Interpreting chest X-rays via CNNs that exploit disease dependencies and uncertainty labels

by Hieu H. Pham, Tung T. Le, Dat Q. Tran, Dat T. Ngo, and Ha Q. Nguyen

Medical Imaging Group

Vingroup Big Data Institute (VinBDI)

Hanoi, December 20, 2019

Table of contents

1. Problem

Problem

The CheXpert task was to build a supervised multi-label classification framework (e.g. CNNs) for predicting the risk of 14 common thoracic diseases.

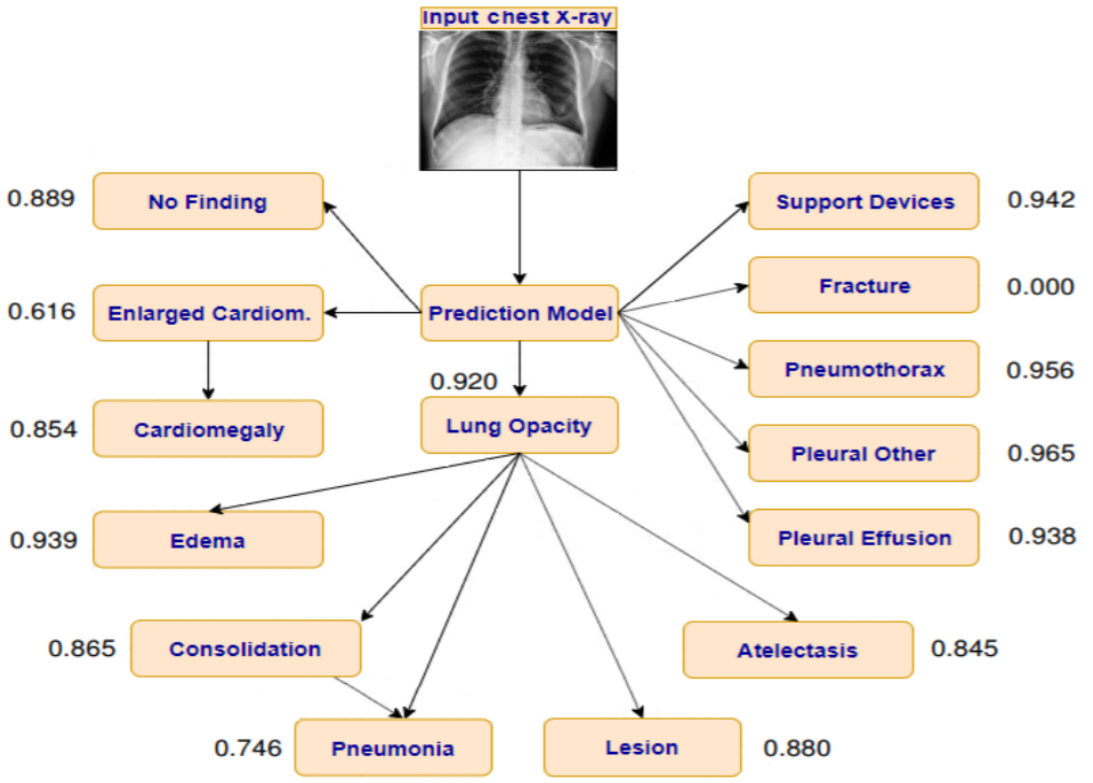


Figure 1: The 14 common thoracic diseases/observations provided by the CheXpert dataset (Irvin et *al*, 2019).

Training setting and evaluation protocol

Setting: Given a training set $\mathcal{D} = \left\{ \left(\mathbf{x}^{(i)}, \mathbf{y}^{(i)} \right) ; i = 1, \ldots, N \right\}$ that contains N CXRs; each input image $\mathbf{x}^{(i)}$ is associated with label $\mathbf{y}^{(i)} \in \{0, 1\}^{14}$.

Training: We train a CNN_θ that maps $\mathbf{x}^{(i)}$ to a prediction $\hat{\mathbf{y}}^{(i)}$ such that the cross-entropy loss function is minimized over the training set $\mathcal{D}$. The sigmoid activation function

$$\hat{y}_k = \frac{1}{1 + \exp(-z_k)}, \quad k = 1, \ldots, 14, \tag{1}$$

is applied to the logits z_k at the last layer of the CNN in order to output each of the 14 labels.

Evaluation: Model performance is measured by the AUC scores over 5 diseases: *Atelectasis*, *Cardiomegaly*, *Consolidation*, *Edema*, and *Pleural Effusion* from the validation set containing 200 studies.

