Multiclass Classification Chest X-ray Using Ensemble Learning

Difan Gu, Jiaqi Luo, Xialin Liu, Yangrui Fan
Department of Computer Science,
University of Illinois at Urbana-Champaign, Urbana, IL 61801
{difangu2, jiaqi10, xialinl2, yangrui3}@illinois.edu
Presentation: https://mediaspace.illinois.edu/media/t/1_9emx8atv

Abstract

Deep learning evolving from traditional machine learning enables multi-layer data abstraction with more powerful processing resources; it is extremely suitable for learning the pattern of medical images. Chest X-ray is one of the most frequent diagnostic tools. Although many disease pathologies are obvious at first sight in the images, misdiagnosis is still prevalent and unavoidable since the manual screening on the X-ray is error prone. Deep learning applications can be helpful in both targeting the pathologies which need more attention and providing computer-aided detections. In this project, we are going to reproduce a baseline model CheXNet proposed in the Article "chest X-rays at a level exceeding practicing radiologist [1]", which is proposed by Stanford ML Group, the model successfully provides radiologist-level diagnosis Pneumonia using Chest X Ray.The model is also extended to classify multiple thoracic pathologies in the paper. But the precision is lower than predicting the single disease. In our work, we modified the CheXNet model to predict all 14 pathologies probabilities and improved the prediction accuracy for most of the diseases. There are several initiations worth to highlight in our multi-label training experiments: firstly, the input dataset is imbalanced in its labelling, the number of cases vary from 227 images to 19,894 images, to overcome this issue we carefully designed our loss function in order to adapt the training for all labels. Secondly, in order to improve the overall performance of CheXNet, we modified the original model architecture, with variant densely connected convolutional neural network models, such as DenseNet 161, DenseNet 169, DenseNet 201, During our comparison of the new models' testing results against the original CheXNet, we observed that different models are good at predicting different pathologies. This result inspired us to build our final ensemble neural networks with DenseNet, AlexNet, ResNet and VGG models, which shows a significant improvement by utilizing the wisdom of the crowd.

Keywords— Deep Learning, Convolutional neural networks, Chest X-ray, Model Ensemble

1 Introduction

1.1 Motivation

X-ray is one of the most popular diagnostic tools to detect varieties of complications for patients' health. It is unsurprising that the need for X-ray is enormous: there are approximately 3.6 billion diagnostic X-ray examinations performed each year across the world [2]. Example X-ray images shown in Fig 1. In order to infer the right diagnosis, an experienced radiologist needs to be trained with many years of experience, and the manual process to read the X-ray examinations also takes from 5 to 15 minutes. Most importantly, radiologists also may make misdiagnosis causing severe consequences not only to patients but also to already limited medical resources. Overall, X-ray is time consuming, error prone as well as requires highly trained doctors but only scarce talents are available.

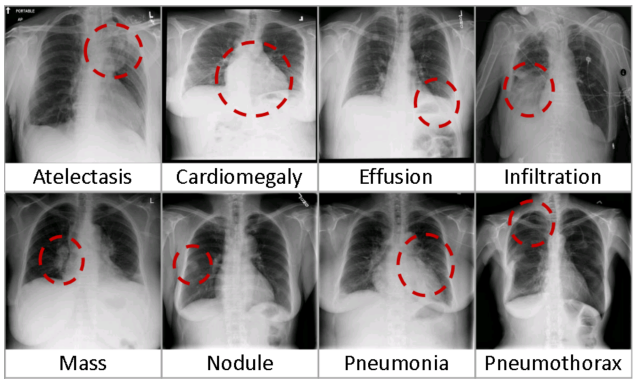

Figure 1. *X-ray images on different Pathologies [3]*

Fortunately, Deep Learning nowadays is able to land some help to radiologists, mitigating three issues mentioned above. First, many state-of-art Deep Learning models are capable of classifying millions of X-ray images within a relatively shorter period of time while requiring almost no manual cost to generate a result.

Moreover, in one of the studies conducted by Rajpurkar et al [4], they proposed a convolution neural network model CheXNet with radiologist-level performance, which can achieve more or less the same sensitivity and specificity for most pathologies' diagnosis and even more accurate than an experienced radiologist in diagnosis of Pneumonia, signalling the Deep Learning on chest X-ray develop towards a mature technology that can be applied in assisting clinical diagnosis. In Figure 2, the performance comparison between the algorithmic diagnosis versus the Resident and the Board-Certified radiologists has been demonstrated by the type of pathologies. On the other hand, Deep Learning models can accumulate knowledge infinitely in theory by constantly consuming data and learning from them to improve the performance. Therefore, a large amount of efforts have been devoted to developing and applying computer-aided detection(CAD) in chest X-ray radiographic images.

