Minh Son Nguyen

Project Report

Chest X-Ray Pneumonia Detection

1. Problem Definition

According to CDC, pneumonia kills more children younger than five years old each year around the globe than any other type of infectious disease, such as HIV infection, malaria, or tuberculosis.

In the United States, more than 250,000 people must seek care in a hospital due to pneumonia each year. Unfortunately, about 50,000 people die from the disease each year in the United States.

Bacterial and viral infections are the two leading causes of pneumonia but require very different forms of treatment. Bacterial pneumonia requires urgent referral for immediate antibiotic treatment, while viral pneumonia is treated with supportive care to get better on its own. So accurate and timely diagnosis is crucial. One key component of diagnosis is radiographic images as chest X-rays are routinely obtained as standard care. However, expeditious system of interpreting images is not always available, especially in the some developing countries in South Asia and sub-Saharan Africa where childhood pneumonia has the highest incidence and highest rates of mortality.

The challenge here is to build a machine learning model to determine if a patient is suffering from pneumonia or not by examining chest X-ray images and furthermore to give doctors more time to distinguish viral and bacterial pneumonia to speed up referrals for children who need immediate intervention. The model should produce very precise results because we do not want children lives to be severely affected by wrong prediction.

2. The Dataset

The problem input is the dataset containing over 5,800 chest X-ray images divided into training set and test set. Images are labeled as normal and pneumonia infected individuals, and pneumonia infections are labeled as bacterial and viral related. Chest

X-ray images were selected from pediatric patients of one to five years old from Guangzhou Women and Children's Medical Center, Guangzhou.

The data source: https://data.mendeley.com/datasets/rscbjbr9sj/2

The output as discussed in the background will be prediction of the condition of the patient, normal or pneumonia.

3. Model Architecture

3.1 Data Preparation

First, a data processing function is created in the file `data_processing.py` to generate training and test data images into batches together with resizing the data. The `ImageDataGenerator` class from Keras is used to make these data generation objects, one for the training set and one for the test set. It provides the ability to use data augmentation automatically when training a model.

The class is instantiated and the configuration for the types of data augmentation are specified by arguments to the class instructor. A range of techniques are supported for this data augmentation, here images in the training data are flipped via `vertical_flip` argument, together with rescaling by a factor of 1/255.

Next, an iterator is created for an image dataset located on disk in a specified directory ('data/'), where images in that directory are organized into subdirectories according to their class ('train' and 'normal'). This is done by calling function `flow_from_directory of ImageDataGenerator` class. Then I feed into these functions specified batch size and target image size. Here I choose batch size of 32 and image size of 150x150.

In order to make predictions on the test set, I run two loops over the test data folder and resize and rescale the images and get the test data and test labels. This is useful for getting the accuracy score as well as the confusion matrix afterwards.

3.2 The convolutional model

The convolutional model is built based on the input layer, five convolutional blocks, a fully connected layer, and an output layer. This is done with the function in `cnn_model` the file `cnn_model.py`

The input layer is used to feed the input with shape 150x150x3.

