DenseNet Feature Embeddings for Thoracic Disease Diagnosis

Jimmy Wu [*1] **Fan Yang** [*1]

Abstract

We devise deep learning models for detecting 14 thorax diseases from chest X-ray scans at a level comparable to or exceeding recently proposed models as well as professional radiologists. In particular, we develop a 169-layer neural network based on the popular DenseNet convolutional architecture. We also extract and analyze feature vectors from this model and find that they exhibit significant clustering properties; we use this insight to develop other models that push classification performance even further.

1. Introduction

Chest X-ray scans are the most frequent type of radiology exam worldwide, and are commonly used to diagnose pneumonia, lung cancer, and dozens of other illnesses. Currently, the most effective method to diagnose pneumonia is chest X-rays (WHO, 2001). However, proper diagnosis is challenging, as a single scan can reveal multiple illnesses, and radiologists often disagree in their diagnoses.

In this work, we build deep neural models for multi-label classification of diseases revealed by chest X-rays. Given an image, our classifier outputs label(s) indicating which of 14 disease classes the image falls into. The output may be one, several, or none of the classes; the multi-label character of the task gives it considerable complexity.

1.1. Problem Formulation

Formally, the chest X-ray diagnosis problem is a multi-label, multi-class classification task. The input is a grayscale image $X \in \{0, \dots, 255\}^{224 \times 224}$, and for purposes of diagnosis, the output is a label vector $y \in \{0, 1\}^{14}$ where each coordinate $y_c = 1$ if and only if the disease c is present in image X. However when machine learning is applied to medical domains, it is customary to use instead an output vector $\hat{y} \in \mathbb{R}^{14}$, where $\hat{y}_c$ is the probability with which the model predicts disease c.

Additionally, we use our models to infer localized regions of the images in which the diseases are present, shedding light on which visual features are most pertinent for diagnostics.

2. Data

We use a dataset recently released by the National Institutes of Health (NIH) containing 112,120 X-ray images, each labeled with a subset of 14 potential diseases. These images were taken from 30,805 unique patients[1]. The diseases range from well-known ones such as pneumonia, emphysema, and edema, to less common pathologies such as pneumothorax.

Furthermore, though we do not use these features in our work, each image comes with a small amount of metadata, such as patient age and gender. A small fraction of the images also come with bounding boxes around diseased regions.

We preprocess the data by downsizing each image from 1024×1024 to 224×224. As in previous works on this dataset, and because our neural network will be pre-trained on ImageNet, we normalize each image with respect to ImageNet. Finally during training, we also randomly apply a horizontal flip to each image with probably $1/2$ to improve robustness.

3. Related Work

The NIH dataset has attracted strong attention in the health and AI communities. Wang et al. (Wang et al., 2017) were the first to perform classification on this data, using pre-trained imaging models such as `AlexNet`, `GoogleNet`, etc. They also generated heatmaps to indicate diseased regions. Huang et al. (Huang et al., 2017) trained a `DenseNet` model, which

[*]Equal contribution [1]Department of Computer Science, Stanford University, Stanford, USA. Correspondence to: Jimmy Wu <jimmyjwu@stanford.edu>, Fan Yang <fanfyang@stanford.edu>.

[1]Data available at `https://nihcc.app.box.com/v/ChestXray-NIHCC`.

was subsequently improved by Yao et al. (Yao et al., 2017). Finally, a very recent paper by Rajpurkar et al. (Rajpurkar et al., 2017) focused on classification of pneumonia, outperforming (Yao et al., 2017) and (Wang et al., 2017) on several disease types.

As we shall demonstrate, our work also focuses on embedding vectors extracted from deep neural networks. Similar studies in this regard include (Mikolov et al., 2013), which presents the Word2vec model for word representations, as well as the "negative sampling" method for training embeddings with useful geometric properties. Our work is also related to (Xie et al., 2016), which proposes a Deep Embedded Clustering model for clustering input points.

4. Methods

4.1. Transfer Learning with DenseNet

Our first effort is to develop a deep learning model for disease classification using the transfer learning paradigm. Starting with a standard DenseNet169 model (Huang et al., 2017) pre-trained on ImageNet, we replace the final fully-connected layer with one that has 14 sigmoid outputs. We then train this model end-to-end on the chest X-ray data.