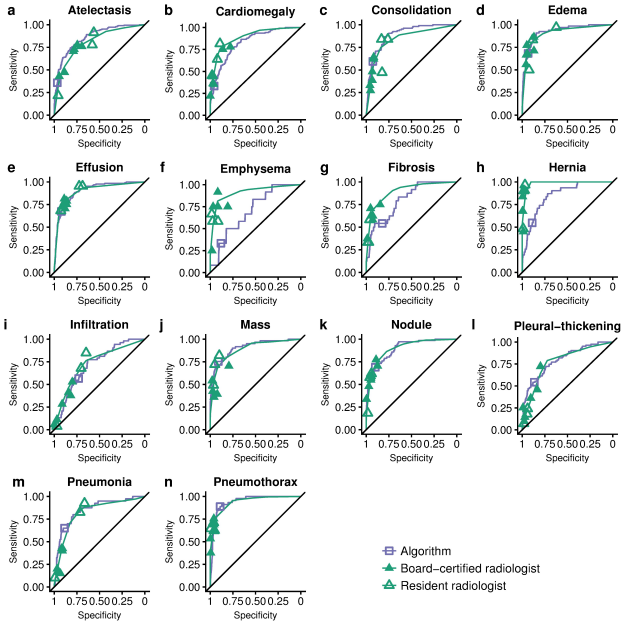

Figure 2. *Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNet algorithm to practicing radiologists [4]*

1.2 Literature Survey

1.2.1 Classification State-of-the-Art

As our study utilizes the ChestX-ray14 dataset from NIH, we start our literature survey from the state-of-art result published by NIH together with the dataset's first releasing, which is a weakly supervised classification and localization CNN model, produced by Wang et al (2017) [3]. This model uses the whole X-ray images for training without bounding box annotation. Instead, the CNN model modified the ImageNet model by adding a transition layer, a global pooling layer, a prediction layer, and a loss layer to target the local patch of diseases. After the original dataset ChestX-ray8 expanded to ChestX-ray14, Rajpurkar et al (2017) [1], constructed CheXNet, a 121-layer Dense Convolutional Network instead of using the whole image as input, CheXNet use downscaling images to 224x224 pixels but using image augmentation to add more transformed inputs for ImageNet. The pre-trained weights from ImageNet model are used as the initial value settings to replace the randomized weights, the output of CheXNet exceeds average performance in detecting pneumonia, with modifications on the original binary classification model, it also generally performs better in detecting all 14 diseases compared to previous models, thus achieves the SOAT result. CheXNet also outputs pathologies from the final convolutional layer to produce bounding boxes for diseased areas by upscaling the pathologies to the image dimensions. While both of these previously mentioned works are using a weakly supervised CNN framework, Guan et al (2018) [5] come up with an attention guided CNN model, AG-CNN, their model guided the CNN to target the lesion disease region by feeding both global image to CNN and also cropped attention region and feed it to local branch. The final Classification layer will consume both branches output to produce prediction. The final performance of the model compared to the previous state-of-the-art, improves in the localization but sacrifices the classification performance.

1.2.2 Localization State-of-the-Art

In the localization field, Candemir et al (2016) [6] structured an automatic organ localization model in chest x-days by comparing the input images' similarity with sample CXR images to locate heart and lung shadows without employing CNN framework. Rajaraman et al (2020)) [7], introduced a model using modality specific convolutional neural network ensembles, which guided the CNN model to train over wanted features only to better target the abnormal region in the X-ray images. This innovation also helps to make the trained model suitably repurposed from lung segmentation to detecting and localizing abnormalities. Given our study interest in this project focuses on classification problems, we will stop our survey for localization and segmentation SOAT works here.

In our project, we choose CheXNet as our baseline model and dive into its detailed architecture by reproducing Rajpurkar et al's work. CheXNet fine-tunes a pre-trained 121 layers' densely connected convolutional neural network (DenseNet). Instead of using the whole image as input, CheXNet uses downscaling images to 224x224 pixels but using image augmentation to add more transformed inputs for ImageNet. The pre-trained weights from ImageNet model are used as the initial value settings to replace the randomized weights, the output of CheXNet exceeds average performance in detecting pneumonia, with modifications on the original binary classification model, it also generally performs better in detecting all 14 diseases compared to previous models, thus achieves the SOAT result. As we look into CheXNet core components, DenseNet121, we learnt the fact that based on the testing results from DenseNet authors Huang et al [8] shown in table new., in terms of top-1 and top-5 error rates, DenseNet-121 is actually the model with the worst performance compared to the other DenseNet variations, our experiment evaluates the different performance outcome by modifying the layer structure of the DenseNet in the CheXNet. Throughout our tests, we track the prediction loss of the training and testing progress for CheXNet with fine-tuned DenseNet161, DenseNet169 and DenseNet201. We expect these variations to outperform the CheXNet original structure with