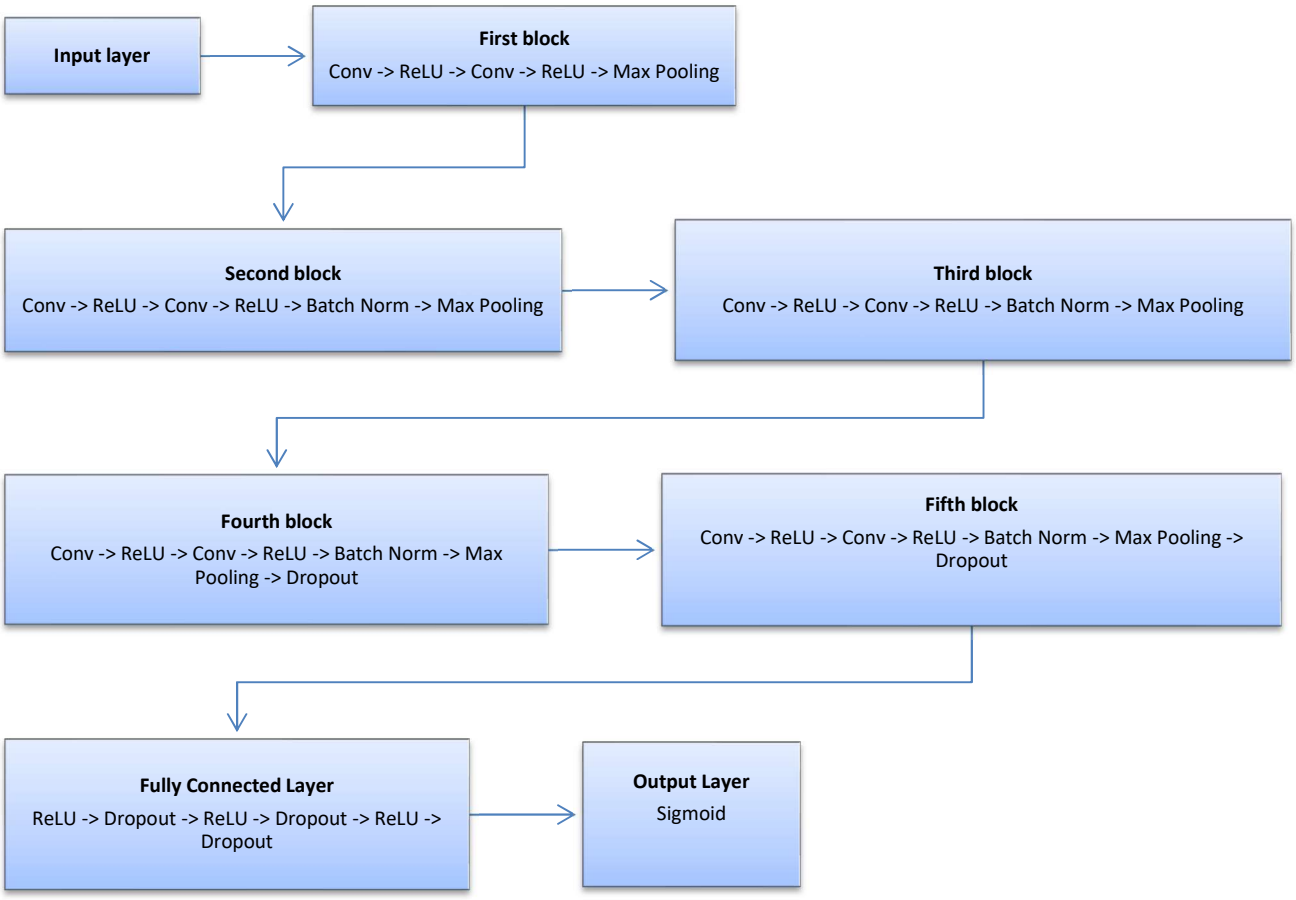
The first convolutional block consists of two normal convolutional layers, each followed by ReLU activation with no padding and then the max pooling layer.

The second and third convolutional block are mostly the same, consisting of two depth-wise separable 2D convolutions, each followed by ReLU activation. Then I apply batch normalization and then the max pooling layer.

The fourth and fifth convolutional block are mostly the same, consisting of two depth-wise separable 2D convolutions, each followed by ReLU activation. After that, batch normalization and max pooling are applied. Then I put a dropout with dropout rate of 0.2.

The fully connected layer starts with flattening the features and then I apply 3 dense layers with ReLU activation, each followed by a dropout with rate 0.6, 0.5, and 0.3 respectively.

Finally, the output layer is applied with sigmoid activation since we want to identify if an image shows the patient has a pneumonia or not. The block diagram is as follows:



3.3 Train the model

The next step is to train the model with Adam optimizer and cross-entropy loss and the 'accuracy' metrics.

I applied callback which is a set of functions to be applied at given stages of the training procedure. Here `ModelCheckPoint,` and `ReduceLROnPlateau` are those being used in this model.

- `ModelCheckPoint:` this function is used to save a copy of best performing model weights only when an epoch improves the metrics.
- `ReduceLROnPlateau:` models often benefit from reducing the learning rate by a factor of 2-10 once learning stagnates. This callback function monitors validation loss and if no improvements is seen for a 'patience' number of epochs, the learning rate is reduced.

I feed the training and test data generator to this specified model and run it (running file `final_project.py`) through 30 epochs with batch size of 32 images. The result and evaluation metrics are presented in part 4 – Experimental Evaluation below.

