SINA AI: Optimal Speed and Accuracy of Pneumonia Detection

Jelaleddin Sultanov* - SINA AI
jelaleddinsultanov@yandex.ru

Abstract

There is a huge workload on current global healthcare diagnostics system, especially in Second and Third World countries. Overall mortality of different diseases are mostly, as WHO stated, caused by inefficient diagnostics in healthcare system. However, with the rise of Deep Learning and Computer Vision technologies, different AI-powered IT solutions were introduced as a solution to this kind of problems. We assume that with the help of AI-powered diagnostics systems, countries could decrease the mortality of people and the load on diagnostics in healthcare. We have developed VGG-16 model and trained it on publicly available dataset of pneumonia disease.

Source code is at
https://github.com/jenapss/Tensorflow-2.3-Pneumania-Detection-System

1. Introduction

There are millions of X-ray, MRI and CT machines all over the world and millions of images have been created by these machines every year. These images mostly have been stored in local PACS storage systems without further being utilized. Also, the cost of current diagnostics makes it inaccessible to majority of population and thus increasing the mortality and load of healthcare system. With Deep Learning and Computer Vision we can utilize these images for training deep learning models, that, in its turn, can reduce the cost of current healthcare diagnostics. With all its advantages, AI applications, nevertheless where they have been applied, they are meant to automated given workflow and reduce the cost. And similarly, AI applications in diagnostics are designed to automate the diagnostic workflow and reduce the cost of diagnostics. The goal of this research is to present our solution to these kind of problems. We have developed AI-powered web application MVP using VGG16 algorithm that is trained on publicly available dataset of pneumonia dataset.

2. Project Implementation.

2.1 Dataset

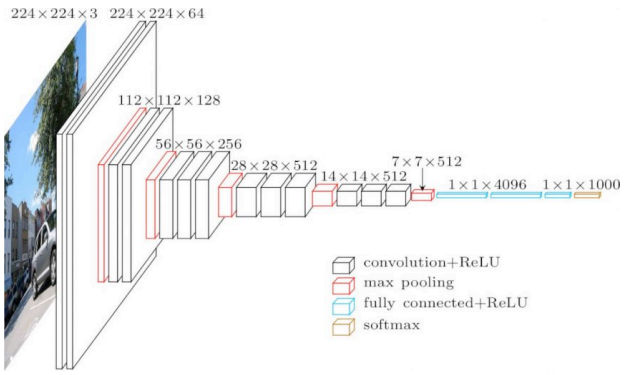


Fig1. Classic VGG-16 architecture

Dataset for this project was found online as open-source collection of annotated chest X-ray images. **[1]**
Chest X-ray images (anterior-posterior) were selected from retrospective cohorts of pediatric patients of one to five years old from Guangzhou Women and Children's Medical Center, Guangzhou. All chest X-ray imaging was performed as part of patients' routine clinical care.

For the analysis of chest x-ray images, all chest radiographs were initially screened for quality control by removing all low quality or unreadable scans. The diagnoses for the images were then graded by two expert physicians before being cleared for training the AI system. In order to account for any grading errors, the evaluation set was also checked by a third expert.

The dataset is organized into 3 folders (train, test, val) and contains subfolders for each image category (Pneumonia/Normal). There are 5,863 X-Ray images (JPEG) and 2 categories (Pneumonia/Normal).

2.2 Model architecture

VGG16[2] is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition". The model

achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes. It was one of the famous models submitted to ILSVRC-2014.
The input to cov1 layer is of fixed size 224 x 224 RGB image. The image is passed through a stack of convolutional (conv.) layers, where the filters were used with a very small receptive field: 3×3 (which is the smallest size to capture the notion of left/right, up/down, center). In one of the configurations, it also utilizes 1×1 convolution filters, which can be seen as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of conv. layer input is such that the spatial resolution is preserved after convolution, i.e. the padding is 1-pixel for 3×3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2×2 pixel window, with stride 2.

Three Fully-Connected (FC) layers follow a stack of convolutional layers (which has different depth in different architectures): the first two have 4096 channels each, and the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class). The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks.
All hidden layers are equipped with rectification (ReLU) non-linearity. It is also noted that none of the networks (except for one) contain Local Response Normalization (LRN), such normalization does not improve the performance of the ILSVRC dataset, but leads to increased memory consumption and computation time.[3]

2.3 Technology Stack

As it was mentioned earlier, the goal of this paper is to demonstrate one of the possible uses of Deep Learning models and that one of the uses is the web application powered by VGG-16 model that is capable to detect pneumonia on X-ray images. To make a web-application, we used

Python Flask web framework. Flask is one of the popular Python web-frameworks.
This projects can be extended with more built in features but the this project MVP of working web-app is also suitable. In training phase of VGG-16 model, we used Tensorflow - open-source end-to-end machine learning and artificial intelligence. As training environment, we chose Kaggle - the free platform for training Machine Learning and Deep Learning models. These are the main technology stack that I used during the model development.

