Learning Image Labels On-the-fly for Training Robust Classification Models

Anonymous authors
Paper under double-blind review

Abstract

Current deep learning paradigms largely benefit from the tremendous amount of annotated data. However, the quality of the annotations often varies among labelers. Multi-observer studies have been conducted to study these annotation variances (by labeling the same data for multiple times) and its effects on critical applications like medical image analysis. This process indeed adds extra burden to the already tedious annotation work that usually requires professional training and expertise in the specific domains. On the other hand, automated annotation methods based on NLP algorithms have recently shown promise as a reasonable alternative, relying on the existing diagnostic reports of those images that are widely available in the clinical system. Compared to human labelers, different algorithms provide labels with varying qualities that are even noisier. In this paper, we show how noisy annotations (e.g., from different algorithm-based labelers) can be utilized together and mutually benefit the learning of classification tasks. Specifically, the concept of attention-on-label is introduced to sample better label sets on-the-fly as the training data. A meta-training based label-sampling module is designed to attend the labels that benefit the model learning the most through additional back-propagation processes. We apply the attention-on-label scheme on the classification task of a synthetic noisy CIFAR-10 dataset to prove the concept, and then demonstrate superior results (3-5% increase on average in multiple disease classification AUCs) on the chest x-ray images from a hospital-scale dataset (MIMIC-CXR) and hand-labeled dataset (OpenI) in comparison to regular training paradigms.

1 Introduction

Supervised deep learning methods, although proven to be effective on many tasks, rely heavily on the quality of the data and its corresponding annotations. Some tasks enjoy almost error-free annotation, such as handwritten numbers and simple natural images. However, for other applications including most medical image analysis tasks, the inherent ambiguity of the task leads to unavoidable noise and fuzziness within the annotations themselves, no matter how experienced the expert labelers are. Meanwhile, under a multi-labeler setting for quality control purpose, the significant intra- and inter-observer variability injects even more uncertainties into the resulting labels. Beyond the above challenges, for the specific task of chest X-ray image classification, due to the fact that most labels of the available large-scale open datasets are automatically mined by Natural Language Processing (NLP) algorithms, there will be yet another layer of error-prone operation on top of existing variability. Ideally, we would prefer multiple manually and reliably labelled close-to-truth annotations, while in reality, most of the data only have a single annotation from an algorithm with relatively low accuracy.

Learning to learn from a variety of data (labels) falls within the scope of meta-learning, which is popular in many machine learning application, e.g., domain adaptation/generalization (Finn et al., 2017; Dou et al., 2019) and few-shot learning (Snell et al., 2017; Liu et al., 2019). Those previous meta-learner models (as illustrated in Fig. 1(b)) often focus on learning the distribution of data (inputs of tasks) and specifying the update strategy of learner model parameters. Indeed, data from different sets (distributions) will contribute to the final learner model. On the contrary, we do not want the model to learn from erroneous labels (from less-experienced labelers) but learn only from "true" labels. We utilize the learner model parameters (via a meta-training process) to sample "true" labels for training a single learner model (as shown in Fig. 1(c)).
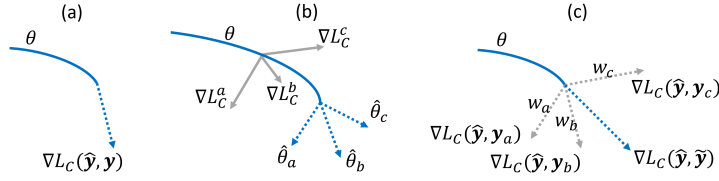
Figure 1: The diagram shows the difference of three learning paradigms in terms of how gradients are utilized for the training, i.e., (a) regular gradient based learning; (b) Meta-learning with multiple learning targets; (c) Proposed: learning the weights of each label $(\mathbf{y}_a, \mathbf{y}_b, \mathbf{y}_c)$ in meta-training and then computing the weighted summation $(\tilde{\mathbf{y}})$ of labels for computing the final loss with prediction $\hat{\mathbf{y}}$.

To address these challenges, we proposed an attention-on-label strategy to benefit the training from multiple labels on the same subject. Instead of the resource-demanding process of asking several human annotators to label the same data, we choose to utilize annotations from different algorithm-based labelers, which only add little overhead beyond the single-labeler scenario. A meta-training scheme is adopted and integrated in our proposed label-sampling module to attend the labels that benefit the model learning the most through additional back-propagation processes.

Our contributions in this work are three-fold: 1) We proposed a training framework to compute the image labels on-the-fly in the classification tasks. A meta-training based approach is introduced to attend and sample the labels from multiple annotators; 2) Gradient flows towards the label are investigated and implemented. Indeed, the multiple sets of labels are inputs to the training framework. Learnable operations of the labels will require additional updates of label-related model parameters; 3) We not only prove the concept on CIFAR-10 but also perform experiments on two chest X-ray datasets with both image-only and image-text classification tasks. In all datasets, superior performance of the proposed method is demonstrated in the image classification tasks compared to baseline methods.

2 RELATED WORKS

Meta learning: Meta learning aims to learn a generalizable model by situating itself at a higher level than conventional learning. This can be achieved in several ways such as finding weights that can be easily adapted to other models (Finn et al., 2017) or domains (Li et al., 2017) during the training process. Meta learning results in models that can converge quickly with a few examples (Ravi & Larochelle, 2017). They all share a similar meta-training process while the various goals of meta-training can divide them to different routes as examples shown in Fig. 1. In this work, we target at weighting the importance of each label set based on its meta-training feedback and learning from the most effective labels.

Learning from noisy labels: Learning from noisy labels (Natarajan et al., 2013; Li et al., 2020; Zhang et al., 2020b) has been a popular topic in deep learning due to its prevalence in many existing datasets with intra- and inter-observer variability, and the inherent uncertainties of both data and task themselves. For medical imaging applications with a high degree of ambiguity, this issue is even more significant. Recent works attempt to address this challenge via a consistency loss with a teacher model (Li et al., 2019), loss weighting with 2nd order derivatives (Zhang et al., 2020b), and for medical image specifically, an online uncertainty sample mining strategy (Xue et al., 2019). Please note that they all focus on noise labels from a single annotator, while we attempt to design an attention mechanism to utilize labels from different sources together.

Multi-label classification in chest X-ray: Because of its wide application and easy accessibility, chest X-ray is one of the major research areas in the field of medical image analysis. Among the pioneering works (Wang et al., 2017; Rajpurkar et al., 2017; Yao et al., 2017; Li et al., 2018; Tang et al., 2018; Hwang et al., 2019; Tang et al., 2020) in this area of deep learning, TieNet (Wang et al., 2018) first introduces an end-to-end trainable CNN-RNN architecture to extract distinctive text representations in addition to image features for improving label quality. More recently, a graph model was incorporated to enhance the learning accuracy (Zhang et al., 2020a).

Multi-observer studies: To ensure the annotation quality, especially for medical images where high expertise is required, it is common to have multiple set of labels on the same set of data (Rasch

label sets	atelectasis	cardiomegaly	consolidation	edema	pneumonia	pneumothorax
negbio_u	10986	11899	3348	13204	19029	1112
negbio_p	47804	40509	11088	27911	16122	9885
u/p ratio	0.229	0.293	0.301	0.473	1.18	0.112
chexpert_u	10662	6235	4446	13817	18915	1177
chexpert_p	47629	46373	11231	28339	16757	11046
u/p ratio	0.223	0.134	0.395	0.487	1.128	0.106

Table 1: Uncertainties (_u) and positives (_p) of 6 sample disease findings from two labelers.

et al., 1999; Cherian et al., 2005). In a sense, each annotation can be regarded as an estimation with uncertainty, good or bad, for the underlying "true label". Thus, algorithms taking this uncertainty factor into consideration are needed in order to make better use of such multi-observer data. Kohl et al. (2018) attempt to learn a distribution from a set of diverse but plausible segmentation from multiple graders. A recent work (Tanno et al., 2019) proposed to model annotators by a confusion matrix which is jointly estimated during classification. In our work, instead of human annotators with different skill-levels, we employed several "algorithmic labelers" to generate multiple annotations from the same raw data. Comparing with their human counterparts, there exists less limitations and cost to increase the number of labelers, while the resulting labels can be more noisy. Hence, we choose a different strategy of attention model and meta-learning to benefit the model learning process.