For a *single* training example, our neural network models optimize the sum (over the classes) of the weighted binary cross-entropies, i.e.

$$-\sum_{c=1}^{14} w_c^+ y_c \log \hat{y}_c + w_c^- (1 - y_c) \log(1 - \hat{y}_c)$$

where c is the class index and $\hat{y}_c = \Pr(y_c = 1|X_c)$ is the predicted probability of the example having label c. $w_c^+ = \frac{|N|}{|P|+|N|}$ and $w_c^- = \frac{|P|}{|P|+|N|}$ are the fractions of negative and positive examples, respectively, in class c.

Of the models derived during 10 epochs of training, we save the one with the highest average AUROC over all the classes.

4.2. Feature Embeddings

In preparation for next steps in our project plan (see the following section), we perform some feature analysis on a trained CheXNet model, which is the name given to the model devised in (Rajpurkar et al., 2017)[2]. In particular, we look at the input to the final fully-connected layer of CheXNet, which is a 1024-dimensional vector $v^{(i)}$. We call this the *feature vector* or *embedding* for a given input $x^{(i)}$.

We build the set of all feature vectors $v^{(1)}, \ldots, v^{(m)}$ for a random subset of about 5% the dataset, then normalize them using the mean and variance of that set. We find that these vectors are fairly sparse: each vector has only about 667 non-zero coordinates. We then compute the following:

- The average pairwise L2 distance between vectors in this set.
- For each class $c \in \{1, \ldots, 14\}$ as well as the "no disease" class, the average pairwise L2 distance between vectors corresponding to examples from class c.

The result of this computation is shown in Table 1.

	L2	L2 square
Global average	**35.29**	**1334.34**
Atelectasis	33.31	1209.62
Cardiomegaly	34.15	1280.41
Effusion	35.22	1349.60
Infiltration	34.89	1300.12
Mass	38.65	1631.34
Nodule	37.76	1608.35
Pneumonia	35.99	1370.38
Pneumothorax	37.30	1522.33
Consolidation	33.35	1172.05
Edema	32.35	1096.89
Emphysema	38.67	1661.70
Fibrosis	33.78	1211.49
Hernia	29.28	903.76
No disease	33.19	1172.74

Table 1. Global and per-cluster average distances between pairs of vectors. Emphasized classes are the ones that relies outside of 95% confident region of Global distribution.

Although the intuition is complicated by the fact that this is a multi-label problem, these distances demonstrate that on average, feature vectors with the same labels are closer in 1024-dimensional space. In fact, under modest assumptions about the underlying data, this difference is statistically significant.

To see this, suppose each feature vector $v^{(i)}$ is a 1024-dimensional random variable in which each dimension t is drawn i.i.d. from a normal distribution, i.e. $v_t^{(i)} \sim \mathcal{N}(0, 1)$. In this case, for a random pair of vectors $v^{(i)}, v^{(j)}$ and each dimension t, we have

$$x_t = v_t^{(i)} - v_t^{(j)} \sim \mathcal{N}(0, 2)$$

[2]The model we use for this purpose is a fully pre-trained CheXNet model posted publicly to GitHub. We use this implementation rather than our own, because our implementation has not been trained long enough to be competitive with the best published models.

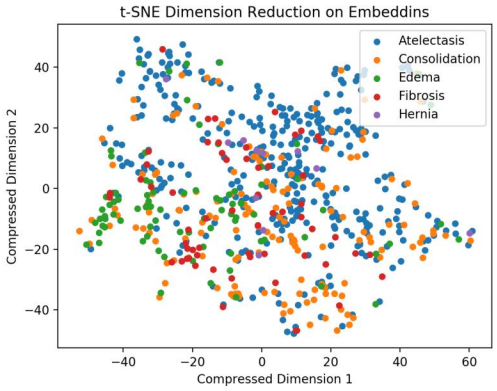

Figure 1. A projection of several disease class examples in 2D space, generated using the t-SNE dimension reduction technique.

