Lung Disease Classification using Transfer Learning

Viraja Ketkar[1] and Sumaira Afzal[2]

Abstract— **The infection of lung that causes inflammation in one or both of the lungs is one of the most serious causes of chronic lung disease, it is imperative to detect that disease to make the recovery successful. When interpreting the x-ray, the radiologist looks for white spots in the lungs (called infiltrates) that identifies an infection. This exam also helps determine if a patient has any complications as abscesses or pleural effusions (fluid surrounding the lungs). Such condition can be detected by medical imaging methods like chest X-ray, CT of the lungs, MRI of the chest, needle biopsy of the lungs. Since subject of medical image mining is currently an up and coming topic and shows a lot of research potential in the area of computational intelligence.By auto-analyzing a patients medical images through data mining and image processing techniques, the risk of human error in disease detection can be reduced. With machine learning application , the earlier identification of diseases, particularly lung disease, we can be helped to detect earlier and estimate with high accuracy rate. In diagnosis from X-ray data, the physician can diagnose a part of the patient's medical condition, so with the X-ray chest image data, the intelligent machine can support the physician in the diagnosis of the disease to increase the accuracy of this system. We are so grateful to the scientists who have gone ahead in their research to offer to humanity, applying machine learning to the problem of X-ray image prediction. This work presents a solution using Convolutional Neural Networks (CNNs).The proposed system represents a very simple yet effective way of boosting the performance of trainedCNNs by composing multiple CNNs, with different CNN topologies along with different learning parameter sets. We are very fortunate to know that there is a huge set of X-ray image data released by NIH which can be accessed through Kaggle.**

Index Terms computational intelligence, machine learning, infiltrates, classification, estimation, accuracy rate, topology, learning parameters

I. INTRODUCTION

While lung disease is rare at the beginning of the 20th century, the frequency has increased steadily in parallel with the increase in smoking habits and has become the most common type of cancer in the world. According to the World Health Organization (WHO) report, lung disease is the leading cause of death among males and the second type of cancer among females all over the world .The main diagnostic methods of lung disease(especially cancer) are; a) chest X-ray, b) CT (Computerized Tomography), c) Endoscopic Ultrasonography, d) Bronchoscopy, Thoracoscopy, Dermoscopy. e) Pulmonary Radiography f) Positron Emission Tomography (PET) g) Magnetic Resonance Imaging. In some cases, however, the disease may not be detected by imaging techniques. This leads to delays for biopsy and the later start of treatment.

We will conduct a study and analysis of this data set, then apply Deep Learning to predict that the patient has a lung disease. The difficulty is a new dataset, and we are thankful to the pioneers to perform analysis on this large dataset which has never been processed full. Data has a lot of noise, and X-ray of the lung is not likely to provide enough information to assess whether a patient may be ill. Our key point here will be: combining the processing of patient information with data from X-rays, using CNN with the well-known pre-trained models like ResNet and Vgg16 network with hyperparameter tuning for this form of data. The architecture of the VGG16 convolutional neural network is trained to distinguish pixels across images, and can be utilized in our case infiltrate information. Our project will demonstrate that by leveraging these techniques, we substantially increase the sensitivity to detect infiltrates in lung , without inflating the false positive rate. Thus, from the available NIH X-ray dataset consisting of around more than 5000 images, we have compared the pre-trained Resnet and VGG model for feature extraction feature classification. In the first section of the study, basic information on entry and lung disease was included. In the second section, data set and materials used in the study were mentioned. In the third chapter, the transfer learning of the convolutional neural networks used in the study is briefly explained(ResNet,Vgg16) after that we begin our data analysis followed by data-pre-processing, model creation , training and testing. In the fourth section, the estimation results obtained from the pre-trained model are presented. In the fifth chapter, which is the last chapter, the results are evaluated.

II. MATERIALS

The lack of large publicly available datasets with annotations means it is still very difficult, if not impossible, to achieve clinically relevant computer-aided detection and diagnosis (CAD) in real world medical sites with chest X-rays. One major hurdle in creating large X-ray image datasets is the lack resources for labeling so many images. This NIH Chest X-ray data set is comprised of 112,120 X-ray images with disease labels from 30,805 unique patients. To create these labels, the authors used Natural Language Processing to text-mine disease classifications from the associated radiological reports. The labels are expected to be greater than 90 percent accurate and suitable for weakly-supervised learning.

