CPSC 340 Machine Learning Take-Home Final Exam
Individual Portion - Question 2
(Spring 2020)

1 Team

Team Members	Muiez Ahmed (33302167, muiez), Daksh Sharma (dakshubc, 11638160), Yue Qian (Floris) Huang (floris, 32747131), Zihan (Lewis) Liu (zihanliu, 30647168)
Kaggle Team Name	COVID-340

2 Introduction

The 2019 novel cornavirus (COVID-19) has become a global pandemic that has surpassed 3,000,000 confirmed cases and claimed the lives of hundreds of thousand of patients. Due to a global shortage of test kits, it is imperative that an automatic detection system be explored as a quick and effective alternative for accurate diagnosis when provided with some form of medical metrics of the presumptive patients, such as by checking X-ray. This problem aims to construct a classifier capable of differentiating between non-COVID and COVID chest X-rays. A major obstacle is the lack of training data. This might cause high overfitting when fitted against a complex model, which is presumably required to detect subtle differences in medical images.

3 Method

The dataset contains 70 training images and 10 test images. Out of the training images, 55 were covid-positive and 15 where not. Given the small training set and class imbalance, as well as the nature of the Chest X-ray image data, our approach was to use `Tensorflow Keras` as our framework and perform transfer learning on pre-trained CNN models. Transfer learning was specifically chosen because it allows us to utilize parameters obtained from large datasets, e.g. `ImageNet` so that we can effectively train on our dataset with significantly fewer samples. It is also computationally more efficient. For this purpose, we selected three pre-trained CNN models: `ResNet50`, `InceptionV3` and `VGG16` [1], and performed `sklearn ensemble` learning on top of the individual results in an attempt to produce a higher prediction accuracy.In order to use the pre-trained CNN models more effectively, we rescaled/denormalized the input images , and passed them through the `preprocess_input` function that came with each model. To work around the insufficient training data problem, we used `Keras ImageDataGenerator` to randomly perturb the training images (70 permutations, preserving class ratio), and add to our training set to double its size. Our experiments were thus based on the preprocessed and augmented dataset of 140 training images (110 COVID and 30 non-COVID).

4 Experiments

Note: the parenthesised text denotes the outcome of each experiment: successful, failed, or inconclusive.

1. Dataset preprocessing and augmentation:
    - image de-normalization (inconclusive): we experimented with the reverse of min-max normalization and standardization. While the resulting image of min-max image looked comparable to an unaltered X-ray image, we only saw little improvements in later epochs of the model fitting process. We were not able to find a reasonable explanation.
    - deskewing per channel (failed): no visible deskewing was observed presumably due to the dim nature of X-ray images
    - resizing (inconclusive): no noticeable improvement presumably due to sub-optimal resizing scheme and that `Keras` pre-trained models provide support for images of all sizes.

[1] As suggested in Narin, et al. Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks

- random augmentation with `Keras` (inconclusive): this resulted in 130 images of each class. Since empirical evidence exists that support its usage, we suspect that the lack of improvement is due to other reasons mentioned in this section.

2. Fine tuning model architecture and hyperparameters(below was done on all three models chosen):
   - architecture (inconclusive): as an example, for `ResNet50` we took the base layers from `Keras` pre-trained model and added output layers on top: 1 global average pooling, 1 fully connected layer activated by ReLu, one dropout layer, and one final output layer activated by softmax. We experimented with various hidden unit sizes they did not seem to affect the result significantly.
   - optimizer (successful): we experimented with `SGD` with momentum, `RMSProp` and `Adam` with different learning rates. We found that a smaller learning rate (i.e. 1e-5) with Adam optimizer led to least overfitting.
   - fitting and hyperparameters tuning (successful): we tried using `fit` and `fit_generator` (which randomly perturb training images during fitting). The results were similar. To deal with the class imbalance problem, we tried using the `class_weight` parameter in `fit`, which produced better result. We also tried tuning batch_size from range 1-10 and epochs range from 15-40 before finally opting for batch_size=2 and epochs=30.
3. Cross-validation and result analysis:
   - `sklearn Stratified K-fold` (successful): we use 5-fold cross-validation on the training set for each of the three CNN base models, using batch_size= 2 and epochs=30 for each experiment.
   - performance matrix: we visualized different aspects of model performance by using a confusion matrix from calculations of accuracy, precision, recall and f1 scores. The result indicated that even with class weight adjustment we still see quite a few false positives, in other words our ability to detect non-COVID images was weaker than the COVID ones.

5 Results

Model	Kaggle Score
Transfer Learning Ensemble	*83.87%*

6 Conclusion

While transfer learning is proven to be successful in training on novel/small/insufficient datasets, we did not manage to produce hugely satisfactory results given the time constraint.

The main challenge of our study was the limited number of images available for training the deep CNNs, as well as the dominating problem of class imbalance. To overcome this problem, we tried a variety of approaches for data preprocessing and hyperparameter tuning on top of performing transfer learning. We suspect that the major issue with our attempt was that of data processing still since the nature of pre-trained models eliminated the chances of our model itself being wildly off-mark. Also, transfer learning literature also lend us confidence that the initial data processing that is to blame. We imagine if the problem was changed to the classification between strictly healthy and COVID X-ray images, it would be less challenging as the model would only have to learn to differentiate more distinct features. In addition, if more training examples were included, we believe our model will yield a better result.

Through the experiments, we learned basic concepts of transfer learning and its significance in addition to honing on our skills to perform effective hyperparameter tuning with complex models. However, we are yet to acquire an in-depth understanding of how to fine tune the weights in the pre-trained models to further optimize our results.