shown how our proposed model can naturally generalize to incorporate disparate sources of information relevant for noisy label aggregation. Namely, we have shown how our model can

integrate both noisy labels and DNN image representations in order to make substantially more accurate predictions than when just using one source of information, demonstrating a form of human-machine complementarity [47].

References

- [1] U. Ascher and L. Petzold. *Computer methods for ordinary differential equations and differential-algebraic equations*. SIAM: Society for Industrial and Applied Mathematics, 1998.
- [2] Christopher P. Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchne. Understanding disentangling in beta-VAE. *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.
- [3] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. *International conference on machine learning (ICML)*, 2020.
- [4] D. Clever, T. Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *International Conference on Learning Representations (ICLR)*, 2016.
- [5] A Dawid and A Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics*, 1979.
- [6] L Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 2012.
- [7] Y Freund and R Schapire. Experiments with a new boosting algorithm. *International Conference on Machine Learning*, 1996.
- [8] Francis Galton. Vox populi. *Nature*, 75, 1907.
- [9] Hui Guo, Boyu Wang, and Grace Yi. Label correction of crowdsourced noisy annotations with an instance-dependent noise transition model. *Advances in Neural Information Processing Systems*, 2023.
- [10] Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive Multi-View Representation Learning on Graphs. *International Conference on Machine Learning (ICML)*, 2020.
- [11] Junxian He, Daniel Spokoyny, Graham Neubig, and Taylor Berg-Kirkpatrick. Lagging Inference Networks and Posterior Collapse in Variational Autoencoders. *ICLR*, 2019.
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
- [13] Mikael Henaff, Joan Bruna, and Yann LeCun. Deep convolutional networks on graph-structured data. *arXiv*, 2015. URL <https://arxiv.org/abs/1506.05163>.
- [14] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. *ICLR*, 2017.
- [15] F Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 1927.
- [16] M. Jain, S. Jegelka, I. Karmarkar, and L. Ruiz. Subsampling graphs with GNN performance guarantees. *arXiv*, 2025. URL <https://arxiv.org/pdf/2502.16703v1>.
- [17] T. Josephy, M. Lease, P. Paritosh, M. Krause, M. Georgescu, M. Tjälve, and D. Braga. Workshops held at the first AAAI conference on human computation and crowdsourcing. *AI Magazine*, 2014.
- [18] H.-C. Kim and Z. Ghahramani. Bayesian classifier combination. *International Conference on Artificial Intelligence and Statistics*, pages 619–627, 2012.
- [19] D. Kingma and J. Ba. Adam: a method for stochastic optimization. *ICLR*, 2015.
- [20] Diederich P Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv*, 2013. URL <https://arxiv.org/abs/1312.6114>.

- [21] T Kipf and M Welling. Variational graph auto-encoders. *arXiv*, 2016. URL <https://arxiv.org/abs/1611.07308>.
- [22] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto, 2009.
- [23] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.
- [24] Shikun Li, Xiaobo Xia, Jiankang Deng, Shiming Ge, and Tongliang Liu. Transferring annotator- and instance-dependent transition matrix for learning from crowds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [25] Yuan Li, Benjamin I. P. Rubinstein, and Trevor Cohn. Exploiting worker correlation for label aggregation in crowdsourcing. *International Conference on Machine Learning*, 2019.
- [26] Hao Liu, Jiacheng Liu, Feilong Tang, Peng Li, Long Chen, and Jiadi Yu. Graph contrastive learning for truth inference. *IEEE International Conference on Data Engineering*, 2024.
- [27] Hao Liu, Jiacheng Liu, Feilong Tang, Long Chen, Jiadi Yu, Yanmin Zhu, Qiwen Dong, Yichuan Yu, and Xiaofeng Hou. Towards a Foundation Model for Crowdsourced Label Aggregation. *International Conference on Learning Representations (ICLR)*, 2026.
- [28] Liyuan Liu, Chengyu Dong, Xiaodong Liu, Bin Yu, and Jianfeng Gao. Bridging discrete and backpropagation: Straight-through and beyond. *37th Conference on Neural Information Processing Systems (NeurIPS)*, 2023.
- [29] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.
- [30] James Lucas, George Tucker, Roger Grosse, and Mohammad Norouzi. Don’t Blame the ELBO! A Linear VAE Perspective on Posterior Collapse. *33rd Conference on Neural Information Processing Systems (NeurIPS)*, 2019.
- [31] W Mason and S Suri. Conducting behavioral research on Smazon’s Mechanical Yurk. *Behavior Research Methods*, 2016.
- [32] Emile Mathieu, Tom Rainforth, N Siddharth, and Yee Whye Teh. Disentangling Disentanglement in Variational Autoencoders. *International Conference on Machine Learning ICML*, 2019.
- [33] A Mnih and K Gregor. Neural variational inference and learning in belief networks. *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.
- [34] O. Monti, A. Shchur, A. Bojchevski, O. Litany, S. Sunnemann, and M. M. Bronstein. Dual-primal graph convolutional networks,. *European Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2018.
- [35] B Mozafari, P Sarkar, M Franklin, M Jordan, and S Madden. Scaling up crowd-sourcing to very large datasets: A case for active learning. *Proceedings of the VLDB Endowment*, 8(2):125–136, 2014.
- [36] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
- [37] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating Diverse High-Fidelity Images with VQ-VAE-2. *33rd Conference on Neural Information Processing Systems (NeurIPS)*, 2019.
- [38] D. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.
- [39] F Rodrigues, F Pereira, and B Ribeiro. Learning from multiple annotators: distinguishing good from random labelers. *Pattern Recognition Letters*, 34(12):1428–1436, 2013.