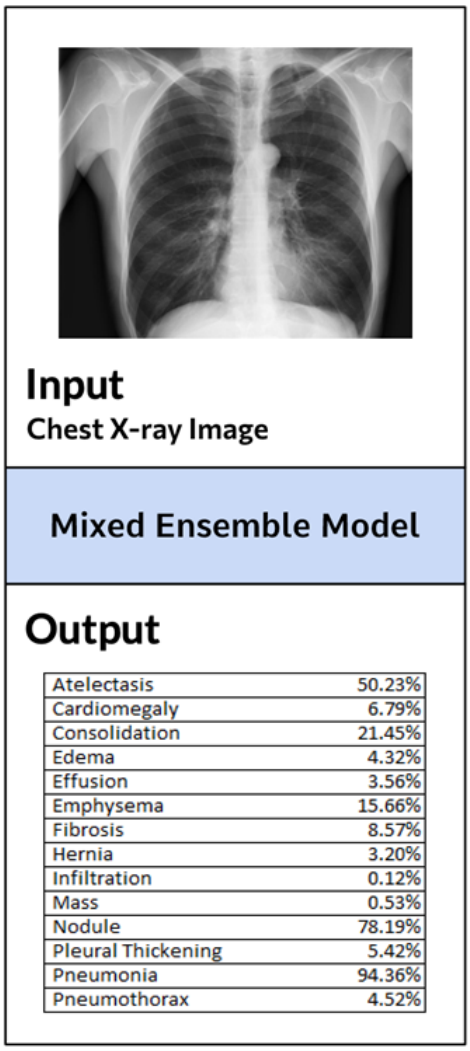

Figure 3. *Our proposed DenseNet Ensemble Model takes a chest X-ray image as input, and outputs a vector of probabilities if the patient is diagnosed positive for each of the 14 pathologies.*

DenseNet121 in the measure of AUC ROC. Our newly integrated CheXNet variation model is able to produce predictions for all 14 pathologies compared to the original CheXNet show in figure 3.

Model	top-1	top-5
DenseNet-121 (k=32)	25.02 (23.61)	7.71 (6.66)
DenseNet-169 (k=32)	23.80 (22.08)	6.85 (5.92)
DenseNet-201 (k=32)	22.58 (21.46)	6.34 (5.54)
DenseNet-161 (k=48)	22.33 (20.85)	6.15 (5.30)

Table 1. *The top-1 and top-5 error rates for different DenseNet models. [8]*

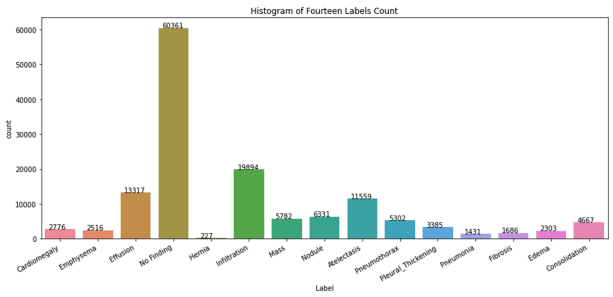

Figure 4. *Distribution of Labels count (including 14 pathologies and no finding)*

2 Approach/Metrics

2.1 Data Processing and Labelling

We collected our input data ChestXray14 from NIH Medical Center [1] [3] and performed a data exploration to extract fundamental descriptive statistics for a data quality diagnosis.

Measure Name	Value
Number of Images	112,120
Number of Pathology Labels	14
Number of Patients	30,805
Number of Image per Patient	4
Image Size	1143x1021 $\sim$ 3827x3567

There are several limitations that caught our attention: Firstly, the labels provided by NIH are produced by text mining from the X-rays' associated radiological reports using natural language processing. NIH also published a paper about this process, in which they provided the accuracy of the text mining outcome being greater than 90%. However, several researchers questioned the actual accuracy is below their statements. We have conducted searches through the Internet to find more reliable labels for ChestXray14 dataset and find that Stanford ML group has generated a modified version of the labelling file with the help of their own radiologists, but the revised labelling file is not released along with the publication of CheXNet. As the authors of CheXNet suggested, retraining the model with modified labels significantly changes the prediction result, we assume once a more accurate labelling representation becomes available to retrain our model [9], our prediction results will perform better in terms of reaching a more reliable clinical prediction result to assist human radiologists.

Secondly, the data is imbalanced in its labeling, even though the images' labels occur at the clinical prevalence stated by the author, the quantity ratio for disease labels varies significantly. The smallest group, Hermia, contains only 227 images, while the largest group, Infiltration, contains 19894 images. This high imbalance in our data collection brings a challenge in the following training stability and potentially harms the model evaluation if we blindly pick out our test datasets.

To accommodate the above addressed issue, we split the whole dataset (112,120 images) into roughly 70% training data (78,468 images), 10% validation data (11,219 images), and 20% testing

[1] https://nihcc.app.box.com/v/ChestXray-NIHCC

data (22,433 images). While the training data and validation data are randomly drawn from the pool, the testing data is drawn randomly first and then adjusted to enrich the test data for each label. The adjustment is performed to address the label imbalance issue in our input dataset ChestXray 14 and to ensure we have enough test data for each label. In all the pre-processing steps of our input data, we intentionally keep our splits not overlapping with each other.

2.2 Problem Definition

Given arbitrary Chest X ray frontal images **I** with size **H*W**, we design a convolutional neural network based model to output a multi-hot encoded vector with 14 dummy variables for the different considered abnormalities, **V** = [v_1, v_2, …, v_{14}], where $v_i \in \{0, 1\}$, denotes if the input image detected abnormal as the i^{th} pathology with value 1, or 0 otherwise.

