Machine Learning Engineer Nanodegree

Capstone Proposal

Aaron Balson Caroltin J
August 2019

Proposal

Domain Background

Imagine being able to detect blindness before it happened.

Millions of people suffer from diabetic retinopathy, the leading cause of blindness among working aged adults. Aravind Eye Hospital in India hopes to detect and prevent this disease among people living in rural areas where medical screening is difficult to conduct. Successful entries in this Kaggle competition will improve the hospital's ability to identify potential patients. Further, the solutions will be spread to other Ophthalmologists through the 4th Asia Pacific Tele-Ophthalmology Society (APTOS) Symposium.

My personal motivation for investigating this problem domain is double fold – to get a taste of these competitions to increase my knowledge which may also contribute a local hospital to serve the community better. The symposium is planned on Sept 22-23 at my city Chennai, which is a good opportunity to meet and discuss with winners/experts.

Problem Statement

Currently, Aravind technicians travel to these rural areas to capture images and then rely on highly trained doctors to review the images and provide diagnosis. The accuracy of such screenings by interns and juniors can vary significantly with one study found a 49 percent error rate[1].

Their goal is to scale their efforts through technology; to gain the ability to automatically screen images for disease and provide information on how severe the condition may be, as accurately as possible. Here the problem is quantified as classifying the correct stage of Retinopathy and one relevant solution is to use image classification model to predict and assist humans with high accuracy.

Datasets and Inputs

We will use the datasets provided by Aravind Hospital itself at Kaggle's competition data page, so they are highly relevant to our problem. It was collected over an extended period of time from different clinics and classified by doctors.

- train.csv - the training labels (has id_code, diagnosis)
- test.csv - the test set (has id_code)
- sample_submission.csv - (has id_code, diagnosis)
- train.zip - the training set images
- test.zip - the public test set images

train=3662, test=1928 entries in total

Images are known to have noise, variance in resolution, size & scale, over/under exposed which are challenges to be taken care during pre-processing. More over the entries are skewed towards class-0 (imbalanced).

```
0    1805
2     999
1     370
4     295
3     193
```

Solution Statement

Using Convolutional Neural Network Image classification model with DenseNet Architecture[2] we can create high precision models required for medical domain to quickly train and accurately identify the 5 classes of diagnosis