as well as

$$\mathbb{E}[x_t^2] = \mathrm{Var}(x_t) + \mathbb{E}[x_t]^2 = 2$$

from which we deduce that

$$\mathrm{Var}(x_t^2) = \mathbb{E}[x_t^4] - \mathbb{E}[x_t^2]^2 = 4 \cdot 3 - 4 = 8.$$

Since the data in each dimension is independent, applying the central limit theorem on the dimensions implies that for the ≥ 667 nonzero indices, we have

$$\begin{aligned} &\frac{1}{\sqrt{667}} \sum_{t=1}^{667} \frac{x_t^2 - \mathbb{E}[x_t^2]}{\sqrt{\mathrm{Var}(x_t^2)}} \sim \mathcal{N}(0,1) \\ \iff &\frac{d^2(v^{(i)}, v^{(j)}) - 667 \cdot 2}{\sqrt{667} \cdot \sqrt{8}} \sim \mathcal{N}(0,1) \\ \iff &d^2(v^{(i)}, v^{(j)}) \sim \mathcal{N}(1334, (73.04)^2) \end{aligned}$$

where $d^2(v^{(i)}, v^{(j)})$ is the squared L2 distance between embeddings i and j. Therefore we would expect the global average squared L2 distance between vectors to be 1334.34, with a 95% confidence region of $[1217.47, 1451.20]$. However, as our calculations in Table 1 show, quite a few classes (those with names in bold) have average distances outside the confidence region, suggesting that the data does not share the same distribution as that of the dataset overall. This leads us to the intuition that feature vectors cluster by class.

4.3. Embedding-Based Classification

As demonstrated above and in Table 1, the feature vectors pulled from the DenseNet model form clusters, one per class, such that on average, distances within clusters are modestly smaller than distances over the whole dataset.

To take advantage of this observation, we extract the feature vectors from the DenseNet and use them as inputs to other classifiers. To exploit the geometric structure of the vectors, we use the k-nearest neighbors algorithm, in which a prediction on input X is made by taking a vote of the k *training* examples it is closest to. We also use other classic machine learning methods such as random forests, in which an ensemble of decision trees vote/average their predictions.

5. Experiments and Results

For the DenseNet169 model described in the previous section, we perform a thorough search over many hyperparameters; we settle on an Adam optimizer with standard parameters ($\beta_1 = 0.9$ and $\beta_2 = 0.999$) (Kingma & Ba, 2014), mini-batches of size 16, and a learning rate that begins at 10^{-4} and decays by a factor of 0.2 after each epoch in which the loss does not improve. To address overfitting, we add dropout at a rate of 0.1; we find that this works better than L2 regularization. This model achieves an average AUROC score (over all classes) of 0.842.

Next, we compute, for each input $X^{(i)}$, the feature vector $f(X^{(i)}) \in \mathbb{R}^{1664}$ obtained by running the input through all but the last layer of the DenseNet model. We then take these to be the inputs to several classic machine learning models:

- **k-nearest neighbors:** Since k-nearest neighbors scales poorly for a large number of high-dimensional examples, we subsample each class so that we retain only a 1% fraction of the original dataset, but each class is represented in the same proportion as in the original dataset. Our best choice of $k = 20$ nevertheless yields significantly poorer results than the DenseNet itself, at an average AUROC of 0.686.
- **Random forest:** This model is competitive with the DenseNet. For a choice of 100 individual trees in the forest, each with a maximum tree depth of 5 (to curb overfitting), the result is an average AUROC of 0.825.

Finally, we ensemble the DenseNet169 model together with the embedding-based random forest model (that is, at prediction time we take the average of their individual predictions), which results in even better scores for many classes.

Pathology	Rajpurkar et al.	DenseNet169	Random Forest	k-Nearest Neighbors	DenseNet+RF
Atelectasis	0.8094	0.8280	0.8267	0.7252	**0.8291**
Cardiomegaly	**0.9248**	0.9147	0.8909	0.6945	0.9117
Effusion	0.8638	0.8888	0.8820	0.8047	**0.8889**
Infiltration	**0.7345**	0.7201	0.7117	0.6598	0.7220
Mass	**0.8676**	0.8524	0.8645	0.6366	0.8517
Nodule	0.7802	0.7849	0.7741	0.5978	**0.7872**
Pneumonia	0.7680	0.7644	0.7603	0.6062	**0.7685**
Pneumothorax	0.8887	0.8816	**0.8974**	0.7594	0.8795
Consolidation	0.7901	0.8092	**0.8164**	0.6997	0.8101
Edema	0.8878	0.8941	**0.9071**	0.7805	0.8914
Emphysema	**0.9371**	0.9222	0.8883	0.7125	0.9310
Fibrosis	0.8047	0.8397	0.8029	0.6391	**0.8403**
Pleural Thickening	0.8062	0.7918	**0.8149**	0.6752	0.7940
Hernia	**0.9164**	0.8903	0.8454	0.6122	0.8712

