Classifying COVID-19 Chest X-Rays Using Custom CNNs and Automatic Machine Learning

AZUSA PACIFIC UNIVERSITY

Joshua Stapleton

DEPARTMENT OF ENGINEERING & COMPUTER SCIENCE - 29th April 2020

Project Goals

- Two-pronged approach
  - Automatic data import from online
  - CNNs (90% accuracy)
  - AutoML (?% accuracy)

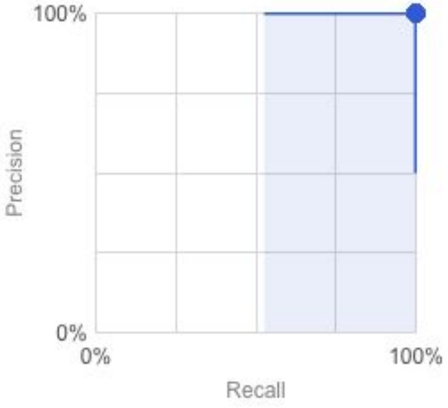
Best Solution Summary

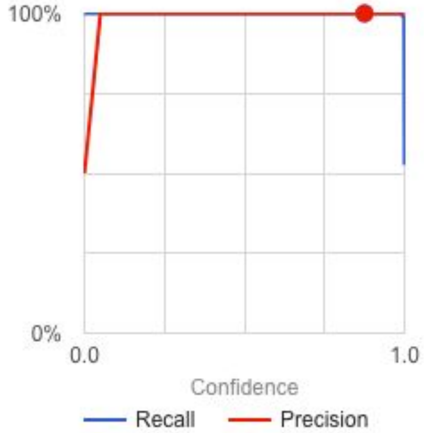
- Google Cloud’s AutoML.
  - Online data pipeline (761 positive and negative images)
  - Jupyter Notebook Development Environment for pre-processing.
  - Tensorflow, Keras, Numpy, Pillows, Tensorboard, BeautifulSoup, and many more!

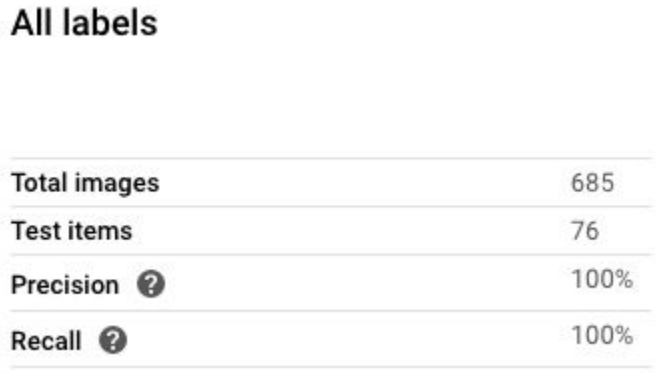
Google’s AutoML K python™ TensorBoard TensorFlow BeautifulSoup

Results of Best Solution

- 100% accuracy.
  - 4 hours of training.
  - Online deployment.
  - F1 score of 100% (with 100% precision and 100% recall.)

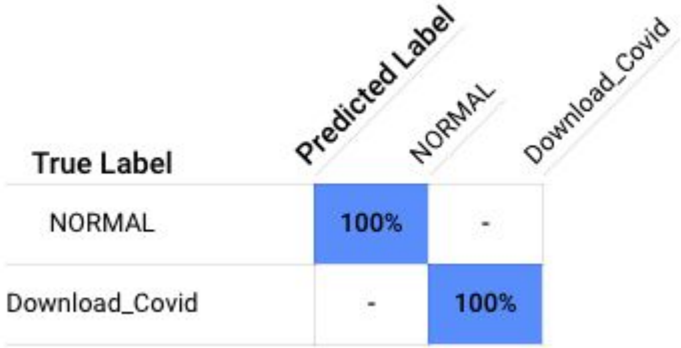






All labels

Total images	685
Test items	76
Precision	100%
Recall	100%



True Label \ Predicted Label	NORMAL	Download_Covid
NORMAL	100%	-
Download_Covid	-	100%