In the training process, we applied weighted cross entropy loss with different abnormal classes. By comparing the class labels $\{b_1, b_2, \ldots, b_{14}\} \in \{0,1\}$ with our network prediction, we formed the following loss function:

$$\begin{aligned} L^{(c)}(I,c) = &- (w(c)_{present} \cdot b_c \cdot \log(p_c)) \\ &- (w(c)_{non-present} \cdot (1-b_c) \cdot \log(1-p_c)) \end{aligned} \tag{1}$$

$$L^{total} = \sum_{c=1}^{14} L^{(c)}(I,c) \tag{2}$$

where, $w(c)_{present} = \frac{P_c+N_c}{P_c}$, $w(c)_{not-present} = \frac{P_c+N_c}{N_c}$, with P_c representing the number of presented specific class c (c $\in$ [1,14]) input in the training dataset and N_c representing the number of input that are not labeled with class c in the training dataset.

In addition, the system is designed to output the final label vectors by ensembling the predictions from various DenseNet models (DenseNet121, DenseNet161, DenseNet169, DenseNet201) trained as multi-label classifiers. Each classifier will output a vector of probability of 14 pathologies, such as $p_{121}, p_{161}, p_{169}, p_{201}$. The final probability vector **P** will be computed based on simple average bagging shown in the following example:

$$P^i = \frac{1}{L} \sum_{l=1}^{l} p_l^i$$

, where $p_l^i \equiv l^{th}$ DenseNet output probability of the i^{th} pathology.

2.3 Base Model and Learning Setup

2.3.1 Model Architecture

We followed the basic framework of CheXNet and reproduced the model using NIH Chest X-ray dataset [3] by applying the PyTorch pretrained DenseNet121 as our initial status. We also modified the final linear layer to produce a probability vector. The network's parameter is optimized using the Adam algorithm with the hyperparameters beta1 set as 0.9, beta2 set as 0.999 respectively. The original CheXNet model proposed by Stanford ML Group [1] is trained using mini batches of size 16 and initial learning rate of 0.001. We used the prediction cross-entropy loss on the validation dataset as the main training metrics.

We apply the same methodology to fine-tune the variation of DenseNet, AlexNet, ResNet50 and VGG models and integrate them to the CheXNet framework separately. These models predict the probability vectors independently given a set of input X-ray images. The final probability prediction will be an ensemble outcome from all networks participation. The fig 5 shows the overall architecture of our framework.

2.3.2 Training

We trained our model on transformed images. As explained in the model architecture, we adopted pre-trained DenseNet, AlexNet, ResNet and VGG as our initial models in the frame, which all require input images normalized in the same way and set to required size. To accommodate the pre-training requirements, we downscaled each input image into the size of 256*256 and then performed TenCrop to cut the image into four corners and the central as well as the horizontal flipped version of these with size of 224*224. Finally, all images are normalized with PyTorch requested mean and standard deviation.

During the training process, for each epoch, we will only save the model which has a better validation loss. We closely monitor our model's loss trending, when there is a sign of plateaus in several epochs, we will tune our learning rate by replacing the original value with dividing by factor of 10. After training, we used AUROC to test our models and compare the performance on the 14 diseases.

We repeated the above process to train the groups of models separately with 70% of the original training data randomly drawn from the pool to guarantee our models independencies. In the training of our variation Dense Convolutional Network models with different numbers of layers, specifically, 161, 169 and 201, we observed as we train models with deeper layers, the gradient vanishing happens earlier in the training stage compared to 121-layers' DenseNet, we finally set the learning rate as 0.0001 for all variation DenseNet. For the other models, AlexNet, ResNet and VGG, we still apply the original learning rate 0.001 without any bottleneck.

2.3.3 Ensembling

What is Ensemble Method and Bagging (Bootstrap Aggregation)?

Ensemble machine learning model is not a new technique in the data science field. Over the past decades, lots of machine learning models have adopted the ensemble process to gain benefit of "wisdom of crowds" from diverse base architectures to enhance the overall performance. In the field of deep learning neural networks, the ensemble model brings even more benefits to the overall framework. It not only provides better performance but also helps to overcome one of the crucial downsides in all deep learning neu-

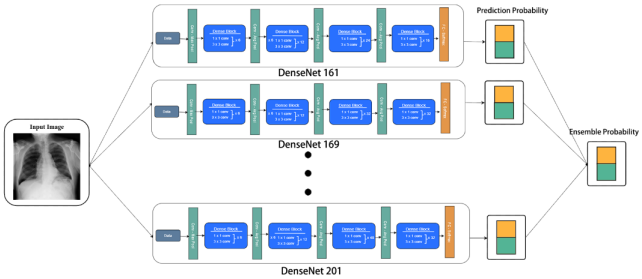

Figure 5. *Summary of the proposed ensemble classification methods. In the classification stage, ensemble is obtained by combining prediction probability from separate classifiers.*

ral networks, which is the high variance. Deep learning networks are nonlinear models and introduce flexibility in the training process but this is performed via a stochastic algorithm, which means the trained models highly rely on the input data pattern and are sensitive to data's specifics. Alternation in the input data can result in significant differences of the final model predictions.

Ensemble multiple deep learning neural networks as a single model can help in reducing the variance and instability of the neural networks to ensure our model can make reproducible predictions even in a clinical prevalence environment. To generate the ensemble member models, there are mainly three approaches, varying the training data used to train each model, varying the choice of each model to be trained, or a combination of both. In our work, we choose the second approach to produce our member models to be ensemble.

Regarding the ensemble methods, there are also different options. Bagging is one of them. The idea of bagging is to create many different learners from the same data set, and aggregates the predictions from each model to form a final prediction. Each learner specializes in different features, which enables bagging leverage predictions of every model to make best predictions.