2. Introduction to Hierarchical Multi-label Classification

Hierarchical multi-label classification

• In many real-world classification problems (e.g. gene functions, musical signals, or documents), the output labels are organized in hierarchy, which can be a tree (for text classification and bioinformatics) or a graph.

• Hierarchical Multi-label Classification (HMC) considers the structural information embedded in the class hierarchy, and uses it to improve classification performance.

• Traditional classification methods that do not take into account the dependencies among classes are called as “flat classification algorithms”.

Hierarchical multi-label classification

An example in Natural Language Processing (text understanding, topic modeling, etc): "**Sky**" and "**Cloud**" usually appear together, while "**Water**" and "**Cars**" almost never co-occur.

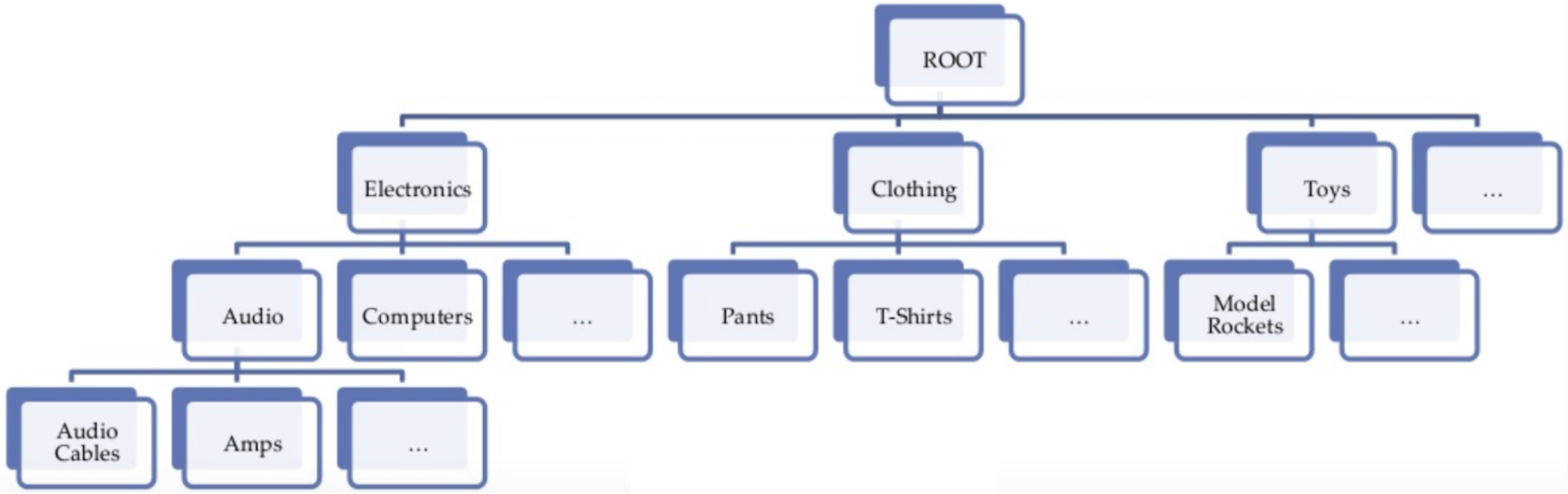


Figure 2: A hierarchy relationship among the words. The figure was taken from public domains.

Hierarchical multi-label classification

In medical imaging, labels are often organized into hierarchies in form of a tree or a graph. E.g. dependencies among labels from chest X-ray are considered as a tree[1].