3 Learning from Data with Multiple Noisy Annotations

The accuracy of NLP algorithm-based labelers has been studied (Peng et al., 2018; Irvin et al., 2019) and verified by a small set of hand-labeled data (based on associated report texts). The noises in annotations could be traced from many sources, e.g., algorithmic errors, incomplete information in reports, and misjudgement from the clinicians. All of these could elevate the uncertainty and impair the reliability of the publish data and associated ground-truth labels, in terms of their utilization in modern machine learning paradigms. "Who to believe?" becomes a fundamental question to answer, which will ultimately have a significant impact on the performance of trained model.

MIMIC-CXR dataset (Johnson et al., 2019) provides the labels sets from two independent algorithm based annotators with positive, negative, and uncertain cases. The availability of all these different sourced labels enables the observation of the uncertainty inherent to some data sample (with different values in multiple label sets). As shown in Table 1, there are many uncertain cases in each kind of finding while the uncertainty / positive ratio may vary in diseases (ranged from 10% to 110%).

Here, we try to tackle this problem by training a single multi-label classification model while considering all the available label sets. A novel attention-on-label training process is introduced. For each set of labels, we perform individual back-propagation as a form of meta-training and then compute the new image/image+text feature using the individual updated model. Based on the new features, we attended to the label set with more representative features (as a weight) and sample the weighted summary of labels for the final update of the model in each iteration. Fig. 2 illustrates the overall architecture and learning processes for both image and label model parameters.

3.1 Multi-label Classification Baseline

We represent each image with x and the labels of classes for each image as a binary label vector $\mathbf{y} = [y_1, ..., y_n, ..., y_N], y_n \in \{0, 1\}$. $y_n = 1$ indicates the presence of corresponding disease pattern or other findings in the image. In the multi-label disease classification setting, the presence of each finding is predicted separately by producing a likelihood after applying the sigmoid on each logit. For the experiments on CIFAR (as multi-class classification), we also use the one-hot binary vectors to represent the labels and predictions. Therefore, the proposed framework could be utilized for both multi-label and multi-class classification tasks.

Our proposed attention-on-label scheme could be applied to a large variety of pre-trained CNN architectures. Without loss of generality, we take the common ResNet-50 (from Conv1 to Res5c) as our backbone network. A global averaging pooling (GAP) layer was applied to transform the activation from convolutional layers into a one dimension image feature F. The reason for applying a
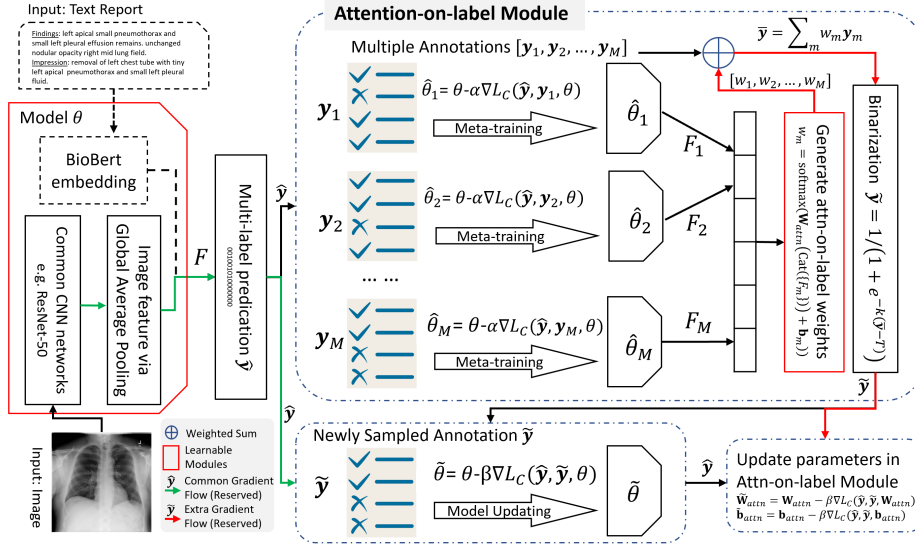
Figure 2: Overview of the proposed attention-on-label learning framework.

GAP layer is the necessity to pass concatenated image and text features to a fully-connected layer for the final classification.

We adopt the most common loss functions for the multi-label classification prediction $\hat{\mathbf{y}}$, i.e., binary cross entropy (BCE) loss: $\mathcal{L}_C(\hat{\mathbf{y}}, \mathbf{y}) = -1/N \sum_{i=1}^{N} y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)$. Other more advanced losses could also be employed, while this type of improvement is out-of-scope in this paper. Here, we would like to demonstrate the feasibility and benefit of applying our proposed meta training process with attention-on-label over a vanilla model. Other critical issues, like the unbalanced numbers of pathology comparing with "normal" classes, are not considered here either to keep the evaluation simple and effective.

3.2 ATTENTION ON LABELS

The overall training procedure is illustrated in Algorithm 1. For each training iteration, we input each data entry (x, Y) from the training set and $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_m, \ldots, \mathbf{y}_M\}$ are M sets of image labels. During the meta-training, we compute the BCE loss ($\mathcal{L}_C(\hat{\mathbf{y}}, \mathbf{y}_m, \theta)$) between the prediction $\hat{\mathbf{y}}$ of current classification model θ and label set $\mathbf{y}_m$ and then perform the back-propagation to compute a new set of model parameters ($\hat{\theta}_m$),

$$\hat{\theta}_m = \theta - \alpha \nabla_\theta \mathcal{L}_C(\hat{\mathbf{y}}, \mathbf{y}_m, \theta), \tag{1}$$

α is the learning rate for this meta-training process. We then compute a set of new features $\{F_m, m \in \{1, ..., M\}\}$ via the inference of image x using each meta-model $\hat{\theta}_m$ individually. $\{F_m\}$ could be either image features (i.e., output of the GAP) or one concatenated with text embedding (detailed in Section 3.3). $\{F_m\}$ represent the feedback of model updates with each label set $\mathbf{y}_m$, i.e., the change that each $\mathbf{y}_m$ has brought to the model θ. Other types of feedback from each noisy label could also be utilized here, e.g., the gradients $\{\nabla_\theta \mathcal{L}_C(\hat{\mathbf{y}}, \mathbf{y}_m, \theta), m \in \{1, ..., M\}\}$. Here, we take $\{F_m\}$ as an example to compute the weight w_m for each label set via a softmax based attention mechanism,

$$w_m = \text{Softmax}(\mathbf{W}_{attn}(\text{Cat}(\{F_m\}) + \mathbf{b}_{attn}), \tag{2}$$

where $\mathbf{W}_{attn}$ and $\mathbf{b}_{attn}$ are learnable parameters in our attention-on-label module. Softmax is the activation function. Cat represents the concatenation as a stack of all features. w_m indicates the importance/correctness of label set $\mathbf{y}_m$ and is applied to compute the weighted average of all label sets for each data sample. Here, we employ a common softmax-based attention mechanism (Xu et al., 2015; Chen et al., 2017) as a sample case and many other more complex learning-based attention mechanisms can be adopted to compute the weights, e.g., self-attention (Vaswani et al., 2017).