Bagging has two main parts, bootstrapping and aggregation. Bootstrapping is to first use sampling with replacement to sample data from the whole dataset, then use the sampled subset of data to train the model. If we sample without replacement, each sample will not be independent with previous ones, which makes it non-random. Aggregation is to combine the predictions from all models and form a final prediction, which can be aggregated based on the counts of predicted output class or the probability of predictions from each model. [10]

Why do we want to ensemble our CNN models?

From the test report, we observe there is no single model that outperforms all the other ones when we evaluate their results using the same set of test dataset. In details, DenseNet161 provides better prediction in Cardiomegaly, Pneumonia, and DenseNet201 wins at prediction for Emphysema, Hernia. As for Mass and Atelectasis, we still can not beat the original CheXNet model. Therefore, the classic scenario of Bias and Variance Tradeoff exists here. Using an ensemble model by aggregating several predictions will dramatically reduce variance in the prediction, which is oftentimes treated as an archenemy in machine learning and especially in deep learning. In other words, it may be likely one of three models will be awfully wrong given a particular observation with a set of features, but it's very unlikely all three of them will be wrong at the same time. To take advantage of the outputs of different CNN models, we reduce variance with slight higher bias to achieve more stable performance. Our next step is to ensemble our trained classifiers to produce a better prediction model in all labels.

How do we perform our ensemble?

As shown in our proposed architecture summary (Fig 5.), we trained four different instances of each densely convolutional neural network and then got the trained DenseNet161, DenseNet169, DenseNet201 models to generate each predict on the test dataset. Given the DenseNet variation models only have four proposed structures, which is not enough to form a group to ensemble, we introduced a few other prevalent models, AlexNet, ResNet50 and VGG network that demonstrated good performance in Chest X ray analysis. As explored in the previous section, there are a number of methods that can be applied to ensemble the weak classifiers, such as bagging, boosting and stacked regression, etc. Considering our input dataset has significant imbalance labeling issues, we assume applying methods like stacked regression might produce a model with obvious bias. Thus our current ensemble stage uses preliminary linear averaging of the output prediction probability from the separate classifiers to produce the ensemble probability as the final prediction.

Our Implemented Group of Models

AlexNet [11]

AlexNet is one of the most influential convolution neural networks, which is widely used in Chest X-ray diagnosis as well. Given it's cheapness in the measure of run time, AlexNet requires the least amount of training time compared to the other pre-trained deep learning models, but still provides on par prediction results. Alexnet also does not request much extra augmentation of the input images. Nowadays, AlexNet is widely adopted in studies of Covid diagnosis using Chest X-Ray and provides almost 90% prediction accuracy rate.[11]

ResNet50 [12]

ResNet50 is another powerful convolution neural network architecture widely used in image classification, and has been used on Covid diagnosis recently [13]. A phenomenon researchers found is that when a model has too many layers, it is hard to train. This is called degradation, and is due to vanishing/exploding gradient, which the deeper the network is, the worse performance we get. ResNet overcomes the degradation problem by introducing a mechanism called Identity Mapping by Shortcuts, which allows input to skip k layer(s), and directly connect the input to $(k+1)^{th}$ layer without any transformation [12]. Compared with the DenseNet model, ResNet model needs less computational power required to train the model in terms of training time per epoch since it uses summation instead of concatenation like what's been adopted in DenseNet, yet it still achieves high prediction accuracy.

VGG16 [14]

In 2015 K. Simonyan and A. Zisserman from the University of Oxford proposed VGG16 in paper " Very Deep Convolutional Networks for Large-Scale Image Recognition". It is a famous convolutional neural network model, which uses multiple very small kernel-sized filters (3X3) to function as the large kernel-sized filters. It shows a good accuracy in image classification [14], especially over large images like X-Ray to perform classification and pattern recognition. In the study of Horry et al, they used a set of pre-trained models to detect positive Covid Cases using Chest X-ray and observed that VGG16 and VGG19 are the best models in the diagnosis[15].