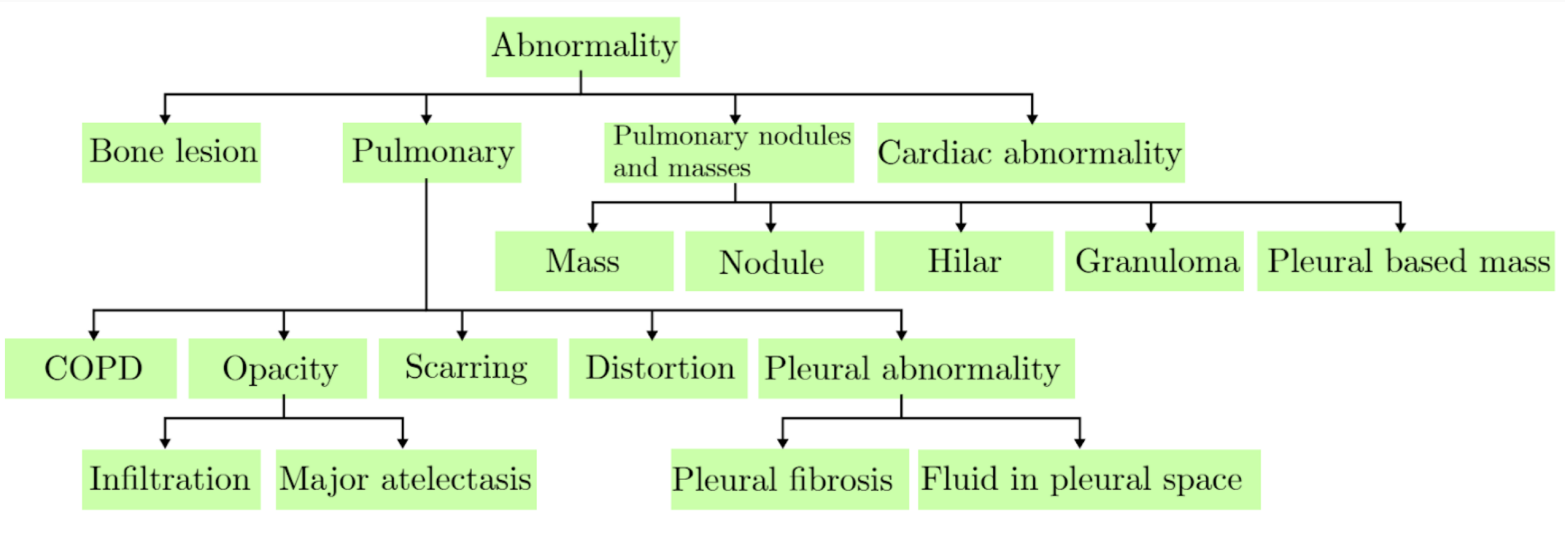


Figure 3: Constructed label hierarchy from the PLCO dataset (Chen et *al*, 2019).

[1] S. Van Eeden et al. "*The relationship between lung inflammation and cardiovascular disease*", American Journal of Respiratory and Critical Care Medicine 186 (1) (2012).

Hierarchical multi-label classification

Labels are often organized into hierarchies in form of a graph in the task of cell function prediction.