Algorithm 1 Meta-training with the attention-on-label module

1: Randomly initialize θ
2: **for** each data entry (x, Y) **do**
3:     **for** $m \in \{1 : M\}$ **do**
4:         Compute updated parameters with gradients: $\hat{\theta}_m = \theta - \alpha\nabla_\theta\mathcal{L}_c(\hat{\mathbf{y}}, \mathbf{y}_m, \theta)$
5:         Compute new features F_m using the newly updated $\hat{\theta}_m$
6:     Stack and concatenate the features: $\text{Concat}(\{F_m\})$
7:     Compute softmax attentions to generate the weight w_m for each feature F_m (Eq. 2)
8:     Sample the new label $\bar{\mathbf{y}}$ for each data sample using the weight w_m
9:     Perform differentiable binarization for each new label in $\bar{\mathbf{y}} \rightarrow \tilde{\mathbf{y}}$
10:     Update the final image model $\tilde{\theta} \leftarrow \theta - \beta\nabla\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}, \theta)$
11:     Manually compute and back-propagate the gradients for the attention-on-label model ($\mathbf{W}_{attn}, \mathbf{b}_{attn}$)

The values of label vectors after weighted average $\bar{\mathbf{y}} = \sum_m w_m\mathbf{y}_m$ are in the range of $[0, 1]$, which is rather ambiguous for the model to learn. Binarization will be useful to cast the value close to either 0 or 1, but it is not differentiable and will disrupt the gradient flow. Therefore, we adopt the differentiable binarization function as first introduced in Liao et al. (2019),

$$\tilde{\mathbf{y}} = \frac{1}{1 + e^{-k(\bar{\mathbf{y}}-T)}}, \tag{3}$$

where k sets the sharpness of a 0 to 1 cliff. T is a threshold to slightly adjust the value range. Finally, we update the model once more with the attended label and a globla learning rate β for this iteration,

$$\tilde{\theta} \leftarrow \theta - \beta\nabla\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}, \theta). \tag{4}$$

Gradient Flows Towards Labels: Extra gradient flows (highlighted in red in Fig.2) are required for training our attention-on-label mechanism, specifically the parameters in Eq. 2. Most of the current learning frameworks have gradient flows (highlighted in green in Fig.2) with the images in the end since the labels are usually fixed or smoothed in advance (Müller et al., 2019). However, the gradients in our proposed framework not only flow to the images but also go through towards the labels since the final labels $\tilde{\mathbf{y}}$ are computed on-the-fly with learned weights/attentions. To our best knowledge, this concept of gradients towards labels is novel and has not been investigated and implemented before. Indeed, the inputs to the attention-on-label module are M sets of labels and the computed features $\{F_m\}$, which are detached (without auto-computed gradients) and stacked during the meta-training. Therefore, additional parameter updating is required at the end of each iteration,

$$\tilde{\mathbf{W}}_{attn} \leftarrow \mathbf{W}_{attn} - \beta\nabla\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}, \mathbf{W}_{attn}), \ \tilde{\mathbf{b}}_{attn} \leftarrow \mathbf{b}_{attn} - \beta\nabla\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}, \mathbf{b}_{attn}). \tag{5}$$

3.3 Image-text Embedding

Clinical textual material, e.g., clinical notes (Pelka et al., 2019) and radiology report (Wang et al., 2018), contains richer information. We include the text report as an input to the classification problem to see if our proposed learning process will still benefit the learning and further improve the classification accuracy. There are a variety of approaches to generate text embedding, e.g., Fisher vectors of word2vec (Klein et al., 2015), bidirectional LSTMs (Wang et al., 2016), and the most recently developed BERT model (Devlin et al., 2018). To keep the simplicity of our baseline model, we embed the text report to a 768 dimension real-valued vector using the uncased version of BioBert features (Lee et al., 2020), followed by two fully connected layers with 512 neurons each.

4 Datasets

CIFAR-10: We simulate 5 different types of annotators (with different experience) in a similar manner as Tanno et al. (2019) by injecting label noises into the training set of **CIFAR-10**, namely 1) hammer-spammer (HS), 2) structured-flips (SF), 3) ordered-confusion (OC), 4) Adversarial (AD), and 5) average (AVG) of 1-4. Each set of noisy labels are generated based on the defined confusion matrices for each type (as shown in Fig. 3). Whether each sample would have a noisy label is randomly selected while the overall noisy distribution should correspond to each confusion matrix, individually. Within all the noisy training data, we randomly select 20% as the validation set.
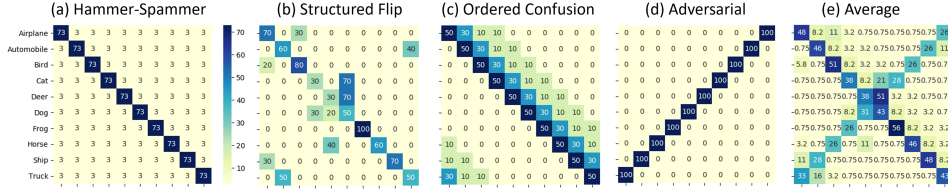
Figure 3: Confusion matrices of noisy labels from 5 different types of simulated annotators.

MIMIC-CXR: The MIMIC Chest X-ray (Johnson et al., 2019) Database is a large publicly available dataset of chest radiographs with labels mined from image-associated text radiology reports using two different NLP based annotation tools, i.e., negbio (Peng et al., 2018) and chexpert (Irvin et al., 2019). The uncertain findings are marked as -1 in the original datasheet. Here, uncertainties are set to either 0 or 1 to form 4 different label sets, i.e., negbio_u0, negbio_u1, chexpert_u0, and chexpert_u1. The dataset contains 377,110 radiographs and labels from the 227,827 free-text radiology reports. Totally, 14 disease findings are listed ($N = 14$). In our experiments, only the frontal view images are adopted, the number of which is equal to the number of reports. We utilize the official data split for training, validation and testing.

MIMIC-CXR 1K hand-labeled Test: Following the same labeling protocol proposed by Demner-Fushman et al. (2012), we randomly selected 1000 images and associated textual reports from the testing set of **MIMIC-CXR** and one of our staff (trained by a board-certified radiologist) hand-labeled the 1000 images by assigning the 14 labels manually to each image based on the reports, which will be released publicly. We believe it will also benefit the research in chest x-ray disease classification.

OpenI: OpenI (Demner-Fushman et al., 2015) is a public dataset of chest X-rays collected from multiple institutes by Indiana University. In total, we fetch 3,851 unique radiology reports and 7,784 associated frontal/lateral images. To keep the consistency with **MIMIC-CXR** dataset, we use the same 14 categories of findings as mentioned above in the experiments. In our experiments, only 3,643 unique front view images and corresponding reports are evaluated.

5   EXPERIMENTS:

The following methods in addition to the proposed method (**Ours**) are included in the comparison:

ResNet-50 (**R50**, He et al., 2016): We take the network based on ResNet-50 as a baseline. It adopts an ImageNet pre-trained ResNet-50 (from Conv1 to Res5c) as the backbone, followed by a GAP layer and a fully-connect layer for the final classification. Optionally, BioBert embedded text features will be concatenated with the output of GAP before the classification.

CM (Tanno et al., 2019): This method multiplies a confusion matrix with the probability that the model produces for each class. The basic assumption is that this confusion matrix can correct the missed labeled data and return the probability for the truth using the learned confusion matrices. We carefully implement it according to the code snapshot provided by the authors.

NG (Zhang et al., 2020a): This method utilizes the prior knowledge of the disease relations as a form of knowledge graph. By injecting such prior knowledge and employing a graph convolutional network, it learns the underlying info for the final classification and report generation task. It is worthy to note that the results we report in the experiments are produced by a model that is both trained and evaluated on the **OpenI** dataset.

TieNet (Wang et al., 2018): It focuses more on how to learn the image and text embedding together using a CNN+RNN framework. Its LSTM based text embedding is relatively more complicated but also more representative through learning. We only adopt the text embedding from a pre-trained BioBERT model for our comparison, which is less customized.

