Hooman Shiranimehr

# Final Report: COVID-19 Detection Based on Chest X-ray

**Problem Statement**

[Coronavirus disease 2019 (COVID-19)](https://en.wikipedia.org/wiki/Coronavirus_disease_2019) is a highly infectious disease that has affected almost every country in the world. Several tests have been developed to detect the disease at different stages. However, these tests can be slow and/or costly.

Here our goal is to investigate using patents’ chest X-ray to detect COVID-19.

In the following sections we first discuss data wrangling. Then we talk about Exploratory Data Analysis. After that, we investigate several models and compare their performance. Finally, we explain our conclusion and future work.

**Data Wrangling**

Data is obtained from <https://github.com/ieee8023/covid-chestxray-dataset>. This is a public open dataset of chest X-ray and CT images of patients which are positive or suspected of COVID-19 or other viral and bacterial pneumonias ([MERS](https://en.wikipedia.org/wiki/Middle_East_respiratory_syndrome), [SARS](https://en.wikipedia.org/wiki/Severe_acute_respiratory_syndrome), and [ARDS](https://en.wikipedia.org/wiki/Acute_respiratory_distress_syndrome)). Data has been collected from public sources as well as through indirect collection from hospitals and physicians.

Dataset contains patients’ information in a metadata file (Figure 1 and Figure 2) as well as corresponding X-ray images (Figure 3 and Figure 4).

Calendar

Description automatically generated

Figure 1

Table

Description automatically generated

Figure 2

A picture containing film, looking, person, sitting

Description automatically generated

Figure 3

A close up of a person

Description automatically generated

Figure 4

As it can be seen from Figure 1 and Figure 2, there are some columns like 'doi','url', 'license', 'clinical\_notes' and 'other\_notes' that are not relevant and are removed.

Also, some columns have missing values. These values replaced by “Unknow” or the average value of that columns as shown in Figure 5.

A picture containing text

Description automatically generated

Figure 5

There are also 21 records with no image. We remove those records.

Next, we check ‘finding’ column for different patents. This column has values such as ‘Pnemonie’, ‘SARS’ and ‘COVID-19’ as shown in Figure 6.

A screenshot of a newspaper

Description automatically generated

Figure 6

A new column ‘COVID-19’ is defined based on ‘finding’ and it is set to ‘True’ when finding includes ‘COVID-19’ and ‘False’ otherwise. Then images corresponding to patients with and without COVID-19 are moved to a folder named ‘Positive’ and ‘Negative’, respectively.

**Exploratory Data Analysis**

After data wrangling, we get 929 images (563 COVID-19 positive and 366 negative). Checking some of these images reveals that image sizes are different. Therefore, we change size of all images to 256x256 as shown in Figure 7.

Text

Description automatically generated

Figure 7

In order to evaluate our models, we use 70/15/15 split for training, validation and test. Also, given that training set is small and only consists of 650 images, we augment training set by generating 11 new images using random rotation, zoom, shift, etc..

**Model Selection**

[MobileNet](https://arxiv.org/pdf/1704.04861.pdf) is a lightweight architecture that uses depth-wise separable convolutions (Figure 8 and Figure 9).

Diagram, engineering drawing

Description automatically generated

Figure 8

A screenshot of a cell phone

Description automatically generated

Figure 9

For our base model, we use MobileNet with weights pre-trained on ImageNet. We discard the last layer of 1000 neurons and consider several models with 1 or 2 new layers added to the base model and different dropout values. Figure 10 shows 2 of the models considered.

A picture containing text

Description automatically generated

Figure 10

Notice that dropout values are relatively large as lower values lead to overfitting.

Figure 11and Figure 12 show loss and accuracy vs epoch for models 1 and 2. As it can be seen from these figures, in both models, accuracy and loss for training set are improved as value of epoch is increased. On validation set, we see some fluctuations but both accuracy and loss seem relatively stable. If we increase number of epochs to 20 or 30, we see gaps between the two curves for each model which means the model will overfit. We choose Model 1 as it has slightly higher accuracy and lower loss.

Chart, line chart

Description automatically generated

Figure 11

Chart, line chart

Description automatically generated

Figure 12

Next, let’s consider ROC and Precision-Recall Curves for test Set. The results are shown in Figure 13 and Figure 14.

Chart, line chart

Description automatically generated

Figure 13

Chart

Description automatically generated

Figure 14

Figure 15 shows accuracy and confusion matrix for the case that threshold is set to 50% probability.

A picture containing text

Description automatically generated

Figure 15

**Conclusion and Future Work**

In the previous sections we discussed data wrangling, EDA and modeling. As we saw, for our problem, Model 1 seems to achieve the best performance. Therefore, based on the requirements, one can use this classifier with a certain threshold to achieve the best acceptable/desired levels of precision and recall.

Note that as our ultimate goal is to detect a disease, it makes sense to focus on reducing false negatives. By doing so, we would we get more false positives which can be fine as the disease can be ruled out by other tests. However, if the false positive is too high, it can cause additional unnecessary cost for retesting and also concern in many patients and their families. Therefore, we need to try to find an optimal threshold to balance these two.

Overall, it can be a challenging task to find the optimal operating point (threshold) and other factors such as user experience, business and finance need to be considered in order to make the best decision.

Although we considered several different classifiers, further optimization is still possible by tuning hyperparameters such as number of additional layers and dropout ratios. Also, it is possible to use additional patients’ information and consider a model with both structured and unstructured data.