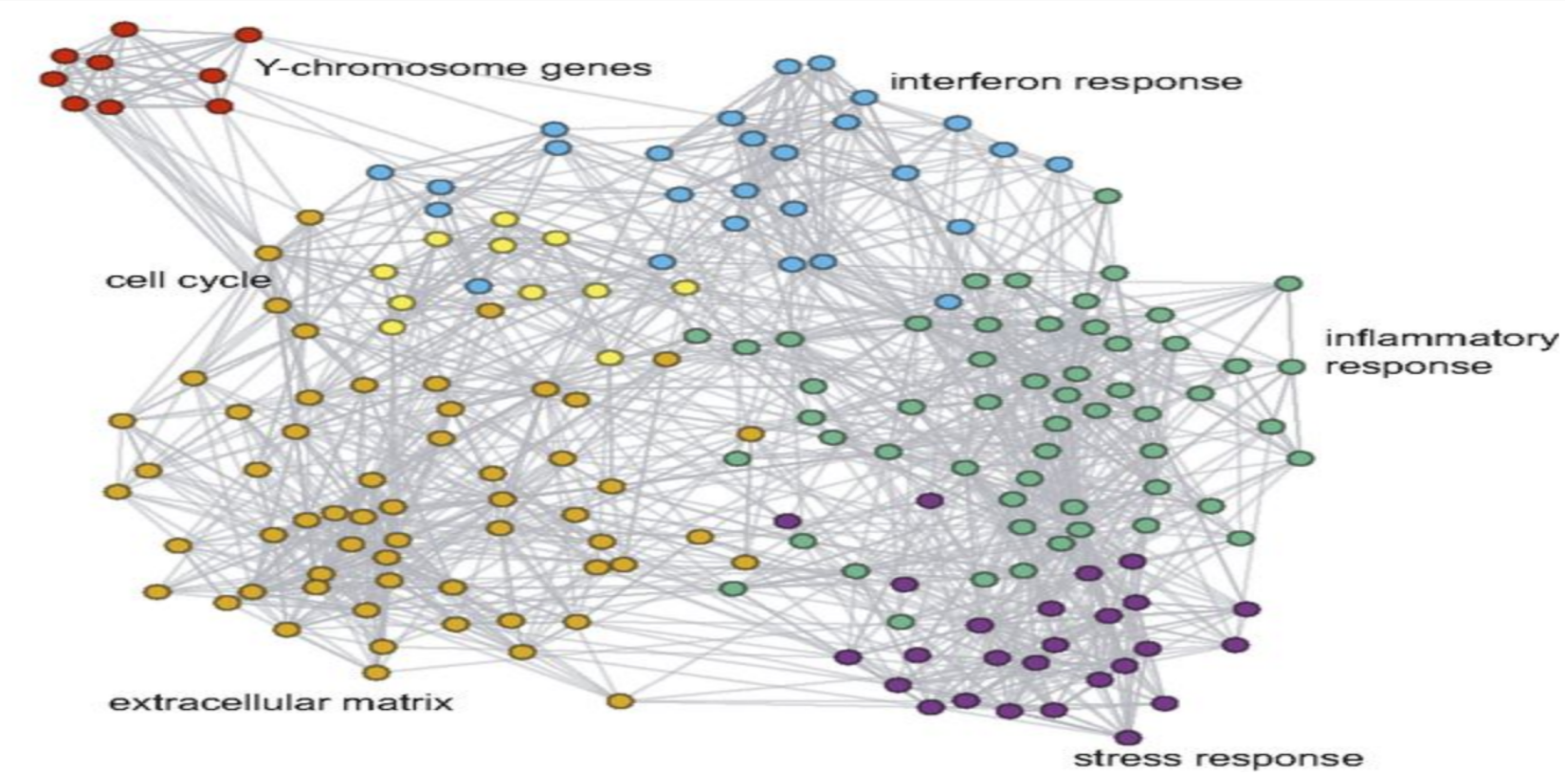


Figure 4: A hierarchy relationship among the cell functions.

Hierarchical multi-label classification

- Traditional methods to multi-label classification learn independent classifiers for each label or divide into multi-binary classifiers $\rightarrow$ they fail to explicitly exploit the label dependencies.
- DNN-based approaches are able to learn efficiently multi labels.
- The label hierarchy plays an important role for improving performance of ML/DL model in hierarchical multi-label classification task[2].

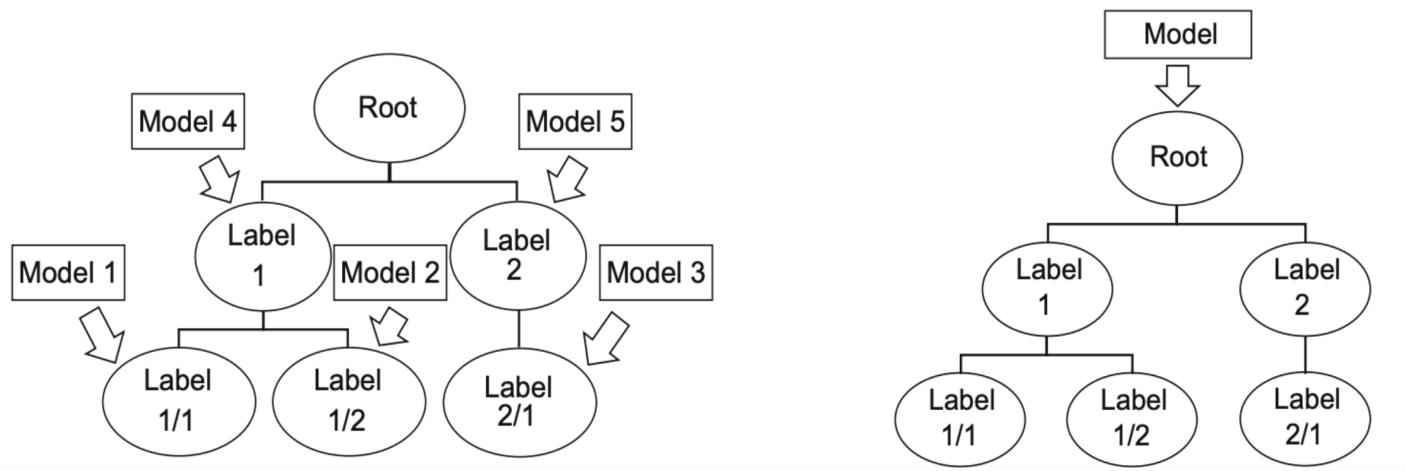


Figure 5: A hierarchy relationship among the cell functions.

[2] Levatić et al. "*The importance of the label hierarchy in hierarchical multi-label classification*" - Journal of Intelligent Information Systems 45.2 (2015): 247-271.

3. Exploiting Disease Dependencies and Uncertainty Labels for Model Development

Step 1: Conditional training

- Given a training set $\mathcal{D} = \left\{ \left(\mathbf{x}^{(i)}, \mathbf{y}^{(i)} \right) ; i = 1, \ldots, N \right\}$ that contains N CXRs; each input image $\mathbf{x}^{(i)}$ is associated with label $\mathbf{y}^{(i)} \in \{0, 1\}^{14}$.
- $\mathbf{y}^{(i)}$ can be represented via a tree $\mathcal{T}$.
- $\mathbf{y}^{(i)} = 1 \rightarrow \mathbf{y}^{(i)}_{\text{parent}} = 1$ for any non-root node $i \in \mathcal{T}$.