Evaluation Metric: Receiver Operating Characteristic (ROC) curve is the standard metric to evaluate the performance of multi-label classification tasks. Here, Area Under the Curve (AUC) values are computed for all the experiments on **MIMIC-CXR** and **OpenI**. We compute the multi-class classification accuracy (using *sklearn.metrics.accuracy_score*) for all the experiments on **CIFAR-10**.

CIFAR-10	HS	SF	OC	AD	AVG	CM	Ours
Noise-Level	30%	40%	50%	100%	45%	-	-
Accuracy	0.808	0.438	0.555	0.025	0.533	<u>0.593</u>	**0.677**

Table 2: Averaged classification accuracy (with 5 noisy label sets) on **CIFAR-10** testing set (clean).

5.1 IMPLEMENTATION DETAILS:

For pre-processing, we resize the image to 256×256 (while keeping the size of 32×32 for **CIFAR-10**) and normalize the image intensities to [0, 1]. No data augmentation is employed in our experiments. As mentioned above, we set the learning rate for the meta-training phase as $\alpha = 0.2$ and global learning rate as $\beta = 1e - 4$. The best model for all hyper-parameters are determined via validation. We set $k = 50$ and $T = 0.5$ empirically in the differentiable binarization module. We use a single NVIDIA Titan-X Pascal for training each classification model with a uniform batch size $B = 32$ across all the experiments. Adam optimizer is utilized for training all the compared models.

5.2 CLASSIFICATION RESULTS ON CIFAR-10

To prove the concept, we employ **CIFAR-10** with 5 types of added noises in labels to illustrate that the proposed attention-on-label scheme can be beneficial for the model training using multiple noise label sets. Table 2 illustrates the multi-class classification accuracy on the **CIFAR-10** testing set. Noise-levels are computed in (1-Accuracy) using corresponding confusion matrices. In general, better quality labels and data lead to better trained models. Noise introduced by structured flips confuse the model training more than other types. Here, HS represents a more experienced set of annotators and models trained with it obtained a high accuracy. AVG represents the results of learning from a label set with simple average of the noise-level (defined by the confusion matrices) of all annotators. Our proposed method achieves over 12% and 8% performance improvements over AVG and the previous state-of-the-art **CM**.

In addition, we also investigate how the label noise level and number of annotators will affect the model performance (detailed in Sec. A.2). With noise level ranged from 10% to 80%, **Ours** can constantly achieve better or similar results as the models trained with the relatively more accurate labels (e.g., AVG and HS). While experimenting with different number (2-5) of annotators, we observe a surge of the classification accuracy after including more than 2 annotators' labels. **Ours** often out-performs **CM** while the number of annotators increase, especially when labels with high noise levels are employed.

5.3 CLASSIFICATION RESULTS ON CHEST X-RAY IMAGES

Classification Results Using Chest X-Ray Images: Table 3 shows the evaluation results for all the compared method using only the images as the input to the model. The left side of Table 3 shows the AUCs of all the finding categories from **R50**, **CM** and **Ours**. The averaged AUC for **Ours** actually drops from the baseline. Considering the fact that the testing set of **MIMIC-CXR** dataset is also using the algorithm based labels, our predictions actually diverge from those noisy labels and lean to the underlying true labels. It is proved by the results illustrated in **OpenI** dataset section (right part of Table 3). **OpenI** dataset has the hand-labeled ground truth and our method is able to achieve over 4% increase in the averaged AUC, which is also greater than what the **CM** method achieves. Although, both **NG** and **TieNet** partially utilized the report textual information in their image classification framework, **Ours** still is able to obtain equivalent or better results in most of the detailed disease categories. As mentioned above, those disease categories with larger amount of uncertainties provide more information and therefore benefit more from the proposed meta-training process, e.g., Atelectasis and Consolidation. Please see the Appendix for more detailed results.

Classification Results Using Both Chest X-ray Images and Report Texts: We observe similar improvements on the image-text classification task. Indeed, the text report contains more information about the disease diagnosis (maybe more than the image itself). We also observe the increase of the overall AUCs. In this case, our proposed meta-training with attention-on-label scheme also helps to boost the classification performance with a significant margin.

AUC	MIMIC-CXR Test (NLP)			OpenI (hand-labeled)				
Disease	**R50**	**CM**	**Ours**	**NG**	**TieNet**	**R50**	**CM**	**Ours**
Image only								
Atelectasis	0.821	0.832	0.826	0.833	0.774	0.781	0.81	0.826
Cardio.	0.825	0.852	0.879	0.913	0.847	0.859	0.881	0.879
Consol.	0.762	0.751	0.906	-	-	0.829	0.842	0.906
Edema	0.887	0.903	0.885	0.931	0.879	0.895	0.924	0.885
E-cardio	0.74	0.757	0.725	-	-	0.795	0.758	0.725
Fracture	0.722	0.771	0.632	0.671	-	0.513	0.596	0.632
Lesion	0.765	0.744	0.643	0.643	0.658	0.585	0.58	0.643
Opacity	0.814	0.82	0.775	0.803	-	0.742	0.738	0.775
No-finding	0.857	0.863	0.775	-	0.747	0.754	0.739	0.775
Effusion	0.906	0.914	0.942	0.942	0.899	0.912	0.932	0.942
Pleural-o.	0.866	0.829	0.705	-	-	0.648	0.676	0.705
Pneumonia	0.809	0.809	0.871	0.863	0.731	0.781	0.823	0.871
Pneum-x	0.866	0.858	0.833	0.843	0.709	0.793	0.882	0.833
Devices	0.92	0.926	0.729	0.805	-	0.628	0.655	0.729
Average	0.825	0.830	0.794	-	-	0.751	<u>0.774</u>	**0.794**
Image & Text								
Average	0.941	0.927	0.931	-	-	0.809	<u>0.824</u>	**0.835**

Table 3: Classification AUCs for 14 findings in Chest X-Rays using the testing set of **MIMIC-CXR** (with NLP generated labels) and **OpenI** dataset (hand-labeled GT). E-cardio: enlarged-cardiomediastinum; Pneum-x: pneumothorax. More details in Table 6 of the Appendix.

AUC	MIMIC-CXR 1K hand-labeled Test						
	negbio_u1	negbio_u0	chexpert_u1	chexpert_u0	**R50**	**CM**	**Ours**
Average	**0.869**	0.805	<u>0.868</u>	0.809	<u>0.842</u>	0.837	**0.868**

Table 4: Averaged classification AUCs for 14 findings using the **MIMIC-CXR 1K hand-labeled Test** set. negbio_u1, negbio_u0, chexpert_u1, and chexpert_u0 are derived from the original NLP mined labels by setting the uncertainty to either 1 or 0. See all diseases' AUCs in the Appendix.

As shown in Table 4, we further demonstrate the performance improvement from our proposed method on the **MIMIC-CXR 1K hand-labeled Test** data. Note that the absolute accuracy is higher than the ones reported on **OpenI** dataset. It may indicate the domain gap between **MIMIC-CXR** and **OpenI** datasets. Additionally, we evaluate the accuracy of all four NLP mined label sets. The choice of uncertainty makes the NLP labels have quite different accuracy (6% gap) on the **MIMIC-CXR 1K hand-labeled Test** set, while our proposed method actually achieve higher or similar accuracies in comparison to the NLP annotators while a model trained with NLP mined labels usually can not reach the same level of accuracy as the labels themselves, e.g., **R50** is trained with negbio_u1.

6   CONCLUSION AND FINAL REMARKS

In this paper, we introduced a novel learning framework (meta-training with the attention-on-label module) for handling data with multiple noisy label sets. The variability that the multiple label sets bring could in fact benefit the learning of a more accurate and robust model. We demonstrated the application of our proposed method on both multi-label and multi-class classification tasks and we believe it will be quite straight-forward to adapt it for other task, e.g., image segmentation. Indeed, the proposed method provides a viable way to handle the challenges of learning large-scale data with algorithm generated labels, where the label is a huge burden, especially for medical image analysis.