3. Observations and Results

There are few disadvantages of the dataset that I chose to use. One of the main disadvantages is that this open-source dataset has imbalanced distribution of “health” and “pneumonia” labeled images. In the figure below, the labels that are written as “0” and “1” mean “normal/health”and “pneumonia”

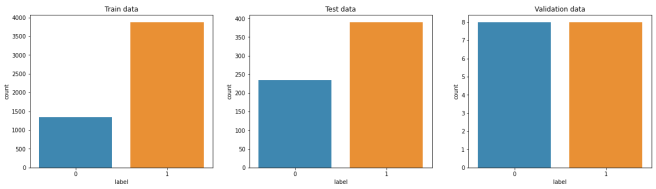
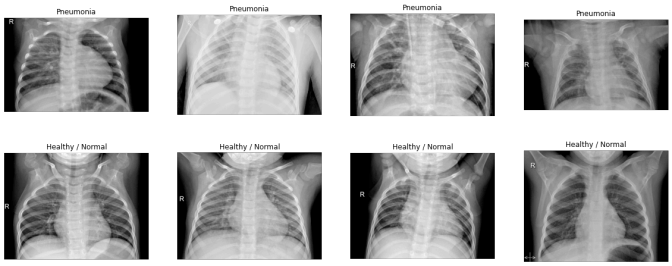


Fig. 2 - Imbalanced dataset

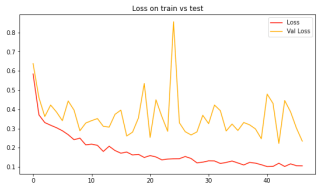
As you see, our dataset is imbalanced and this usually can potentially influence on generalization ability of the model. Nevertheless, it is still possible to go on and train our model. But before starting the model training, let’s have a look at our dataset:



As you see, this is first raw visualization of our dataset. This dataset was already has already passed examination from radiology professionals and low quality and unreadable scans were removed. Now let have a look at our model summary:

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 196, 196, 8)       400
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 196, 196, 8)       3144
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 65, 65, 8)         0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 65, 65, 16)        3216
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 65, 65, 16)        6416
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 21, 21, 16)        0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 21, 21, 32)        4640
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 21, 21, 32)        9248
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 10, 10, 32)        0
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 10, 10, 64)        18496
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 10, 10, 64)        36928
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
conv2d_8 (Conv2D)            (None, 5, 5, 128)         73856
_________________________________________________________________
conv2d_9 (Conv2D)            (None, 5, 5, 128)         147584
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 2, 2, 128)         0
_________________________________________________________________
flatten (Flatten)            (None, 512)               0
_________________________________________________________________
dense (Dense)                (None, 128)               65664
_________________________________________________________________
dropout (Dropout)            (None, 128)               0
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 258
=================================================================
Total params: 369,850
Trainable params: 369,850
Non-trainable params: 0
_________________________________________________________________
```

This is a 5 layer sequential model built with Tensorflow framework. In the figure above you can observe all parameters of the model such as input size, and number of parameters at each layer. The result of our training is depicted below:



As we can notice from the training performance plot above, the loss both at training and test sets decreases as training and testing accuracy increases up to 95%. We can also notice that the difference between losses of training and test sets, this difference can tell us that there is some degree of overfitting. The overfitting is common problem among machine learning projects and there are various kinds of regularization methods to decrease the overfitting effect on the models. Overfitting usually affects model's overall generalization ability by decreasing of generalization ability of the method. But however, now lets have a look to classification report on the dataset and confusion matrix on test dataset:

```
Classification report on test data
              precision    recall  f1-score   support

           0       0.93      0.87      0.90       234
           1       0.92      0.96      0.94       390

    accuracy                           0.92       624
   macro avg       0.93      0.91      0.92       624
weighted avg       0.92      0.92      0.92       624
```

Fig 4 Classification

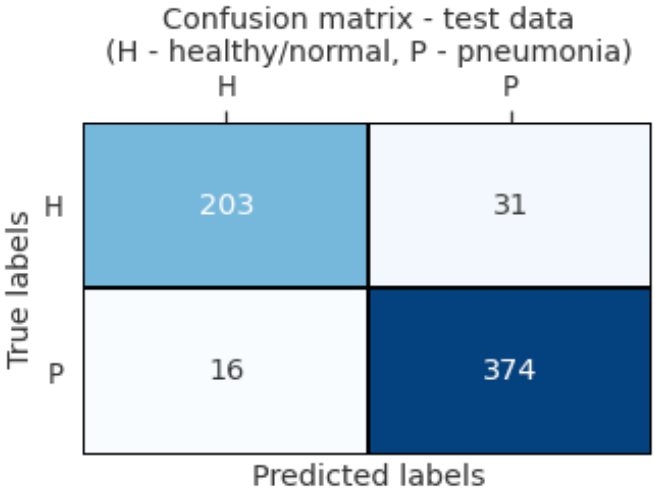


Fig 5 Confusion Matrix

Conclusion

This project introduces the fundamental idea of application of AI-powered systems in healthcare. Although I was able to achieve 93% of accuracy, this does not assume that it is ready to be applied in production due to several constrains such as dataset size. It is accepted that "entry level" for dataset size to be used in production is 10,000 images. Though, I am pretty sure that this model development can be taken further along with setting up automated data workflows that perform continuous training(CT) and model validation within a CI/CD pipeline based on updated data versioning.

4. References

[1] *Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification*. Daniel Kermany, Kang Zhang, Michael Goldbaum

[2] *Very Deep Convolutional Networks for Large-Scale Image Recognition*. Karen Simonyan, Andrew Zisserman

[3] Deep Learning using Rectified Linear Units (ReLU). Abien Fred Agarap