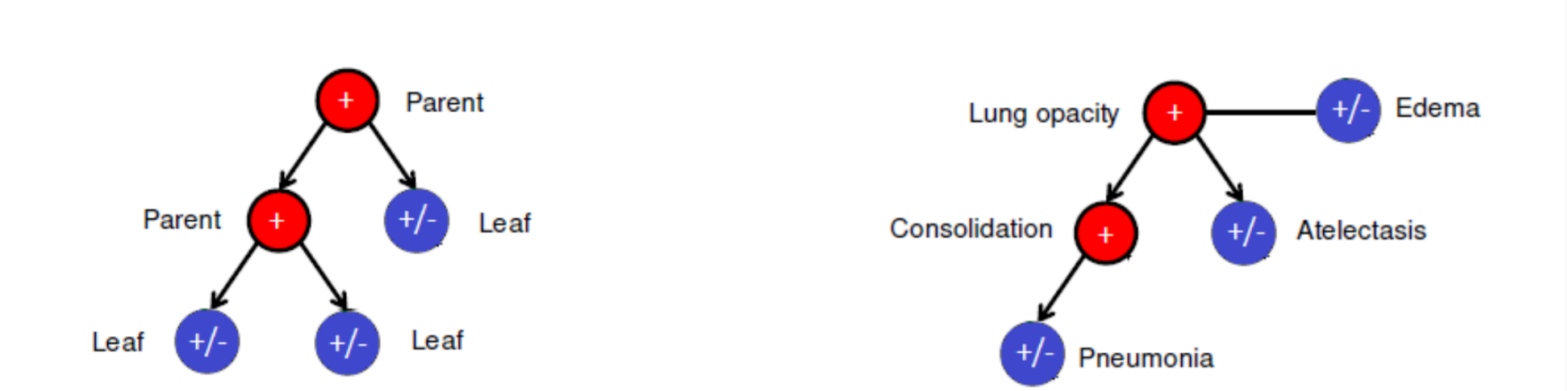


Figure 6: Illustration of the key idea behind the conditional training (**left**). In this stage, a CNN is trained on a training set where all parent labels (red nodes) are positive, to classify leaf labels (blue nodes). For example, we train a CNN to classify Edema, Atelectasis, and Pneumonia on training examples where both Lung Opacity and Consolidation are positive (**right**).

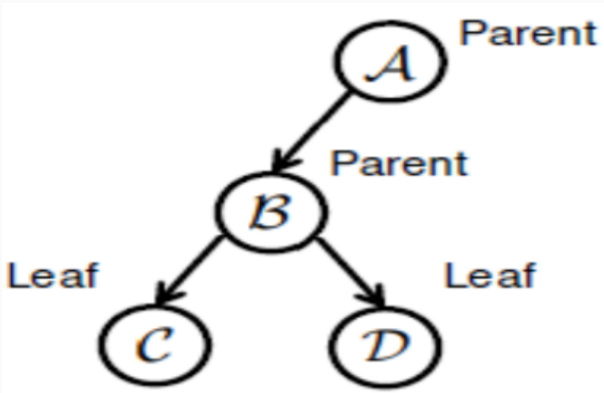
Step 2: Transfer learning

- Transfer learning will be exploited in the second training phase.
- All the layers of the pretrained network except the last fully connected layer are then freezeed and retrained on the full dataset.
- This training stage aims at improving the capacity of the network in predicting parent-level labels, which could also be either positive or negative.

Inference

- During the inference phase, all the labels should be unconditionally predicted.

- The unconditional probability of each label being positive should be computed by multiplying all conditional probabilities (i.e. the Bayes rule) produced by the CNN along the path from the root node to the current label.



$$p(\mathcal{C}) = p(\mathcal{A})p(B|\mathcal{A})p(\mathcal{C}|\mathcal{B}),$$

$$p(\mathcal{D}) = p(\mathcal{A})p(B|\mathcal{A})p(\mathcal{D}|\mathcal{B}).$$

Figure 7: An example of a tree of 4 diseases: $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$.

Leveraging uncertainty in CXRs with label smoothing

- The CheXpert labeler heavily depends on expert systems (i.e. using keyword matching with hard-coded rules), which left many CXR images with uncertainty labels $\rightarrow$ we may not have full access to the true labels.