The proposed label sampling module convert the hard labels (0 or 1) to soft labels (a value between 0 and 1) and it also reduces the overfit towards erroneous labels, which is similar to the idea of label smoothing (Müller et al., 2019). Different to hard label smoothing, we assigned the new label on-the-fly (based on the network feedback) instead of instantly replacing 0 and 1 with $0 + \epsilon$ and $1 - \epsilon$. Such setting is extremely effective to handle the partial label errors in multi-label classification. Indeed, label smoothing is a form of loss-correction (Lukasik et al., 2020) and we also prove the proposed label weighting is equivalent to directly performing the weighted average of the losses (generated from using different label sets) to form the final loss (see the proof in A.1).

REFERENCES

Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5659–5667, 2017.

Thomas Cherian, E Kim Mulholland, John B Carlin, Harald Ostensen, Ruhul Amin, Margaret de Campo, David Greenberg, Rosanna Lagos, Marilla Lucero, Shabir A Madhi, et al. Standardized interpretation of paediatric chest radiographs for the diagnosis of pneumonia in epidemiological studies. *Bulletin of the World Health Organization*, 83:353–359, 2005.

Dina Demner-Fushman, Sameer Antani, Matthew Simpson, and George R Thoma. Design and development of a multimodal biomedical information retrieval system. *Journal of Computing Science and Engineering*, 6(2):168–177, 2012.

Dina Demner-Fushman, Marc D Kohli, Marc B Rosenman, Sonya E Shooshan, Laritza Rodriguez, Sameer Antani, George R Thoma, and Clement J McDonald. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310, 2015. doi: 10.1093/jamia/ocv080.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *Advances in Neural Information Processing Systems*, pp. 6447–6458, 2019.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 1126–1135. JMLR.org, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Eui Jin Hwang, Sunggyun Park, Kwang-Nam Jin, Jung Im Kim, So Young Choi, Jong Hyuk Lee, Jin Mo Goo, Jaehong Aum, Jae-Joon Yim, Julien G Cohen, et al. Development and validation of a deep learning–based automated detection algorithm for major thoracic diseases on chest radiographs. *JAMA network open*, 2(3):e191095–e191095, 2019.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 590–597, 2019.

Alistair E. W. Johnson, Tom J. Pollard, Seth J. Berkowitz, Nathaniel R. Greenbaum, Matthew P. Lungren, Chih-ying Deng, Roger G. Mark, and Steven Horng. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific Data*, 6(1):317, 2019.

Ashish Khetan, Zachary C Lipton, and Anima Anandkumar. Learning from noisy singly-labeled data. *arXiv preprint arXiv:1712.04577*, 2017.

Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. Associating neural word embeddings with deep image representations using fisher vectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4437–4446, 2015.

Simon Kohl, Bernardino Romera-Paredes, Clemens Meyer, Jeffrey De Fauw, Joseph R Ledsam, Klaus Maier-Hein, SM Ali Eslami, Danilo Jimenez Rezende, and Olaf Ronneberger. A probabilistic u-net for segmentation of ambiguous images. In *Advances in Neural Information Processing Systems*, pp. 6965–6975, 2018.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. *arXiv preprint arXiv:1710.03463*, 2017.

Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. Learning to learn from noisy labeled data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5051–5059, 2019.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020.

Zhe Li, Chong Wang, Mei Han, Yuan Xue, Wei Wei, Li-Jia Li, and Li Fei-Fei. Thoracic disease identification and localization with limited supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8290–8299, 2018.

Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. *arXiv preprint arXiv:1911.08947*, 2019.

Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 10551–10560, 2019.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? *arXiv preprint arXiv:2003.02819*, 2020.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems*, pp. 4696–4705, 2019.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pp. 1196–1204, 2013.

Obioma Pelka, Felix Nensa, and Christoph M Friedrich. Branding-fusion of meta data and musculoskeletal radiographs for multi-modal diagnostic recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

Yifan Peng, Xiaosong Wang, Le Lu, Mohammadhadi Bagheri, Ronald Summers, and Zhiyong Lu. Negbio: a high-performance tool for negation and uncertainty detection in radiology reports. *AMIA Summits on Translational Science Proceedings*, 2018:188, 2018.

Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

Coen Rasch, Isabelle Barillot, Peter Remeijer, Adriaan Touw, Marcel van Herk, and Joos V Lebesque. Definition of the prostate in ct and mri: a multi-observer study. *International Journal of Radiation Oncology* Biology* Physics*, 43(1):57–66, 1999.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. *arXiv preprint arXiv:1803.09050*, 2018.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pp. 4077–4087, 2017.

Y Tang, Y Peng, K Yan, M Bagheri, BA Redd, CJ Brandon, Z Lu, M Han, J Xiao, and RM Summers. Automated abnormality classification of chest radiographs using deep convolutional neural networks. *NPJ Digital Medicine*, 3, 2020.

Yuxing Tang, Xiaosong Wang, Adam P Harrison, Le Lu, Jing Xiao, and Ronald M Summers. Attention-guided curriculum learning for weakly supervised classification and localization of thoracic diseases on chest radiographs. In *International Workshop on Machine Learning in Medical Imaging*, pp. 249–258. Springer, 2018.

Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C Alexander, and Nathan Silberman. Learning from noisy labels by regularized estimation of annotator confusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11244–11253, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Cheng Wang, Haojin Yang, Christian Bartz, and Christoph Meinel. Image captioning with deep bidirectional lstms. In *Proceedings of the 24th ACM international conference on Multimedia*, pp. 988–997, 2016.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2097–2106, 2017.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, and Ronald M Summers. Tienet: Text-image embedding network for common thorax disease classification and reporting in chest x-rays. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9049–9058, 2018.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057, 2015.

C. Xue, Q. Dou, X. Shi, H. Chen, and P. Heng. Robust learning at noisy labeled medical images: Applied to skin lesion classification. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pp. 1280–1283, April 2019.

Li Yao, Eric Poblenz, Dmitry Dagunts, Ben Covington, Devon Bernard, and Kevin Lyman. Learning to diagnose from scratch by exploiting dependencies among labels. *arXiv preprint arXiv:1710.10501*, 2017.

Yixiao Zhang, Xiaosong Wang, Ziyue Xu, Qihang Yu, Alan Yuille, and Daguang Xu. When radiology report generation meets knowledge graph. In *Proceedings of AAAI*, 2020a.

Zizhao Zhang, Han Zhang, Sercan O Arik, Honglak Lee, and Tomas Pfister. Distilling effective supervision from severe label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9294–9303, 2020b.

A APPENDIX

Due to the limited space in the main text, we want to discuss some important related issues here and meanwhile demonstrated additional experiments of the proposed attention-on-label module. It starts with a discussion about what the attention-on-label really does and follows by extensive experimental results on both CIFAR-10 and chest X-rays image datasets.

A.1 ATTENTION-ON-LABEL IS A FORM OF LOST-CORRECTION

Here, we try to demonstrate how the proposed attention-on-label module really works and discuss its connection to other prior arts. First, we present a theorem, indicating that attention weights applied on the labels can also be employed to re-weight the losses that are computed using label sets from different annotators.

