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

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

Medical Imaging Department

Vingroup Big Data Institute (VinBDI), Hanoi, Vietnam

Short paper #23, Medical Imaging with Deep Learning 2020

Problem

- **Problem** : Build a predictive model for diagnosing the presence of 14 observations in chest X-rays.

- **Proposed Approach** : Given a training set $\mathcal{D} = \left\{ \left(\mathbf{x}^{(i)}, \mathbf{y}^{(i)} \right) \right\}$ that contains N chest X-rays ; each input image $\mathbf{x}^{(i)}$ is associated with label $\mathbf{y}^{(i)} \in \{0, 1\}^{14}$. 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 $\mathcal{D}$.

- **Training and Evaluation** : The model was trained on CheXpert dataset (>235K chest X-ray scans) and evaluated on 200 studies over 5 diseases : *Atelectasis*, *Cardiomegaly*, *Consolidation*, *Edema*, and *Pleural Effusion* using AUC metric.

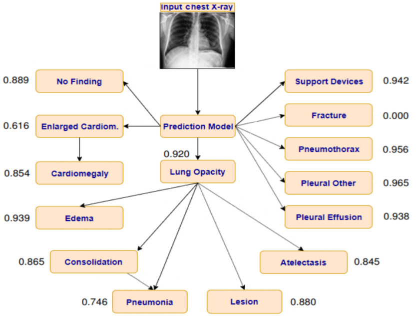

Fig.1 : Building a CNN-based model to predict the probability of 14 different observations from chest X-rays.

Exploiting disease dependencies and uncertainty labels

- Diagnoses or observations in chest X-ray are often conditioned upon their parent labels. This should be leveraged during the model training and prediction.
- For example, each input image $\mathbf{x}^{(i)}$ is associated with label $\mathbf{y}^{(i)} \in \{0, 1\}^{14}$ where $\mathbf{y}^{(i)}$ can be represented via a tree $\mathcal{T}$; $\mathbf{y}^{(i)} = 1 \rightarrow \mathbf{y}^{(i)}_{\text{parent}} = 1$ for any non-root node $i \in \mathcal{T}$.
- A CNN was pretrained on a partial training set containing all positive parent labels (*conditional training*), then retrained it on the full dataset (*transfer learning*).

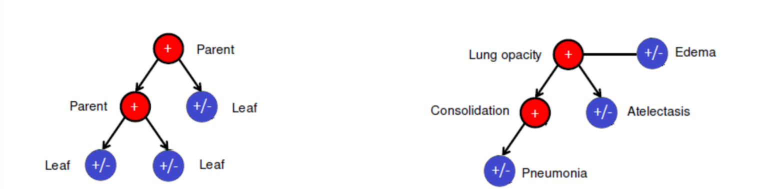

Fig.2 : A CNN was trained on a training set where all parent labels (red nodes) are positive, to classify leaf labels (blue nodes). For example, we train a CNN to classify *Edema*, *Atelectasis*, and *Pneumonia* on training examples where both *Lung Opacity* and *Consolidation* are positive.

Leveraging uncertainty in CXRs with label smoothing

- The chest X-ray labeler heavily depends on expert systems (*i.e.* using keyword matching with hard-coded rules), which left many chest X-ray 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 the `U-ones+LSR` policy that maps the original label $y_k^{(i)}$ to

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

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

- 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 $\mathbf{0}$.

Experimental results

We trained a strong set of six CNN models. Its ensemble model achieved an average AUC of 0.940, which set a new state-of-the-art result on CheXpert validation set and ranks first on the leaderboard of the CheXpert competition.

Table 1 – 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**