- Several policies have been proposed in to deal with these uncertain samples, e.g. they can be all ignored (**U-Ignore**), all mapped to positive (**U-Ones**), or all mapped to negative (**U-Zeros**).

- We propose to apply a new advance in machine learning called label smoothing regularization (LSR)[3].

[3] R. J. Muller, S. Kornblith, G. E. Hinton, "*When does label smoothing help?*", ArXiv abs/1906.02629.

Leveraging uncertainty in CXRs with label smoothing

• We propose the **U-ones+LSR** policy that maps the original label $y_k^{(i)}$ to

$$\bar{y}_k^{(i)} = \begin{cases} u, & \text{if } y_k^{(i)} = -1 \\ y_k^{(i)}, & \text{otherwise,} \end{cases} \tag{2}$$

where $u \sim U(a_1, b_1)$ is a uniformly distributed random variable between a_1 and b_1 that close to **1**.

• Similarly, we propose the **U-zeros+LSR** policy that softens the **U-zeros** by setting each uncertainty label to a random number $u \sim U(a_0, b_0)$ that is closed to **0**.

4. Experimental Results and Perspectives

Experimental Results

Ablation study: The baseline DenseNet-121 model was trained with the U-Ones+CT+LSR policy and obtained an AUC of 0.894, which was a 4% improvement compared to the baseline trained with the U-Ones.

Table 1: Experimental results on the CheXpert dataset.

Method	Atelectasis	Cardiomegaly	Consolidation	Edema	P. Effusion	Mean
Ignore	0.768	0.795	0.915	**0.914**	0.925	0.863
Ignore+CT	**0.780**	**0.815**	**0.922**	**0.914**	**0.928**	**0.872**
U-Zeros	0.745	0.813	0.882	0.921	**0.930**	0.858
U-Zeros+CT	0.782	**0.835**	0.922	0.923	0.921	0.877
U-Zeros+LSR	0.781	0.815	0.920	0.923	0.918	0.871
U-Zeros+CT+LSR	**0.806**	0.833	**0.929**	**0.933**	0.921	**0.884**
U-Ones	0.800	0.780	0.882	0.918	0.920	0.860
U-Ones+CT	0.813	0.816	0.895	0.923	0.912	0.872
U-Ones+LSR	0.818	0.834	0.874	0.925	0.921	0.874
U-Ones+CT+LSR	**0.825**	**0.855**	**0.937**	**0.930**	**0.923**	**0.894**

Experimental Results

Our final model, which is an ensemble of six single models (DenseNet-121,-169,-201, Inception-ResNet-v2, Xception, and NASNetLarge), achieved an average AUC of 0.940.

Table 2: Performance comparison using AUC metric with the state-of-the-art approaches on the CheXpert dataset. The highest AUC scores are boldfaced.

Method	Atelectasis	Cardiomegaly	Consolidation	Edema	P. Effusion	Mean
Ignore-LP	0.720	0.870	0.770	0.870	0.900	0.826
Ignore-BR	0.720	0.880	0.770	0.870	0.900	0.828
Ignore-CC	0.700	0.870	0.740	0.860	0.900	0.814
Ignore	0.818	0.828	0.938	0.934	0.928	0.889
U-Zeros	0.811	0.840	0.932	0.929	0.931	0.888
U-Ones	0.858	0.832	0.899	0.941	0.934	0.893
U-MultiClass	0.821	0.854	0.937	0.928	0.936	0.895
U-SelfTrained	0.833	0.831	0.939	0.935	0.932	0.894
Ours	**0.909**	**0.910**	**0.957**	**0.958**	**0.964**	**0.940**