- [40] Filipe Rodrigues, Mariana Lourenco, Bernardete Ribeiro, and Francisco C Pereira. Learning supervised topic models for classification and regression from crowds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [41] R Rodrigues and F Pereira. Deep learning from crowds. *AAAI*, 2018.
- [42] Jinseop S, Matthew J. Greene Kim, Aleksandar Zlateski, Kisuk Lee, Mark Richardson, Srinivas C. Turaga, Michael Purcaro, Matthew Balkam, Amy Robinson, Bardia F. Behabadi, Michael Campos, Winfried Denk, and H. Sebastian Seung. Space–time wiring specificity supports direction selectivity in the retina. *Nature*, 2014.
- [43] R Schapire. The strength of weak learnability. *Machine Learning*, 1990.
- [44] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations (ICLR)*, 2015.
- [45] R Snow, B O’Connor, and A Ng. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2008.
- [46] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 2014.
- [47] Mark Steyvers, Heliodoro Tejeda, Gavin Kerrigan, and Padhraic Smyth. Bayesian modeling of human–AI complementarity. *PNAS*, 2022.
- [48] P. Stinson and N. Kriegeskorte. A nonparametric model for item-dependent Bayesian classifier combination. *Uncertainty in Artificial Intelligence (UAI)*, 2025.
- [49] P Tetlock. *Expert Political Judgment: How Good Is It? How Can We Know?* Princeton University Press, 2017.
- [50] Antonio Torralba, Bryan C Russell, and Jenny Yuen. Labelme: Online image annotation and applications. *Proceedings of the IEEE*, 2010.
- [51] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *31st Conference on Neural Information Processing Systems (NIPS)*, 2017.
- [52] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *ICLR*, 2018.
- [53] J Wang, T Kraska, M Franklin, and J Feng. Crowder: Crowdsourcing entity resolution. *Proceedings of the VLDB Endowment*, 2012.
- [54] Yixin Wang, David M. Blei, and John P. Cunningham. Posterior Collapse and Latent Variable Non-identifiability. *Advances in Neural Information Processing Systems 34 (NeurIPS)*, 2021.
- [55] Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. *International Conference on Learning Representations (ICLR)*, 2022.
- [56] P Welinder, S Branson, P Perona, and S Belongie. The multidimensional wisdom of crowds. *Advances in Neural Information Processing Systems*, pages 2424–2432, 2010.
- [57] Gongqing Wu, XinGui Zhuo, Xianyu Bao, Xuegang Hu, Richang Hong, and Xindong Wu. Crowdsourcing truth inference via reliability-driven multi-view graph embedding. *ACM Transactions on Knowledge Discovery from Data*, 17(5), 2023.
- [58] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. Are anchor points really indispensable in label-noise learning? *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
- [59] Li’ang Yin, Jianhua Han, Weinan Zhang, and Yong Yu. Aggregating crowd wisdoms with label-aware autoencoders. *IJCAI*, 2017.
- [60] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, and Y. Shen. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

A Appendix

A.1 Proof of Proposition 1

Recall **Proposition 1**: The conditional likelihood of the decoder in Eqn. 3 cannot be maximized under an arbitrary input $J = [T||R]$ if $D_e < L \max_n |\mathcal{N}(n)|$.

Proof. For a set of labeled edges $\{E[(m, n)]\}$ the conditional likelihood is maximized when for a set of latent ground-truth labels $\{t_n\}_n$ the following conditions are met:

$$\left. \begin{array}{l} h_m^T W_l j_n > 0, \quad l = t_n \\ h_m^T W_{l'} j_n < 0, \quad l' \neq l \end{array} \right\} \forall m \in \mathcal{N}(n) \Bigg\} n = 1, \dots, N.$$

For $n^* = \operatorname{argmax}_n |\mathcal{N}(n)|$, we have a system of $L|\mathcal{N}(n^*)|$ strict inequalities linear in each of the D_e elements of j_n . Almost surely wrt the distribution of h and W induced by random initialization (and any previous training steps), these constraints cannot be satisfied for any choice of t_n for $D_e < L|\mathcal{N}(n^*)|$. $\square$

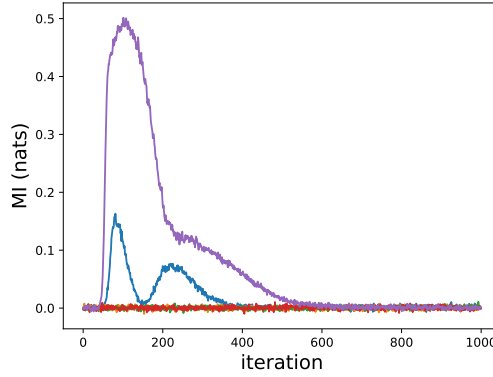


Figure A1: Example traces of sample estimates of $MI(t, r)$ to gauge disentanglement for all crowdsourcing benchmarks in Table 1 (datasets where 5000 iterations are used are subsampled every 5 time points). Only two datasets show transient substantial MI, and the model eventually eliminates any MI between t and r over the course of training.