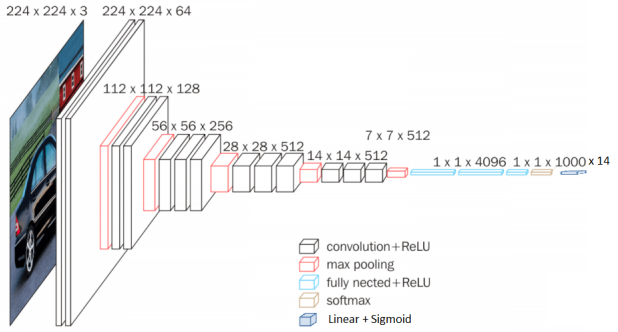

Figure 6. *VGG16 Convolutional Neural Network model Architecture [16]*

Layers	Output Size	DenseNet-121($k = 32$)	DenseNet-169($k = 32$)	DenseNet-201($k = 32$)	DenseNet-161($k = 48$)
Convolution	112 × 112	7 × 7 conv, stride 2			
Pooling	56 × 56	3 × 3 max pool, stride 2			
Dense Block (1)	56 × 56	1 × 1 conv 3 × 3 conv × 6	1 × 1 conv 3 × 3 conv × 6	1 × 1 conv 3 × 3 conv × 6	1 × 1 conv 3 × 3 conv × 6
Transition Layer (1)	56 × 56	1 × 1 conv			
	28 × 28	2 × 2 average pool, stride 2			
Dense Block (2)	28 × 28	1 × 1 conv 3 × 3 conv × 12	1 × 1 conv 3 × 3 conv × 12	1 × 1 conv 3 × 3 conv × 12	1 × 1 conv 3 × 3 conv × 12
Transition Layer (2)	28 × 28	1 × 1 conv			
	14 × 14	2 × 2 average pool, stride 2			
Dense Block (3)	14 × 14	1 × 1 conv 3 × 3 conv × 24	1 × 1 conv 3 × 3 conv × 32	1 × 1 conv 3 × 3 conv × 48	1 × 1 conv 3 × 3 conv × 36
Transition Layer (3)	14 × 14	1 × 1 conv			
	7 × 7	2 × 2 average pool, stride 2			
Dense Block (4)	7 × 7	1 × 1 conv 3 × 3 conv × 16	1 × 1 conv 3 × 3 conv × 32	1 × 1 conv 3 × 3 conv × 32	1 × 1 conv 3 × 3 conv × 24
Classification Layer	1 × 1	7 × 7 global average pool			
		1000D fully-connected, softmax			

Table 2. *Densely Connected Convolutional Networks architectures [8]*

DenseNet [8]

We used DenseNet variants including 121, 161, 169 and 201 to generate the prediction. As shown above, the DenseNet family shares a similar architecture. The only feature differentiating them is the complexity of Dense Block. As the model becomes more complicated, DenseBlock increases in size and thus the flexibility of the model significantly increases too.

However, when layers go deeper in a convolutional neural network, the problem of vanishing information may occur when information about input or gradient passes through many layers during training the model. In order to solve this, authors introduced an architecture with a simple connectivity pattern to ensure the maximum information flow between layers in the network by directly connecting a layer to all subsequent layers in such a way that each layer can obtains additional inputs from all of its preceding layers and passes its own feature-maps to all its subsequent layers. [8]

The network comprises L layers. Each layer ℓ implements a non-linear transformation $H_\ell(.)$, which can be a composite function of Batch Normalization (BN), rectified linear unit (ReLU), Pooling, or Convolution (Conv) . The feature-maps x received at layer ℓ will be a concatenation of the feature-maps produced in ℓ's preceding layers, in other words, $x_\ell = H_\ell([x_0, x_1, ..., x_{\ell-1}])$. [8]

When the size of feature-maps are not fixed, the concatenation may not work. However, the pooling layer, one of the key parts in the convolutional network, changes the size of feature-maps. To facilitate this, authors divide the architecture into multiple densely connected dense blocks, where the transition layers, which are layers between blocks, consist of BN layer, convolutional layer, and pooling layer.

3 Results

Table 3. *AUROC comparison for the Intermediate classifiers trained with 70% sample input data and the Ensemble Model*

	DenseNet161	DenseNet169	DenseNet201	ResNet50	AlexNet	VGG16	Ensemble Model
Mean	0.84595	0.8402	0.84464	0.84308	0.80830	0.82498	0.85356
Atelectasis	0.82557	0.82292	0.82759	0.83071	0.79873	0.81131	0.83520
Cardiomegaly	0.90858	0.89979	0.90596	0.90696	0.89669	0.88111	0.91719
Effusion	0.88446	0.88218	0.88316	0.88677	0.87250	0.88318	0.89048
Infiltration	0.71016	0.70606	0.71009	0.70860	0.69756	0.70682	0.71766
Mass	0.86241	0.85438	0.85901	0.86110	0.80002	0.84286	0.87013
Nodule	0.8063	0.79158	0.80122	0.79817	0.72154	0.77901	0.80744
Pneumonia	0.76818	0.75772	0.77633	0.76655	0.74017	0.74976	0.78068
Pneumothorax	0.87861	0.87661	0.87870	0.87568	0.86126	0.87399	0.89177
Consolidation	0.81167	0.8055	0.80996	0.81013	0.79779	0.80281	0.81919
Edema	0.89254	0.89209	0.89452	0.89800	0.88409	0.87796	0.90103
Emphysema	0.93650	0.93144	0.93221	0.93300	0.85804	0.91212	0.93594
Fibrosis	0.84733	0.83493	0.84698	0.85245	0.80813	0.80764	0.85396
Pleural_Thickening	0.79951	0.78235	0.78652	0.79380	0.74104	0.77083	0.79690
Hernia	0.91152	0.92532	0.91272	0.88124	0.83865	0.85024	0.93233

To apply the ensemble method, we trained DenseNet161, DenseNet169, DenseNet201, ResNet50 [13], AlexNet [11] and VGG16 [17] to build up our ensemble model. Each of the intermediate classifiers is trained by 70% of the training dataset by replacement sampling. Table 3 is showing the AUROC for the individual models' prediction alone and the ensemble model's prediction. The ensemble model shows a great improvement compared to each of the single convolutional neural networks on the average level and also improve the prediction accuracy for majority types of pathologies except Emphysema and Pleural Thickening.