Experimental Results

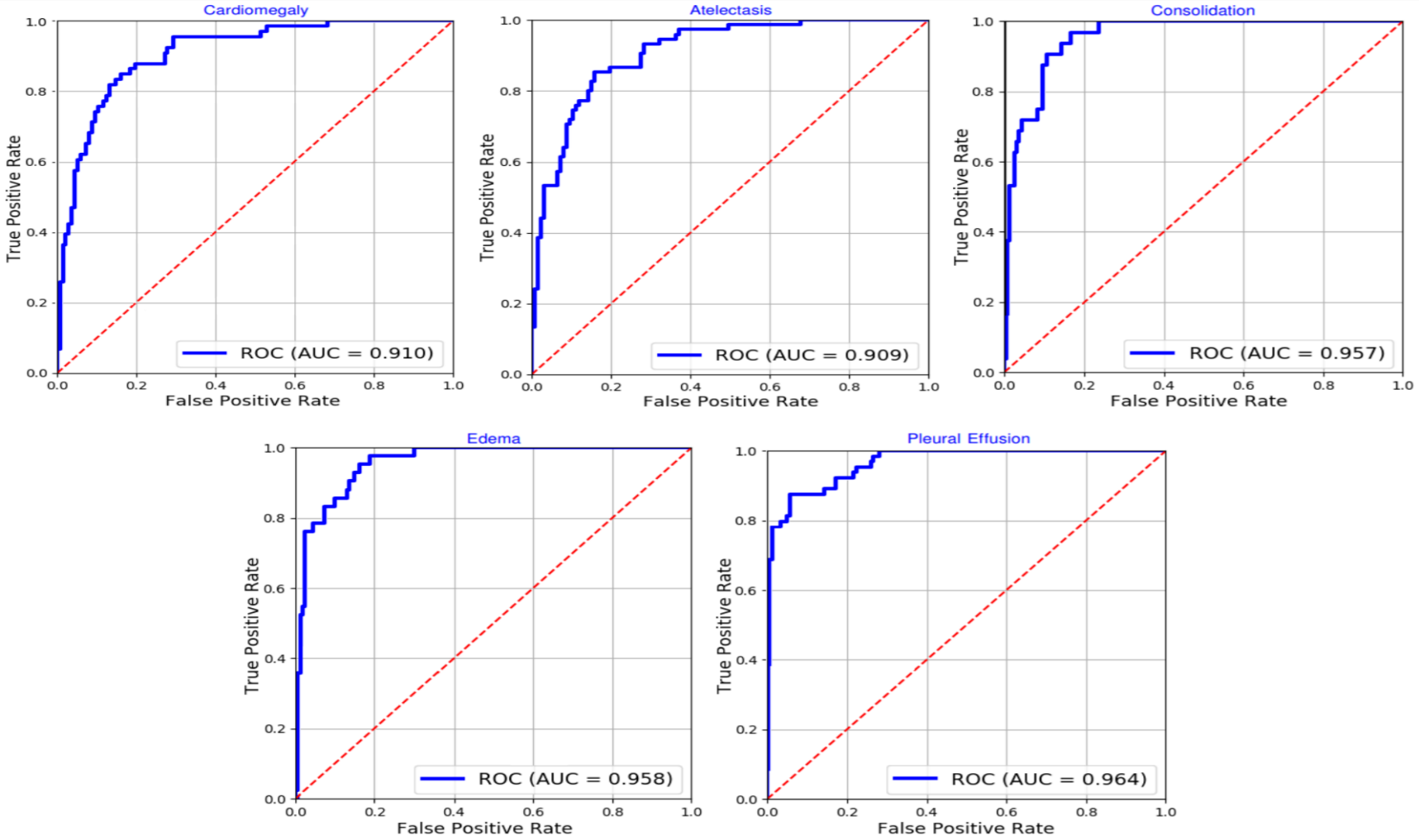


Figure 8: ROC curves of our ensemble model for the 5 pathologies on CheXpert validation set.

Experimental Results

Our model, namely Hierarchical-Learning-V1, currently takes the first place in the CheXpert competition.

Leaderboard

Will your model perform as well as radiologists in detecting different pathologies in chest X-rays?

Rank	Date	Model	AUC	Num Rads Below Curve
1	Sep 01, 2019	Hierarchical-Learning-V1 (ensemble) *Vingroup Big Data Institute* https://arxiv.org/abs/1911.06475	0.930	2.6
2	Oct 15, 2019	Conditional-Training-LSR *ensemble*	0.929	2.6
3	Dec 04, 2019	Hierarchical-Learning-V4 (ensemble) *Vingroup Big Data Institute* https://arxiv.org/abs/1911.06475	0.929	2.6

Figure 9: The CheXpert's leaderboard. Updated on December 20, 2019.

Labelling quality

A high rate of uncertainty samples found in the chexpert dataset.