Class descriptions

There are 15 classes (14 diseases, and one for "No findings"). Images can be classified as "No findings" or one

*This work was not supported by any organization

[1]Viraja Ketkar is student of School of Continuing Education, York University, Toronto, Canada ketkarviraja@gmail.com

[2]Sumaira Afzal is student of School of Continuing Education, York University, Toronto, Canada sumaira.afzalm@gmail.com

or more disease classes:

- Atelectasis
- Consolidation
- Infiltration
- Pneumothorax
- Edema
- Emphysema
- Fibrosis
- Effusion
- Pneumonia
- Pleural-thickening
- Cardiomegaly
- Nodule Mass
- Hernia

III. METHODOLOGY

A. Deep learning architecture

A deep learning architecture that has been studied extensively is the Convolutional Neural Network (CNN) which is a multi-layered image classification technique that incorporates spatial context and weight sharing between pixels. A CNN learns the optimal image features for a specific image classification problem by adopting an effective representation of the original image. Inspired by the process of visual perception in human beings, it requires little to no preprocessing. The basic components of a CNN are stacks of different types of specialized layers (convolutional, activation, pooling, fully-connected, softmax, etc.) that are interconnected and whose weights are trained using the backpropagation algorithm. The deepest layers of the network function as low-level feature extractors. The training phase of a CNN requires huge numbers of labelled data to avoid the problem of overfitting; however, once trained, CNNs are capable of producing accurate and generalizable models that achieve state-of-the-art performance in general pattern recognition tasks. Some examples include LeNet, the first CNN proposed to classify handwritten digits; AlexNet, a deep network designed for image classification; ZFNet, a newer model that outperforms AlexNet; VGGNet , which increases depth using 3 x 3 convolution filters; GoogLeNet, which includes inception modules (which is a new organizational structure); and ResNet, a residual network that is much easier to optimize than VGGNets. The CNN architecture and some cited examples are discussed in more detail.

B. Deep learned features

CNNs are a class of deep feed-forward neural networks. Like most neural networks, CNNs are composed of interconnected neurons that have inputs with learnable weights, biases, and activation functions.

CNN layers have neurons arranged in three dimensions: width, height and depth. This means that every layer in a CNN transforms a 3D input volume into a 3D output volume of neuron activations. CNNs are built with five classes of layers: convolutional (CONV), activation (ACT), pooling (POOL), followed by a last stage, including Fully-Connected (FC), and classification (CLASS).

The CONV layer is the core building block of a CNN and is also what makes CNNs so computationally expensive. These layers compute the outputs of neurons that are connected to local regions by applying a convolution operation to the input. The spatial extent of connectivity of these local regions is a hyperparameter called the receptive field, and a parameter sharing scheme is used in CONV Layers to control the number of parameters. This means that the parameters of CONV layers are shared sets of weights (also called kernels or filters) that have relatively small receptive fields.

POOL layers perform non-linear downsampling operations. Max pooling is the most common non-linear operation: it partitions the input into a set of non-overlapping rectangles and outputs the maximum for each group. In this way POOL reduces the spatial size of the representation while simultaneously reducing 1) the number of parameters, 2) the possibility of overfitting, and 3) the computational complexity of the network. It is common practice to insert a POOL layer between CONV layers.

ACT layers apply some activation function, such as the non-saturating ReLU (Rectified Linear Unit) function or the saturating hyperbolic tangent , or the sigmoid function .

FC layers have neurons that are fully connected to all the activations in the previous layer and are applied after CONV and POOL layers.

In this work, we test and compare the following CNN architectures:

ResNet : this is the winner of ILSVRC 2015. This network is approximately twenty times deeper than AlexNet and eight times deeper than VGGNet. The main novelty of this CNN is the introduction of residual (RES) layers, making it a network-in-network architecture. ResNet uses special skip connections and batch normalization, and the FC layers at the end of the network are substituted by global average pooling. Instead of learning unreferenced functions, ResNet explicitly reformulates layers as learning residual functions with reference to the layer inputs. As a result, ResNet is much deeper than VGGNet, although the model size is smaller and thus easier to optimize than VGGNet.