Table 4. *AUROC comparison for DenseNet121, DenseNet161, DenseNet169, DenseNet201 and Ensemble model*

	DenseNet121	DenseNet161	DenseNet169	DenseNet201	Ensemble Model
Mean	0.85134	0.85195	0.84427	0.84358	0.85356
Atelectasis	0.83152	0.83086	0.82807	0.82668	0.83520
Cardiomegaly	0.91657	0.91712	0.90409	0.90587	0.91719
Effusion	0.88596	0.88645	0.88399	0.88419	0.89048
Infiltration	0.71183	0.71194	0.70591	0.70802	0.71766
Mass	0.86590	0.86577	0.85404	0.85713	0.87013
Nodule	0.80909	0.80982	0.79722	0.79838	0.80744
Pneumonia	0.78231	0.78345	0.76878	0.77179	0.78068
Pneumothorax	0.88118	0.88600	0.88349	0.87990	0.89177
Consolidation	0.81326	0.81292	0.81398	0.81383	0.81919
Edema	0.90290	0.90463	0.89701	0.89464	0.90103
Emphysema	0.93854	0.93942	0.92872	0.94041	0.93594
Fibrosis	0.85656	0.85854	0.84068	0.84840	0.85396
Pleural_Thickening	0.78789	0.78697	0.78403	0.79554	0.79690
Hernia	0.93528	0.93342	0.92984	0.94434	0.93233

Shown in Table 4, we also compared the AUROC between CheXNet models with the variant DenseNet layer structures and our final ensemble model. All CheXNet models with new DenseNet variations are trained with 100% training dataset, which ensemble model is ensemble by classifiers, which is trained by 70% training dataset (see in Table3). As Table 4 showed, among CheXNet models, the variation models with deeper layer DenseNet structures show improvement on predicting different pathologies. CheXNet with DenseNet161 slightly improves in the prediction of 4 pathologies and the one with DenseNet201 outperforms in the

diagnosis of Emphysema and Hernia. But it's significantly demonstrated in the above table, that our final Ensemble Model shows a better AUROC on the mean level and outperforms the other models in most of the pathologies. And for the type of pathologies where our Ensemble Model does not perform the best, its AUROC is very close to the winner in that group. From these numbers of facts, we believe the Ensemble Model we proposed is a better model compared to the CheXNet original model plus its variations in the multi-label prediction task.

4 Discussion

Challenges in Project Planning

Our proposed model met our initial set objectives to outperform the baseline model. To achieve this, we have to plan our tasks while exploring the possible directions and adjust our methodology carefully. From the initial reproduction stage, we break down the CheXNet architecture and follow the paper report to set our hyperparameters. As we learnt the core component of CheXNet is the densely connected convolutional neural network, we set our second stage as fine-tuning CheXNet by modifying its core component, from which we found promising results as the variations show slightly improved performance, but this still not close to our final objectives. After closely examining the evaluation measurements, we are inspired by the findings that improvements exist in multiple models, which lead us to settle down the final approach and work towards the creation of our ensemble model.

Challenges in Lab Environment Settings

In this project, we choose Google Colab as our coding platform to train and evaluate our models. Our input data in total takes up 60GB disk space, which is stored in our shared google drive. During our first trial, we unzipped all the images into the same folder and loaded batches of input from there, which permanently raised disk I/O error messages. After investigation, we found this was caused by having too many files located in the same folder. To resolve this issue, we reorganized our data into separate folders, more importantly, we also coded our DataLoader class CheXrayDataset accordingly to accommodate the new file structure.

As we start our model training using GPU, we apply the similar settings in our homework practice. However, it did not function properly under the Colab environment and our training tasks usually crashed for "RuntimeError: CUDA out of memory." which forces us to look for solutions to control the usage of memory during the training process. We learnt how tuning batch size helps to decrease the memory usage in detail and skipping track of gradients during the training helps to free space.

Challenges in Model Ensembling

We have performed two round model ensembles. In the first round, we tested using only DenseNet variation models trained with 100% training data, which has an average AUROC as 0.85787. In the final round, we added more models into the pool and all trained with 70% sampling data, the ensemble results' AUROC, 0.85356, turned a little bit lower as compared to the first round. We investigated each model's effect for the group prediction by applying different weights on the linear average ensemble calculation, but we did not find a member significantly pulling down the overall outcome. Finally, we target this decrease in AUROC is caused by our imbalance image label issue is not accommodated during the data sampling drawing stage. And from the result shown in Table 3, it's clear that most of our lower performance prediction diseases are the ones with relatively smaller total label count in the training data pool.

5 Conclusion

Throughout the study, we have fine-tuned and trained the majority of the convolutional neural network models which are popular in Chest X-ray related deep learning studies. In the stage of outcome evaluations, we found they all have unique specialties in extracting patterns for certain subset of pathologies. Specifically, AlexNet requires the least amount of running time to provide on par prediction outcomes; DenseNet models present high accuracy predictions for multiple pathologies with deeper network structure, surprisingly they are also quite efficient in the memory utilization during the training sessions, and the final trained models' sizes are compact as well. While VGG16 outputs on par prediction results, its trained model takes up to 6 times of the DenseNet trained models' size. ResNet50 shows on average performance in all measurements.