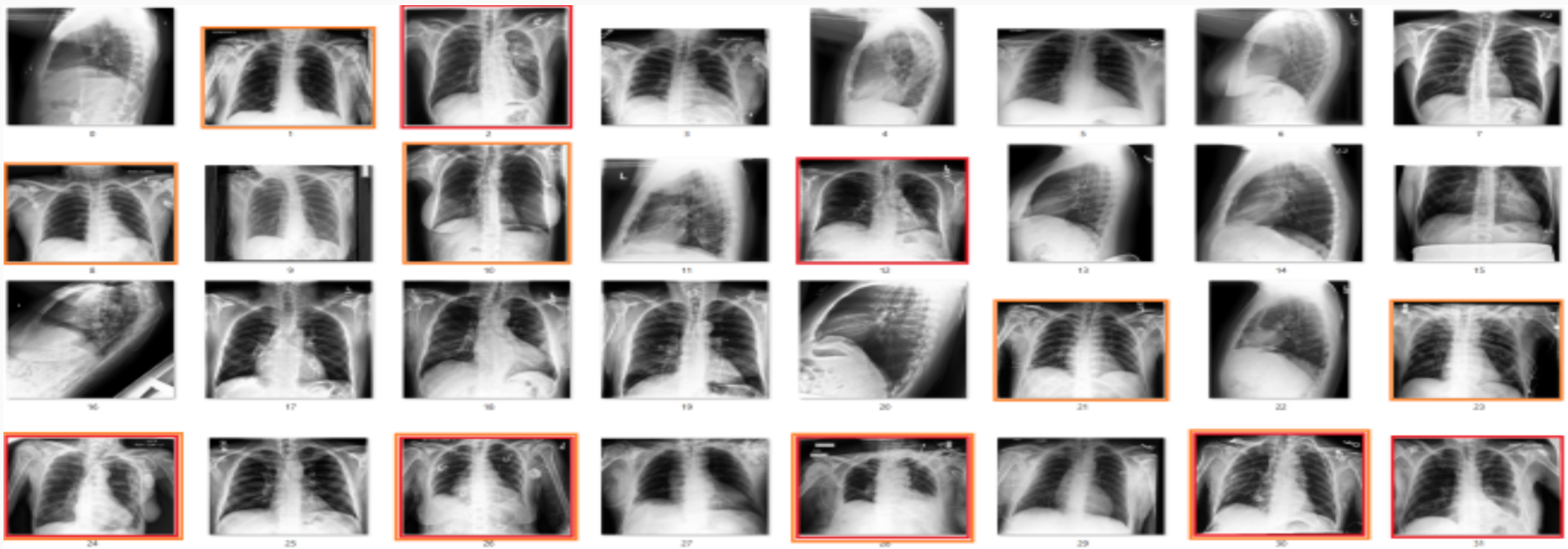


Figure 10: CheXpert's labeler annotates all these images as "No Finding". However, a radiologist tells that red highlights show cases containing pathology.

Result and Perspectives

Confident Learning: How to estimate uncertain labels from a dataset. Finding and learning with label errors in datasets?

- Pruning noisy data (Curtis G. Northcutt et *al*, 2017).
- Fixing label errors or modifying the loss function (Nagarajan Natarajan et *al*, 2013).
- Counting to estimate noise (George Forman *al*, 2007).
- Jointly learning noise rates during training, and ranking examples to train with confidence (Ibrahim et *al*, 2017).

Perspectives

- The current available chest X-ray datasets (Chest X-ray14, CheXpert, MIMIC-CXR) are not fit for training a medical AI system to do diagnostic work, we need more labeled data.

- How to deal with the geographic variation problem in chest X-ray datasets?

Table 2. Resulting AUCs for the 8 radiological findings common to the three datasets. Best results for each test set are in bold.

Test set	Training set	Atelectasis	Cardiomegaly	Consolidation	Edema	Lesion	Pneumonia	Pneumothorax	No Finding	Mean
ChestX-ray14	**ChestX-ray14**	**0.8165**	**0.8998**	**0.8181**	**0.9066**	**0.7935**	0.7633	**0.8796**	**0.7789**	**0.8320**
	CheXpert	0.7850	0.8646	0.7771	0.8584	0.7291	0.7287	0.8464	0.7569	0.7933
	MIMIC-CXR	0.8024	0.8322	0.7898	0.8609	0.7457	**0.7656**	0.8429	0.7652	0.8006
CheXpert	ChestX-ray14	0.5137	0.5736	0.6565	0.7097	0.6741	0.6259	0.7330	0.2682	0.5943
	CheXpert	**0.6930**	**0.8687**	**0.7323**	**0.8344**	**0.7882**	**0.7619**	**0.8709**	**0.8842**	**0.8042**
	MIMIC-CXR	0.6576	0.8197	0.7002	0.7946	0.7465	0.7219	0.8046	0.8564	0.7627
MIMIC-CXR	ChestX-ray14	0.5810	0.6798	0.7692	0.8098	0.6561	0.6740	0.7675	0.2562	0.6492
	CheXpert	0.7587	0.7650	0.7936	0.8685	0.7527	0.6913	0.8142	0.8452	0.7861
	MIMIC-CXR	**0.8177**	**0.8126**	**0.8229**	**0.8922**	**0.7788**	**0.7461**	**0.8845**	**0.8718**	**0.8283**

Thank you for your attention!