VGGNet : this is a CNN that placed second in ILSVRC 2014. The two best-performing VGG models(VGG-16 and VGG-19), with 16 and 19wt layers, respectively, are available as pretrained models. Both models are very deep and include 16 CONV/FC layers. The CONV layers are extremely homogeneous and use very small (33) convolution filters. A POOL layer is inserted after two or three CONV layers (instead after each CONV layer as is the case with AlexNet).

The large training data increases the learning effectiveness of a CNN. One effective way to expand training data is Data augmentation when necessary and to reduce overfitting during CNN training by artificially expanding the training set. Data augmentation applies transformations and deformations to the labeled data, thus producing new samples as additional training data. A key attribute of the data augmentation process is that the labels remain unchanged after applying the transformations. In this work we perform random data augmentation with horizontal and vertical flipping, rotation range, translation of pixels, and scaling.

We fine-tune the weights of the pretrained CNNs by fixing the deep CONV layers of the network and by fine-tuning only the higher-level FC layers since these layers are specific to the details of the classes contained in the target dataset.

C. Data Exploration

We explored how many unique labels are there? looked at the label distribution and converted the series to dataframe for plotting purposes.

Further enhancing the pre-processing , defined dummy labels for one hot encoding - removing no finding and simplifying to 14 primary classes, then performed One Hot Encoding of Finding Labels to dummy-labels.Examined how many cases present for each of our 14 clean classes (which excl. 'No Finding'),get sorted value-count for clean labels.

TABLE I
VALUE COUNT FOR CLEAN LABELS

Infiltartion	19894.0
Effusion	13317.0
Atelectasis	11559.0
Nodule	6331.0
Mass	5782.0
Pneumothorax	5302.0
Consolidation	4667.0
Pleural-Thickening	3385.0
Cardiomegaly	2776.0
Emphysema	2516.0
Edema	2303.0
Fibrosis	1686.0
Pneumonia	1431.0
Hernia	227.0

D. IMAGE PRE-PROCESSING

Using keras image preprocessing we preprocessed image. (Keras documentation:https://keras.io/pre-processing/image/) To perform significant image augmentation,we created ImageDataGenerator utilizing most of the parameter options to make the image data even more robust.

TABLE II
IMAGE PRE-PROCESSING PARAMETERS

	Simple-CNN	Resnet	Vgg16
rescale	1./255		
shear-range	0.1	0.1	0.1
zoom-range	0.15	0.15	0.15
rotation-range	5	5	5
width-shift-range	0.1	0.1	0.1
height-shift-range	0.05	0.05	0.05
horizontal-flip	True	True	True

To generate batches of augmented/normalized data. We set following parameters for flow-from-dataframe() for training and validation.

Table II , Table III and Table IV show the parameters for image-processing, data augmentation/normalization and model-creation/training parameters respectively.

TABLE III
AUGMENTING/NORMALIZED DATA PARAMETERS TO FLOW-FROM-DATAFRAME()

	Simple-CNN	Resnet	Vgg16
image-size	128,128	128,128	224,224
color-mode	gray scale	gray scale	gray scale
training-batch-size	32	32	32
validation-batch-size	256	256	256
test-batch-size	1024	1024	1024

TABLE IV
MODEL CREATION/COMPILATION/TRAINING PARAMETERS

	Simple-CNN	Resnet	Vgg16
loss	B-crossentropy	B-crossentropy	B-crossentropy
optimizer	adam	adam	adam
metrics	accuracy	accuracy	accuracy
epochs	5	2	1
steps-per-epoch	50	15000/160	15000/160
validation-steps	20	5000/1280	5000/1280
Activation Function	softmax	softmax	softmax

IV. EVALUATION MATRIX

Any typical medical image analysis system is evaluated by using different key performance measures such as accuracy, F1-score, precision, recall, sensitivity, specificity and dice coefficient. We have set the performance measure as accuracy.

V. RESULTS

The accuracy of the model training and the results obtained for different epochs are given in Table 5.