0 - No DR

1 - Mild

2 - Moderate

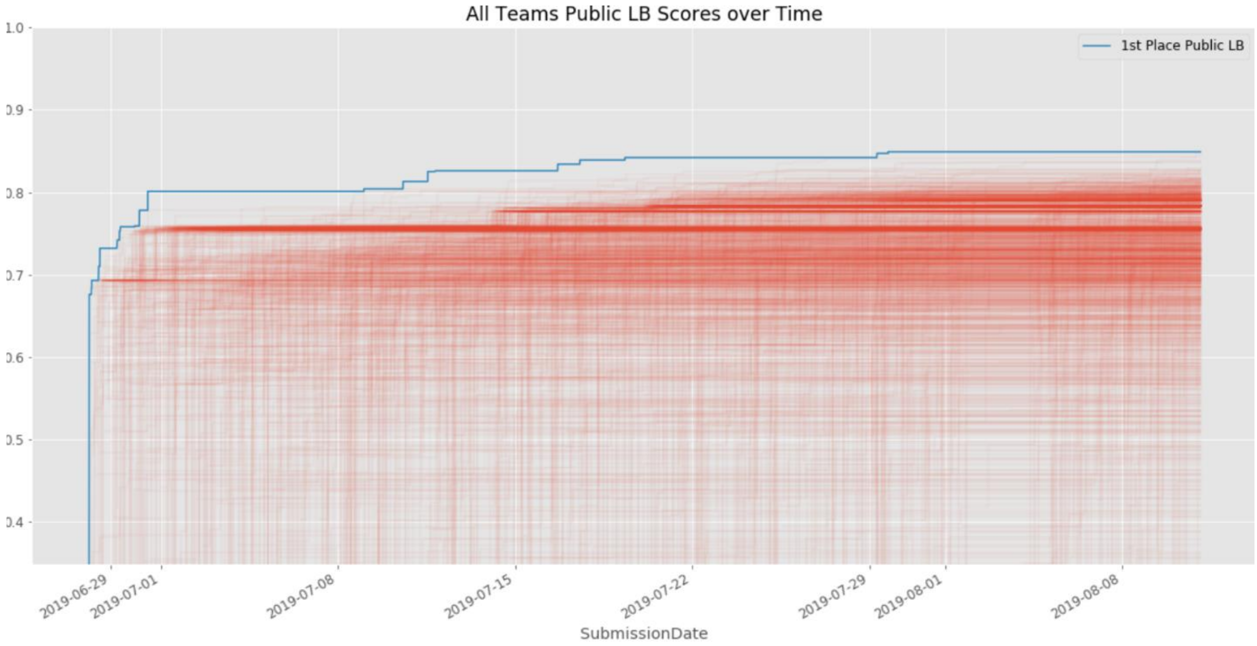
3 - Severe

4 - Proliferative DR

We will be using the free kernel from Kaggle workspace to experiment. We will use transfer learning to expediate training with DenseNet (over Standard ConvNet or ResNet) because the network can be thinner and compact as a result which also has a higher computational and memory efficiency (to run within time limits).

Benchmark Model

One study found a 49% error rate for diagnosing Retinopathy among internists, diabetologists, and medical residents[1]. Another study shows the Sensitivity (95% CI) of direct ophthalmoscopy (from patients) for any retinopathy, NSTDR and STDR was found to be 55.67% (50.58–60.78), 37.63% (32.67–42.59) and 68.25% (63.48–73.02) respectively[4]. When diagnosing from images, this number can only get worse. I will try to set 68.25% as benchmark score to beat using the *quadratic weighted kappa* (QWK) metrics required for the competition. Currently the leader score is ~85%.



Evaluation Metrics

Submissions are scored based on the *quadratic weighted kappa*, which measures the agreement between two ratings. This metric typically varies from 0 (random agreement between raters) to 1 (complete agreement between raters). In the event that there is less agreement between the raters than expected by chance, this metric may go below 0. The quadratic weighted kappa is calculated between the scores assigned by the human rater and the predicted scores.

Images have five possible ratings, 0,1,2,3,4. Each image is characterized by a tuple *(e,e)*, which corresponds to its scores by *Rater A* (human) and *Rater B* (predicted). The quadratic weighted kappa is calculated as follows. First, an N x N histogram matrix *O* is constructed, such that *O* corresponds to the number of images that received a rating *i* by *A* and a rating *j* by *B*. An *N-by-N* matrix of weights, *w*, is calculated based on the difference between raters' scores:

An *N-by-N* histogram matrix of expected ratings, *E*, is calculated, assuming that there is no correlation between rating scores. This is calculated as the outer product between each rater's histogram vector of ratings, normalized such that *E* and *O* have the same sum.

Project Design

1. Setup ENV and download/import required packages.
2. Load given datasets to understand its format.
3. Exploratory data analysis on csv and images.
4. Perform pre-processing, resizing/rescaling to match that of ImageNet (224x224) as we intend to use DenseNet (which was trained on ImageNet) for transfer learning.
5. Use Multilabel instead of multiclass encoding e.g. represent class 4 as [1,1,1,1,0] instead of traditional [0,0,0,1,0] for better kappa score.
6. Perform data augmentation using Data Generator since training set is small.
7. Adopt DenseNet-121 Architecture and append our dense and output layers at end their conv layers.
8. Train the model and keep track of accuracy and kappa score.
9. Plot training vs validation loss to inspect overfitting.
10. Fine tune hyperparameters and optimizers based on kappa score.
11. Test the model with given data set.
12. Create submission file with test results and submit to Kaggle for public score.
13. Wait for public score and aim to improve.

Citations & References

1 https://www.sciencedaily.com/releases/2019/03/190318141135.htm

2 https://towardsdatascience.com/review-densenet-image-classification-b6631a8ef803

3 https://www.kaggle.com/c/diabetic-retinopathy-detection/leaderboard

4 https://www.sciencedirect.com/science/article/abs/pii/S1871402114000277

http://2019.asiateleophth.org/

https://aravind.org/vision-mission/