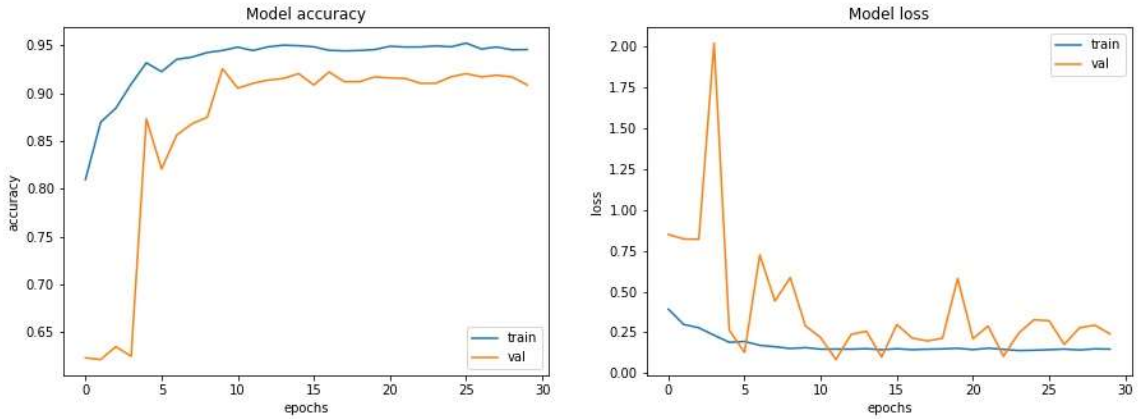
4. Experimental Evaluation:

The hypothesis here is the specific convolutional neural network model that we choose to map the input, which is the image data of normal and pneumonia patients, to the output, which is the condition of the patients, 0 if normal, 1 if infected.

As described earlier, the training dataset which consists of 5,232 normal and pneumonia chest X-ray images is used to learn the hypothesis and the test dataset which consists of 624 chest X-ray images (390 pneumonia, 234 normal) is used to evaluate it.

I use Accuracy of the test set as primary performance evaluation metrics along with Precision, Recall and F1 score for reference. Here with 390 out of 624 or 63% of test set images are of pneumonia patients, so I choose 63% as the baseline accuracy.

After taking time to do hyperparameter tuning with batch size, number of epochs, target image size, dropout rate, number of hidden units in each layer, I got the result demonstrated in the graph below with the model accuracy and model loss plotted against number of epochs.



We can see that after 30 epochs, it is converging with the model accuracy of training and validation increasing and coming together, the training and validation loss decreasing.

The final result is shown in the `output.txt` file with the accuracy of test data evaluation 91.5 %. This is substantially better than the baseline accuracy. Also, the Recall from test data is 96.92%, a particularly good result, it means that the model does exceptionally well in predicting the true pneumonia images over those actual infected. This is what we really want. Based on this result, it is clearly shown that my hypothesis is supported.

5. Related Work:

The author in this related work *https://becominghuman.ai/detecting-pneumonia-with-deep-learning-3cf49b640c14* used transfer learning, in which he utilized 34-layer ResNet to train on the same dataset, yielding the accuracy on the test set of 90.8%. Here, our model is built without the pre-trained ResNet, which means that it has more freedom to do hyperparameter tuning and also our result is slightly better.

6. Future Work:

In the future, one suggestion is to collect more labeled chest X-ray images to put it to the training and test set. Since the size of our dataset is relatively small, it is not easy to get a significantly better result than what I have done.

7. Conclusion:

It is amazing to see how deep learning contribute to solving real-world problems. In this project, I have shown a machine learning model classifies positive and negative

pneumonia chest X-ray images with a remarkable recall and accuracy. This is accomplished by putting several convolutional layers together with hyperparameter tuning. In the future, along with collecting more chest X-ray images from normal and pneumonia patients and labeling images, this work could be expanded to detect bacterial and viral infected patients.

8. Reference:

- https://www.who.int/maternal_child_adolescent/news_events/news/2011/pneumonia/en/
- https://data.mendeley.com/datasets/rscbjbr9sj/2
- https://www.cdc.gov/dotw/pneumonia/index.html
- https://www.who.int/gho/child_health/mortality/causes/en/
- https://becominghuman.ai/detecting-pneumonia-with-deep-learning-3cf49b640c14