Theorem 1 (Label-Sampling Formulation). *The loss computed with the model prediction $\hat{\mathbf{y}}$ and the sampled label $\tilde{\mathbf{y}} = \sum_{m=1}^{M} w_m \mathbf{y}_m$,*

$$\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}) = \mathcal{L}_C(\hat{\mathbf{y}}, \sum_{m=1}^{M} w_m \mathbf{y}_m) \qquad (6)$$

is equivalent to a weighted summation of losses computed with each set of label $\mathbf{y}_m$,

$$\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}) = \sum_{m=1}^{M} w_m \mathcal{L}_C(\hat{\mathbf{y}}, \mathbf{y}_m), \tag{7}$$

where

$$\mathcal{L}_C(\hat{\mathbf{y}}, \mathbf{y}) = -\frac{1}{N} \sum_{i=1}^{N} y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \tag{8}$$

is the binary cross-entropy loss. w_m *is the attention weights as defined in the main text,* N *represents the number of classes in the label, and* M *is the total number of label sets (annotators).*

Thus, the proposed attention-on-label mechanism can be transformed to a loss re-weighting module, where the weights are the same ones that are generated from the back-propagation feedback using different label sets in the meta-training process. Indeed, it shares a similar insight with Ren et al. (2018) while Ren et al. (2018) re-weighted the losses based on the variation of input images. Similar to Theorem 1, we can also learn that re-weighting the losses is equivalent to the weighted sample of network predictions (maybe with different data samples as input). Accordingly, **the noise in data (on both images and labels) can be observed and corrected via studying the losses.** It also leads to a broader research topic of modeling the distribution of losses, e.g., recent work on learning with noisy label (Li et al., 2020) by modeling the losses with Gaussian mixture models. We believe our work probably could be explored further in that direction.

Proof of Theorem 1. Recall that $\tilde{\mathbf{y}} = \sum_{m=1}^{M} w_m \mathbf{y}_m$ is the attention-weighted sampling of different label sets. $\mathbf{y}_m = [y_1^m, ..., y_n^m, ..., y_N^m], y_n^m \in \{0, 1\}$ is one of the M set labels.

$$\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}) = -\frac{1}{N} \sum_{i=1}^{N} [\tilde{y}_i \log(\hat{y}_i) + (1 - \tilde{y}_i) \log(1 - \hat{y}_i)]$$

$$= -\frac{1}{N} \sum_{i=1}^{N} [\sum_{m=1}^{M} w_m y_i^m \log(\hat{y}_i) + (1 - \sum_{m=1}^{M} w_m y_i^m) \log(1 - \hat{y}_i)]$$

$$= -\frac{1}{N} \sum_{i=1}^{N} \{\sum_{m=1}^{M} w_m [y_i^m \log(\hat{y}_i) - y_i^m \log(1 - \hat{y}_i)] + \log(1 - \hat{y}_i)\}$$

$$= -\frac{1}{N} \sum_{i=1}^{N} \{\sum_{m=1}^{M} w_m [y_i^m \log(\hat{y}_i) + (1 - y_i^m) \log(1 - \hat{y}_i) - \log(1 - \hat{y}_i)] +$$
$$\log(1 - \hat{y}_i)\}$$

$$= -\frac{1}{N} \sum_{i=1}^{N} \{\sum_{m=1}^{M} w_m [y_i^m \log(\hat{y}_i) + (1 - y_i^m) \log(1 - \hat{y}_i)] +$$
$$(1 - \sum_{m=1}^{M} w_m) \log(1 - \hat{y}_i)\}$$

Since w_m is computed using the softmax function and $\sum_{m=1}^{M} w_m = 1$,

$$\mathcal{L}_C(\hat{\mathbf{y}}, \tilde{\mathbf{y}}) = -\frac{1}{N} \sum_{i=1}^{N} \{\sum_{m=1}^{M} w_m [y_i^m \log(\hat{y}_i) + (1 - y_i^m) \log(1 - \hat{y}_i)]\}$$

$$= \sum_{m=1}^{M} w_m \{-\frac{1}{N} \sum_{i=1}^{N} [y_i^m \log(\hat{y}_i) + (1 - y_i^m) \log(1 - \hat{y}_i)]\}$$

$$= \sum_{m=1}^{M} w_m \mathcal{L}_C(\hat{\mathbf{y}}, \mathbf{y}_m)$$
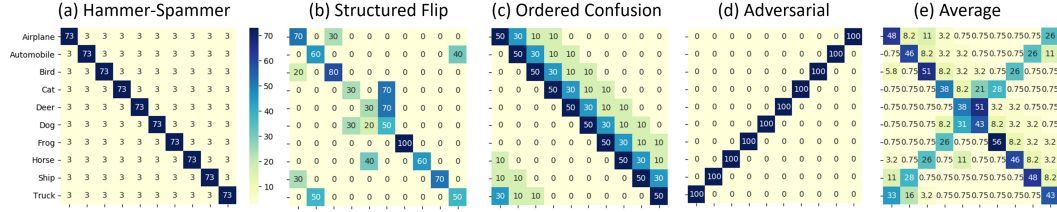
$\square$

Figure 4: Confusion matrices of noisy labels from 5 different types of simulated annotators.

A.2 Additional Experiments on CIFAR-10

A.2.1 Details of Adding Noise to CIFAR-10 Training Data

We simulate 5 different types of annotators (with different experience) in a similar manner as Tanno et al. (2019) by injecting a noise into the ground-truth label of the training set in **CIFAR-10**.

Hammer-Spammer (HS): For each class, the annotation is correct with probability $p \in [0, 1]$ and otherwise chooses labels uniformly as defined in Khetan et al. (2017) .

Structured-Flips (SF): Similar to the HS annotator, SF is correct with a probability p and otherwise we flip the label of each class to another label (as easily confused ones), which is chosen as pre-defined pairs for each class. The pairs are defined as Airplane vs Bird, cat vs dog, dear vs cat, horse vs deer, ship vs airplane, and truck vs automobile.

Ordered-Confusion (OC): This annotator is likely to confuse the target class with "neighbouring" classes as defined in the corresponding confusion matrix.

Adversarial (AD): AD is an adversarial annotator, who has an high accuracy of grouping images from the same category while constantly giving the wrong labels.

Average (AVG): The AVG annotation is composed using a average of all 4 confusion matrices from previously defined 4 types of annotators.

The overall noise-level is defined as $1 - p$. Each set of noisy labels are generated based on the defined confusion matrices for each type (as samples shown in Fig.4, where the noise-levels for HS, SF, OC, AD are 30%, 40%, 50%, 100% individually). Whether each data entry would have a noisy label is randomly selected while the overall noisy distribution should correspond to each confusion matrix, individually.

A.2.2 How Does Different Noise Level of Labels Affect the Performance?

For this experiment, we want to see how the noise-level will affect the model training and the performance of our proposed method and **CM**. We include 4 sets of labels, i.e., HS, SF, OC and AVG. For each time, all 4 sets share the same noise-levels (ranged from 10% to 80%). We excuse the AD label set from this experiment since its noise level can not be adjusted. We compare the prediction accuracy (in multi-class classification tasks) of baseline models (**R50**) that are trained using individual label sets and our proposed method. The performance of **CM** is also evaluated. Both **CM** and **Ours** are using all 4 label sets for the training. The results are shown in Figure 5. The proposed method can constantly achieve better or similar results as the models trained with the relatively more accurate labels (e.g., AVG and HS). Furthermore, our method outperforms **CM** in most of the noise-levels and has even bigger margins for noise-level 40% and 50%. Additionally, we can also see that SF noise harms the performance the most since similar objects with erroneous labels can confuse the model more.

A.2.3 How Does Different Number of Available Annotation Sets Affect the Performance?