Table 2. AUROCs for models trained in a recent paper (Rajpurkar et al., 2017), as well as ours. In all but five diseases, one of our models (either our DenseNet169 model, or our ensemble of the DenseNet169 with the embedding-based random forest) outperforms the earlier work, in some cases by large margins.

More detailed metrics for our models, as measured on the test set, are shown in Table 2.

6. Conclusion

In this work, we demonstrate improved deep learning models for diagnosing a myriad of thoracic diseases from chest X-ray scans. These diseases are often common and preventable, but have up to now required professional radiologists to diagnose. Our results show that these tasks can be at least partially performed by machine learning methods, which could potentially solve the problem of the global shortage of radiology experts.

In addition to improving classification performance for many disease classes, we also demonstrate that feature embedding vectors intercepted from deep neural models exhibit interesting geometric properties—namely clustering—and that this can be exploited for better accuracy.

6.1. Future Work

We believe the research direction studied here only scratches the surface of what is possible with embedding vectors. We plan to continue work on this project in the coming academic quarter, focusing on the following improvement areas:

- Learning a better embedding: Currently we extract our feature vectors from a DenseNet model trained with a final logistic classification layer. We are now beginning to try a new approach, using the negative sampling loss technique introduced by (Mikolov et al., 2013) to train feature vectors that have large separation between different classes. The hope is that this will enable even stronger clustering properties and thus better accuracy with methods such as k-nearest neighbors. However, currently the loss calculation is too slow; we are exploring ways to speed it up.

- We will attempt to use embeddings in a transfer learning situation—in particular, to detect a new disease for which we do not have enough data a priori. We plan to first train a neural network on 13 disease classes, then ask it to classify a new disease based on whether the embedding vector of a given input is far from the clusters for the pre-trained disease classes.

- Localization: We plan to use the bounding boxes provided in the dataset to generate visualizations of where our neural networks believe the diseases to exist in the patient's body. This will help us understand our models as well as possible ways to improve them.

- Data augmentation: Currently we simply resize each X-ray image to 224×224. We can instead augment the dataset by first resizing each raw image to 256×256, then taking various crops (either deterministically or randomly) to derive several different images of size 224×224.

- As a practical matter, we plan to use more powerful

hardware. We have provisioned a machine with 8 times more GPUs, which will allow us to iterate on our models much faster.

Code Repository

The code described herein is available at https://github.com/jimmyjwu/chest_X-ray_diagnosis.

Contributions

Jimmy was responsible for training and tuning the DenseNet169 model, preliminary analysis of feature embeddings, and applying non-neural models on the feature embeddings. Fan was responsible for statistically analyzing the embeddings, generating visualizations, and training a new feature embedding (this is ongoing work and does not appear in this report). Both Fan and Jimmy built infrastructure, reproduced earlier work, read papers, brainstormed ideas, and wrote this document.

Acknowledgements

We thank Aarti Bagul and Hao Sheng for their kind help and mentorship.

References

Huang, Gao, Liu, Zhuang, Weinberger, Kilian Q, and van der Maaten, Laurens. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, volume 1, pp. 3, 2017.

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

Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Greg S, and Dean, Jeff. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013.

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

Wang, Xiaosong, Peng, Yifan, Lu, Le, Lu, Zhiyong, Bagheri, Mohammadhadi, and Summers, Ronald M. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3462–3471. IEEE, 2017.

WHO. Standardization of interpretation of chest radiographs for the diagnosis of pneumonia in children. 2001.

Xie, Junyuan, Girshick, Ross, and Farhadi, Ali. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pp. 478–487, 2016.

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