TABLE V
RESULTS

Architecture	Epoch	Time Elapsed	Loss	Acc	val-loss	val-acc
Simple-CNN	5	67s	4.17	0.1925	4.16	0.18
RESNET	2	452s	4.75	0.83	0.76	0.84
VGG16	1	2769s	6.55	0.84	1.68	0.84

VI. FINE TUNING

We think with this project there may be more to do to increase our results like:

- Training with more epoch, change some parameters to faster convergence models such as learning rate.
- Increasing the size of training shots, this will increase the chance of getting important features, but it also means that the model will be more complex and the training time and prediction will be longer.
- For optimized VGG, we can experiment on ensembles of pre-trained models.
- Some metric parameters of the metrics will also be tested more.

- For these models we want to try adding some more layers so that it can extract more features, but the training time will be very long.

VII. CONCLUSIONS

In order to evaluate the results obtained from the pre-trained models designed in the study, it is seen that only few epochs are used in, when the training process is examined first. At the end of each epoch, the error rate obtained by writing the lowest error rate shows an accuracy rate of 84 percent after first few epochs. However, this accuracy is the lowest rate of those epochs. When more epochs are considered in this case, it is seen that the average accuracy rate can be maximized. This accuracy is an indication that the model is successful. In the obtained error value, it is possible to say that the first epoch is a very high error , but it is approaching zero by decreasing to the next epoch. Another important factor is the length of the education process. The training ended after at different time span at the end of these epochs. This value may vary depending on the power of the processor, the size of the image, the number of images in the data set, and the number of convolutional and maxpooling layers used in the models. In fact, the number of filters used in the convolutional layer and the filter size affect both accuracy and training duration. The best solution is to have a complex CNN with the following insights: Research for resolved issues, domain information, support data, methods, and solution data for similar projects. Some potential techniques are listed and investigated. Sample data is downloaded and analyzed, preprocessing, metric selection Testing multiple architectures, optimizing and testing on a sample dataset. Use good architects to test the full dataset, continue optimizing and statistics. This project is based on a very new set of data and not many people find out, this is a very good problem and if done well it will make a big contribution to the community. This project has tested many new and interesting methods such as RESNET and VGG have shown that they have recorded remarkable results. X-rays are difficult to see clearly, the data is not standardized, and NLP labeling can be used to obtain the disease. It is also difficult to apply a very new method without much documentation to optimize it. Big data on the full dataset is also a big challenge for us being limited to computer power. The results of this project have achieved our initial expectation, but to be able to apply in hospitals, more improvements are needed to increase the precision of the model.

ACKNOWLEDGMENT

We would thank our course instructor Karthik Kuber for his support during this project. We would also thank the researchers who shared their research.Lastly we will thank the people who made the code available for reference like Kaggle and stack-overflow.

REFERENCES

[1] H. Greenspan, B. van Ginneken, R. M. Summers, Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique, IEEE Transactions on Medical Imaging 35 (5) (2016) 11531159.

[2] Y. Liu, H. Cheng, J. Huang, Y. Zhang, X. Tang, J.-W. Tian, Y. Wang, Computer aided diagnosis system for breast cancer based on color doppler flow imaging, Journal of medical systems 36 (6) (2012) 39753982.

[3] S. Shi, Q. Wang, P. Xu, X. Chu, Benchmarking state-of-the-art deep learning software tools, in: Cloud Computing and Big Data (CCBD), 2016 7th International Conference on, IEEE, 2016, pp. 99104.

[4] M. M. Rahman, B. C. Desai, P. Bhattacharya, Medical image retrieval with probabilistic multi-class support vector machine classifiers and adaptive similarity fusion, Computerized Medical Imaging and Graphics 32 (2) (2008) 95108.

[5] A. A. Saraiva , D. B. S. Santos , Nator Junior C. Costa , Jose Vigno M. Sousa , N. M. Fonseca Ferreira , Antonio Valente and Salviano Soares, Models of Learning to Classify X-ray Images for the Detection of Pneumonia using Neural Networks

[6] Deep Learning with Python by FRANOIS CHOLLET

[7] https://www.kaggle.com/kmader/train-simple-xray-cnn

[8] https://www.kaggle.com/cpagel/adjust-simple-xray-cnn

[9] https://www.kaggle.com/nih-chest-xrays/data

[10] https://www.kaggle.com/adamjgoren/nih-chest-x-ray-multi-classification