In this experiment, we try to vary the number of available label sets (from 2, 3, 4 and 5 different types of annotators) used for training the proposed model (**Ours**) and **CM** at different noise level (at 10%,

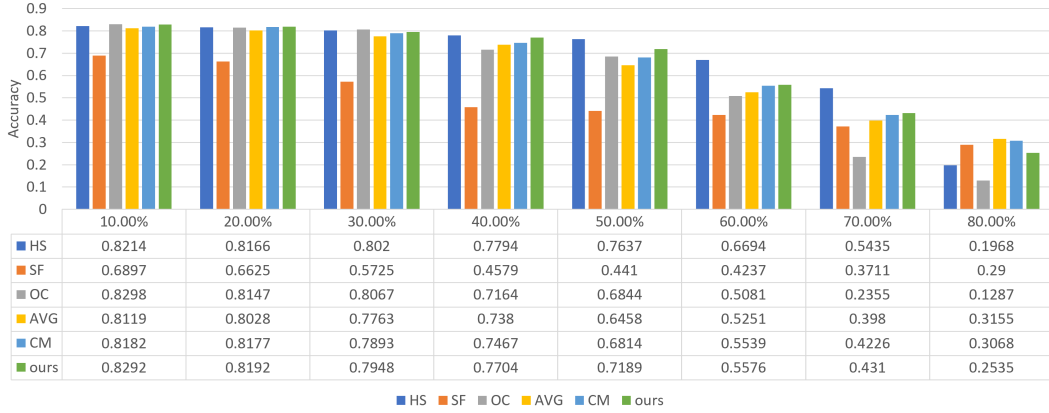|	10.00%	20.00%	30.00%	40.00%	50.00%	60.00%	70.00%	80.00%
HS	0.8214	0.8166	0.802	0.7794	0.7637	0.6694	0.5435	0.1968
SF	0.6897	0.6625	0.5725	0.4579	0.441	0.4237	0.3711	0.29
OC	0.8298	0.8147	0.8067	0.7164	0.6844	0.5081	0.2355	0.1287
AVG	0.8119	0.8028	0.7763	0.738	0.6458	0.5251	0.398	0.3155
CM	0.8182	0.8177	0.7893	0.7467	0.6814	0.5539	0.4226	0.3068
ours	0.8292	0.8192	0.7948	0.7704	0.7189	0.5576	0.431	0.2535

Figure 5: Classification accuracy using different label sets and methods over a range of noise-levels.
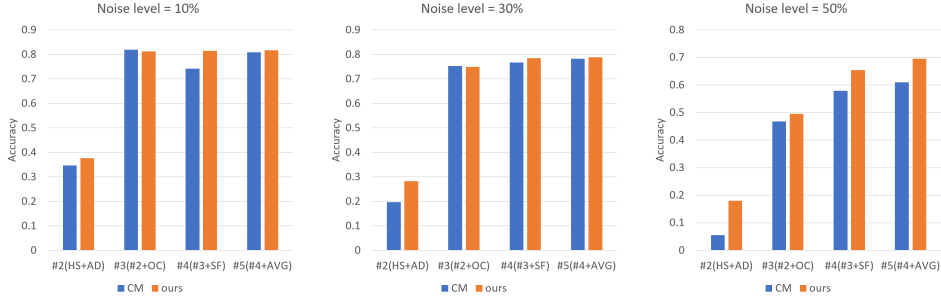

Figure 6: Classification accuracy using different number of label sets at noise-level 10%, 30%, and 50%.

30%, and 50%). We start with training model using 2 quite different label sets, i.e., HS and AD. Then we add OC, SF, and AVG one at a time to see how the increasing number of the label sets could affect the final classification performance. HS represents a relatively experienced annotators and AD can be seen as a totally 'bad' annotator. We start with these two sets of labels. Then, label sets from other types of annotators are added. As shown in the Figure 6, a surge of the accuracy can be observed after including more than 2 annotators. Considering that the adversarial annotator (with noise-level 100%) is among the initial two, both **CM** and **Ours** can learn better immediately after a third one (as a confirmation) jumps in. We can also observe that our method performs much better when the noise level is relatively high (50% in this case), while the CNN model itself may have already been capable of learning well from the labels with low noise levels.

A.3 BASELINE CLASSIFICATION RESULTS WITH DIFFERENT ANNOTATIONS

As shown in Table 1, we observe a large difference in uncertainty among various algorithm generated label sets. Therefore, we first want to see how these different label sets will affect the model training. In Table 5, we illustrate the averaged AUCs for four different label sets. $negbio\_u0$ is generated by setting all the uncertain cases to 0 (as negative cases) while $negbio\_u1$ is produced by giving all the uncertain cases to 1 (as positive cases). Similar process is applied to the $chexpert$ label sets as well. As shown in Table 5, the testing performance of all four models are relatively on the same level for both **MIMIC-CXR** and **OpenI**. It indicates that the CNN based model is not so sensitive to the change of labels and can overcome the noise in the label set to a certain degree, but we should note that it does not necessarily improve the overall performance of the trained model. We believe a large amount of data with higher quality labels will benefit the training and make the trained model more accurate and robust.

Baseline Model	negbio_u1	negbio_u0	chexpert_u1	chexpert_u0
AVG AUC (MIMIC-CXR)	0.825	0.821	0.824	0.810
AVG AUC (OpenI)	0.751	0.756	0.755	0.752

Table 5: Averaged AUCs with the baseline R50 from four label sets.

	MIMIC-CXR Test (NLP)			OpenI (hand-labeled)				
Disease	R50	CM	Ours	NG	Tie	R50	CM	Ours
Image only								
Atelectasis	0.821	0.832	0.826	0.833	0.774	0.781	0.81	0.826
Cardio.	0.825	0.852	0.879	0.913	0.847	0.859	0.881	0.879
Consol.	0.762	0.751	0.906	-	-	0.829	0.842	0.906
Edema	0.887	0.903	0.885	0.931	0.879	0.895	0.924	0.885
E-cardio	0.74	0.757	0.725	-	-	0.795	0.758	0.725
Fracture	0.722	0.771	0.632	0.671	-	0.513	0.596	0.632
Lesion	0.765	0.744	0.643	0.643	0.658	0.585	0.58	0.643
Opacity	0.814	0.82	0.775	0.803	-	0.742	0.738	0.775
No-finding	0.857	0.863	0.775	-	0.747	0.754	0.739	0.775
Effusion	0.906	0.914	0.942	0.942	0.899	0.912	0.932	0.942
Pleural-o.	0.866	0.829	0.705	-	-	0.648	0.676	0.705
Pneumonia	0.809	0.809	0.871	0.863	0.731	0.781	0.823	0.871
Pneum-x	0.866	0.858	0.833	0.843	0.709	0.793	0.882	0.833
Devices	0.92	0.926	0.729	0.805	-	0.628	0.655	0.729
Average	0.825	0.830	0.794	-	-	0.751	<u>0.774</u>	**0.794**
Image & Text								
Atelectasis	0.985	0.986	0.981	-	0.976	0.901	0.909	0.925
Cardio.	0.946	0.953	0.949	-	0.962	0.915	0.928	0.949
Consol.	0.911	0.913	0.904	-	-	0.914	0.891	0.907
Edema	0.955	0.952	0.956	-	0.995	0.903	0.915	0.939
E-cardio	0.923	0.916	0.936	-	-	0.581	0.714	0.598
Fracture	0.935	0.766	0.876	-	-	0.705	0.683	0.739
Lesion	0.865	0.885	0.814	-	0.96	0.615	0.607	0.649
Opacity	0.969	0.968	0.967	-	-	0.849	0.854	0.877
No-finding	0.975	0.972	0.968	-	0.936	0.79	0.82	0.867
Effusion	0.974	0.973	0.974	-	0.977	0.944	0.948	0.943
Pleural-o.	0.92	0.877	0.892	-	-	0.723	0.778	0.739
Pneumonia	0.927	0.931	0.933	-	0.994	0.812	0.834	0.889
Pneum-x	0.929	0.919	0.926	-	0.96	0.879	0.879	0.853
Devices	0.971	0.969	0.97	-	-	0.796	0.787	0.821
Average	0.941	0.927	0.931	-	-	0.809	<u>0.824</u>	**0.835**

Table 6: Classification AUCs for 14 findings in both Chest X-ray images and associated text reports using the testing set of **MIMIC-CXR** (with NLP generated labels) and **OpenI** dataset (hand-labeled GT). E-cardio: enlarged-cardiomediastinum; Pneum-x: pneumothorax

A.4   ADDITIONAL EXPERIMENTS ON CHEST X-RAY IMAGES AND TEXTUAL REPORTS

In addition to the classification results using image only shown in the main text, we show the complete results for classifying the image and text together on the testing set of **MIMIC-CXR** (with NLP mined image labels) and **OpenI** (with hand-labeled labels). We observe similar results on the image-text classification task in comparison to image-only ones. The data from **OpenI** dataset are from a different institute to the one of **MIMIC-CXR** data (our training data). Therefore, the domain gap shall be considered when we examine the performance. Table 6 additionally shows the AUCs for each disease findings. Indeed, the text report contains more information about the disease diagnosis (maybe more than the image itself). We also observe increase of the overall AUCs. In this case, our proposed meta-training with attention-on-label scheme also helps to boost the classification performance with a significant amount. As pointed out before, TieNet achieves better classification results in some of the categories since they adopted a more complicated text embedding network (hard to implement with no open code released) and we believe better results could be obtained if our learning process was applied on the same model.