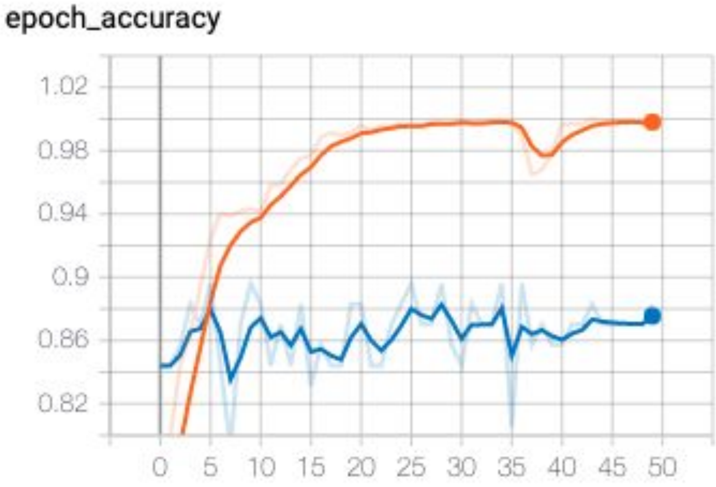
Results of CNN

covid

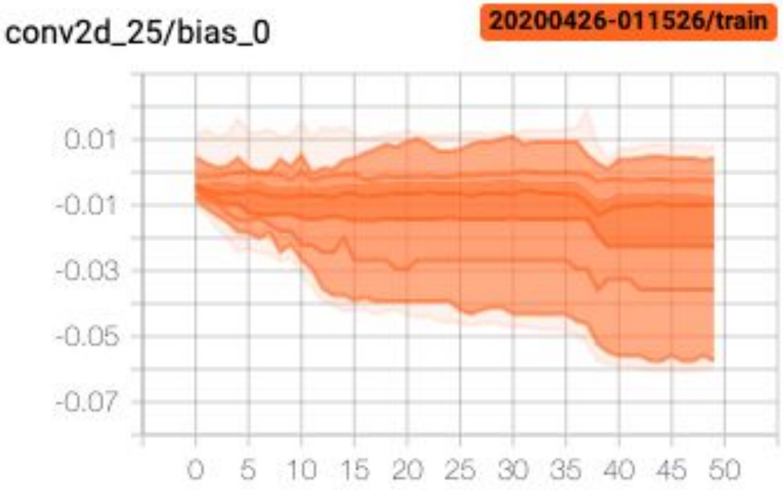
non-covid

covid

non-covid



epoch_accuracy



conv2d_25/bias_0

20200426-011526/train

CNN structure

How The Results Compare...

- Kumar et al: *Accurate Prediction of COVID-19 using Chest X-Ray Images through Deep Feature Learning model with SMOTE and Machine Learning Classifiers.* *https://www.medrxiv.org/content/10.1101/2020.04.13.20063461v1*

  ~ 97.7% (XGBoost) - 5840 Images.

- Narin et al: *Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks.* https://arxiv.org/pdf/2003.10849.pdf

  ~ 98.0% (ResNet50) - 50 images.

- Minaee et al: Deep-COVID: *Predicting COVID-19 From Chest X-Ray Images Using Deep Transfer Learning.* *https://www.researchgate.net/publication/340806168_Deep-COVID_Predicting_COVID-19_From_Chest_X-Ray_Images_Using_Deep_Transfer_Learning*

  *~ 97.0% (*DenseNet-121*)* - 5,000 Images.

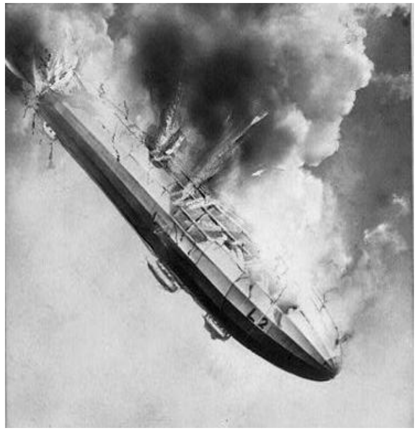
- Our results are very similar, (if not better) considering data-set size.

Prominent Tweaks / Ingenuities...

- Simple is better
  - 2 levels of approach (first custom, then auto)
  - Tensorboard (50% accuracy increase + 100x faster)
  - Data standardization (image scaling)
  - Online data pipeline.
  - All feeding into Auto.

Other Solutions Attempted / Challenges

- AlexNet
  - Terrible Accuracy. (~40-50%)
  - Very long training time. (~12 hours)
- SciKit Learn
  - No available CNNs (tried Random Forests, Extra Trees Classifiers, SVMs.)
  - Decent accuracy, but difficult to customize. (60-70%)
- Tensorboard
  - Creating logs directory
  - Fails sometimes (memory error, runtime error)



“Hypothetical” Future Plan

- Data data data!
  - Higher diversity of images.
  - Stricter testing (model confidence.)
- Potential deployment
  - “Real world testing” (optimized for hospital use.)
  - Multi-Machine model deployment (works across many platforms.)
  - Downloading from cloud

(Many) Main Lessons Learned

- Always an easier way to do something.
- You don’t always have to know what is going on.
- Data. Is. Everything.
- 90% of the work is debugging.
- Does not have to be perfect, just start
- Patience because the code might fail you