As discussed in earlier section 2.3.3 Ensembling, our goal is to design a model which is suitable for multi-label predictions and fine-tuning the existing prevalent CNN models doesn't grant us the breakthrough in terms of ROC AUC for the prediction of all pathologies. For example, we have seen some minor improvements in DenseNet 161 that have better ROC AUC in the measurement of average value, but we still have the improvements focusing on 4 out of 14 types of diseases we want to predict. It's similar for 169 and 201, while they can not outperform the original CheXNet even on the overall level. We change our perspective to see the question from another perspective: what if we can use the concept of bagging to combine multiple models together to generate a "democratic" decision by assigning weight, in our case we take a simple average, to increase the generality of the model. In Deep Learning, models have extremely complex architectures, as shown in Table2. Therefore, larger variance exists in each individual model.

Bagging is able to mitigate the problem by generalizing each prediction. Compared with the trained models with 70% of the training dataset by sampling with replacement, the full model generates a dominatingly better result in almost every single pathology except for Emphysema and Pleural Thickening by a small margin.

We also notice that those intermediate models achieve slightly worse prediction results compared to models trained with full data. This could be due to the fact that although training data, validation data, and test data are first randomly drawn, we did later adjust to handle label unbalanced issues. Such adjustment does not exist in bootstrapping in the bagging step when training the model, so it could be the potential cause of lower performance. Still, the model generates a dominatingly better result in most pathologies compared with the four DenseNet 121, 161, 169, 201 models trained with full data, and for those that do not perform better predictions, it is only off by a small margin. We can see that the ensemble model gives a higher stability, higher AUROC.

6 Contribution

Team Member	Performed Tasks
Xialin Liu	Source Data Exploration and Visualization; Fine-tune, Train and Test DenseNet201, AlexNet; Write Paper Report.
Yangrui Fan	Fine-tune, Train and Test DenseNet161, ResNet-50; Research and help implement Ensemble model; Write Paper Report.
Difan Gu	Presentation; Final project slides compilation; Ensemble researching; Write Paper Report
Jiaqi Luo	Research; Fine-tune, Train and Test DenseNet169, AGG16; Write Paper Report.

References

[1] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, *et al.*, "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning," *arXiv preprint arXiv:1711.05225*, 2017.

[2] C. Mitchell, "World radiography day: Two-thirds of the world's population has no access to diagnostic imaging," *Pan American Health Organization / World Health Organization*, Nov 2012.

[3] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "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, July 2017.

[4] P. Rajpurkar, J. Irvin, R. L. Ball, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. P. Langlotz, *et al.*, "Deep learning for chest radiograph diagnosis: A retrospective comparison of the chexnext algorithm to practicing radiologists," *PLoS medicine*, vol. 15, no. 11, p. e1002686, 2018.

[5] Q. Guan, Y. Huang, Z. Zhong, Z. Zheng, L. Zheng, and Y. Yang, "Diagnose like a Radiologist: Attention Guided Convolutional Neural Network for Thorax Disease Classification," *arXiv e-prints*, p. arXiv:1801.09927, Jan. 2018.

[6] S. Candemir, S. Jaeger, W. Lin, Z. Xue, S. Antani, and G. Thoma, "Automatic heart localization and radiographic index computation in chest x-rays," in *Medical Imaging 2016: Computer-Aided Diagnosis* (G. D. Tourassi and S. G. A. III, eds.), vol. 9785, pp. 302 – 309, International Society for Optics and Photonics, SPIE, 2016.

[7] S. Rajaraman, I. Kim, and S. K. Antani, "Detection and visualization of abnormality in chest radiographs using modality-specific convolutional neural network ensembles," *PeerJ*, vol. 8, 2020.

[8] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, 2017.

[9] L. Oakden-Rayner, "Chexnet: an in-depth review," *Luke Oakden-Rayner (PhD Candidate/Radiologist) Blog*, 2018.

[10] D. Sarkar and V. Natarajan, *Ensemble Machine Learning Cookbook: Over 35 practical recipes to explore ensemble machine learning techniques using Python*. Packt Publishing Ltd, 2019.

[11] T. Wang, Y. Zhao, L. Zhu, G. Liu, Z. Ma, and J. Zheng, "Lung ct image aided detection covid-19 based on alexnet network," in *2020 5th International Conference on Communication, Image and Signal Processing (CCISP)*, pp. 199–203, 2020.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[13] I. M. Baltruschat, H. Nickisch, M. Grass, T. Knopp, and A. Saalbach, "Comparison of deep learning approaches for multi-label chest x-ray classification," *Scientific reports*, vol. 9, no. 1, pp. 1–10, 2019.

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[15] M. J. Horry, S. Chakraborty, M. Paul, A. Ulhaq, B. Pradhan, M. Saha, and N. Shukla, "X-ray image based covid-19 detection using pre-trained deep learning models," *engrXiv*, Apr 2020.

[16] M. u. Hassan, "Vgg16 - convolutional network for classification and detection." https://neurohive.io/en/popular-networks/vgg16/, Nov 2018.

[17] C. Sitaula and M. B. Hossain, "Attention-based vgg-16 model for covid-19 chest x-ray image classification," *Applied Intelligence*, pp. 1–14, 2020.