AUC	MIMIC-CXR Test (NLP-label)			MIMIC-CXR 1k Test (hand-labeled GT)						
Disease	R50	CM	Ours	R50	negbio_u1	negbio_u0	chexpert_u1	chexpert_u0	CM	Ours
Atelectasis	0.985	0.986	0.981	0.962	0.914	0.832	0.908	0.829	0.958	0.96
Cardio.	0.946	0.953	0.949	0.865	0.821	0.805	0.822	0.814	0.871	0.862
Consol.	0.911	0.913	0.904	0.802	0.892	0.772	0.875	0.777	0.796	0.861
Edema	0.955	0.952	0.956	0.877	0.955	0.901	0.948	0.903	0.880	0.912
E-cardio	0.923	0.916	0.936	0.744	0.847	0.759	0.847	0.759	0.745	0.769
Fracture	0.935	0.766	0.876	0.671	0.733	0.689	0.747	0.718	0.746	0.769
Lesion	0.865	0.885	0.814	0.76	0.777	0.727	0.777	0.719	0.739	0.759
Opacity	0.969	0.968	0.967	0.877	0.877	0.861	0.871	0.863	0.890	0.880
no-finding	0.975	0.972	0.968	0.909	0.845	0.845	0.815	0.815	0.906	0.909
Effusion	0.974	0.973	0.974	0.923	0.941	0.906	0.940	0.913	0.925	0.931
pleural-o.	0.92	0.877	0.892	0.803	0.906	0.825	0.906	0.825	0.749	0.844
pneumonia	0.927	0.931	0.933	0.872	0.957	0.685	0.955	0.694	0.878	0.902
pneum-x	0.929	0.919	0.926	0.852	0.874	0.831	0.917	0.860	0.788	0.889
Devices	0.971	0.969	0.97	0.872	0.838	0.837	0.836	0.837	0.860	0.885
Average	0.941	0.927	0.931	0.842	**0.869**	0.805	**0.868**	0.809	0.837	**0.868**

Table 7: Classification AUCs for 14 findings in Chest X-Ray image and text report using the testing set of **MIMIC-CXR** (with NLP generated labels) and **MIMIC-CXR 1K hand-labeled Test** set. negbio_u1, negbio_u0, chexpert_u1, and chexpert_u0 are four sets of image labels from the original NLP mined labels with uncertainty. negbio_u1 is produced by setting uncertainty -1 to 1 and we set -1 to 0 for negbio_u0. Similiar settings for chexpert_u1 and chexpert_u0.

A.5   CLASSIFICATION RESULTS ON MIMIC-CXR 1K HAND-LABELED TEST DATA

Following the same labeling protocol proposed in Demner-Fushman et al. (2012), we randomly selected 1000 images and associated textual reports from the testing set of **MIMIC-CXR** dataset. One of our staffs (trained by a board-certified radiologist) hand-labeled the image by assigning the 14 labels manually to each image based on the associated reports. Identifying disease mentions in a report is a more accessible process comparing to examine the chest x-ray images to recognize the disease pattern in images, which will require years of professional training. The 14 labels adopted are identical to the NLP mined labels, i.e., Atelectasis, Cardiomegaly, Consolidation, Edema, Enlarged-cardiomediastinum, Fracture, Lung-lesion, Lung-opacity, No-finding, Pleural-effusion, Pleural-other, Pneumonia, Pneumothorax, and Support-devices.

As shown in Table 7, we observe the similar superior performance from our proposed method on the **MIMIC-CXR 1K hand-labeled Test** data. Note that the absolute accuracy is higher than the ones reported on **OpenI** dataset. It may indicate the domain gap between **MIMIC-CXR** and **OpenI** dataset.

Additionally, we evaluate the accuracy of NLP mined label sets, e.g., negbio_u1 and negbio_u0. negbio_u1 and negbio_u0 are two sets of image labels derived from the original NLP mined labels with uncertainty (labeled as -1). negbio_u1 is produced by setting uncertainty to 1 and we set -1 to 0 for negbio_u0. Two sets of NLP labels have quite different accuracy (6% gap) on the hand-labeled set. Similar findings can be observed between chexpert_u1 and chexpert_u0. It may be because the manual image labeling leans to the positive for those uncertain cases when our annotators will give 1 to those cases that the radiologist reported the observation of symptoms on the image (e.g., a density) while he/she is not sure what disease it is (e.g., it could be pneumonia, effusion, or consolidation). This aligns with the protocol used in Demner-Fushman et al. (2012). Our proposed method actually achieve a higher or similar accuracy as the NLP annotators while a model trained with NLP mined labels usually can not reach the same level of accuracy.

A.6 PSEUDO-CODE OF THE MAIN TRAINING FUNCTION

```python
def train(cls_model, bert_model, attn-on-label, train_loader, args):

    for batch_idx, (images, labels_negbio_u1, labels_negbio_u0,
                    labels_chexpert_u1, labels_chexpert_u0,
                    reports) in enumerate(tqdm(train_loader)):
        cls_model.train()
        bert_model.eval()
        attn-on-label.train()
        optimizer.zero_grad()

        # computing model prediction
        text_embedding = bert_model(reports)
        preds = cls_model(images, text_feat)

        label_list = [labels_negbio_u1, labels_negbio_u0,
                      labels_chexpert_u1, labels_chexpert_u0]
        feat_all = []

        # meta-training for each set of label
        for targets_m in label_list:
            meta_loss = binary_cross_entropy_loss(preds, targets_m)

            # Meta-training for classification model
            grads = get_grad(meta_loss, cls_model.parameters())
            cls_weights = cls_model.parameters() - args.meta_lr * grads

            # Computing Image Features using Meta-trained model
            feat_tmp = cls_model(inputs, text_feat, weights=cls_weights,
                    get_feat=True)
            feat_all.append(feat_tmp.detach())

        # attention-on-label module
        feat_all = stack(feat_all)
        attn_loss = softmax(attn-on-label.attn_fc(feat_all))
        # compute new label
        new_labels = mean(attn_loss * stack(label_list))
        # differentiable binarization
        new_labels = 1.0 / (1.0 + exp((new_labels - 0.1) * -50.0))

        class_loss = binary_cross_entropy_loss(preds, new_labels)

        # updating the attn-on-label parameters
        grads = get_grad(class_loss, attn-on-label.parameters())
        attn_weights = attn-on-label.parameter() - args.attn_lr * grad
        attn-on-label.attn_fc.weight.copy(attn_weights['attn_fc.weight'])
        attn-on-label.attn_fc.bias.copy(attn_weights['attn_fc.bias'])

        # update the classification model
        class_loss.backward()
        optimizer.step()
```

Listing 1: Train Function