Detecting Diabetic Retinopathy

By Nick Leffell, Trevor O’Donnell, and Dennis Tran

Diabetic Retinopathy

- A complication of diabetes that affects the eyes.
- Caused by damage to the blood vessels in tissue at the back of the eye (retina).
- Diabetic patients will develop diabetic retinopathy after they have had diabetes for between 3-5 years.
- DR is generally diagnosed within 3 years of type 1 diabetes, but may already be present when type 2 diabetes is diagnosed.
- Poorly controlled blood sugar is a risk factor.
- Mild cases may be treated with careful diabetes management. Advanced cases may require laser treatment or surgery.

Background

“Millions of people suffer from diabetic retinopathy, and is 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.

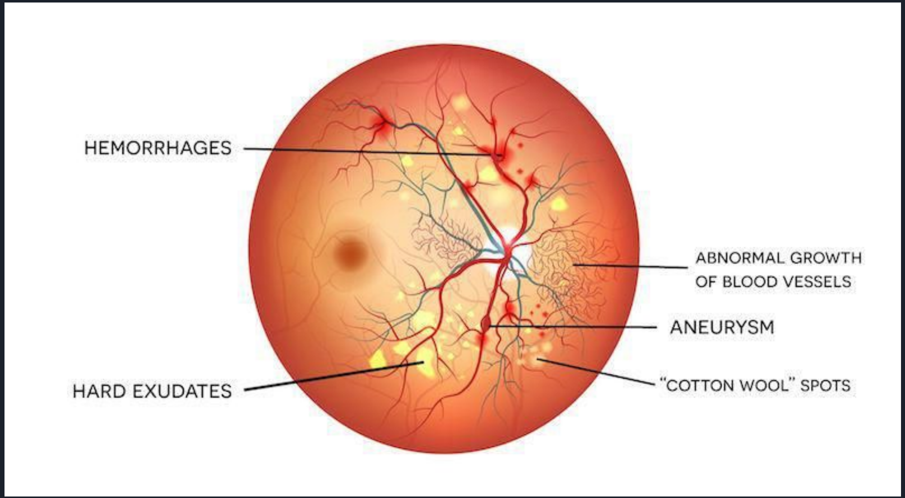
Currently, Aravind technicians travel to these rural areas to capture images and then rely on highly trained doctors to review the images and provide diagnosis. Their goal is to scale their efforts through technology; to gain the ability to automatically screen images for disease and provide information on how severe the condition may be.”

Build a machine learning model to speed up disease detection, working with thousands of images collected in rural areas to help identify diabetic retinopathy automatically.

Objective

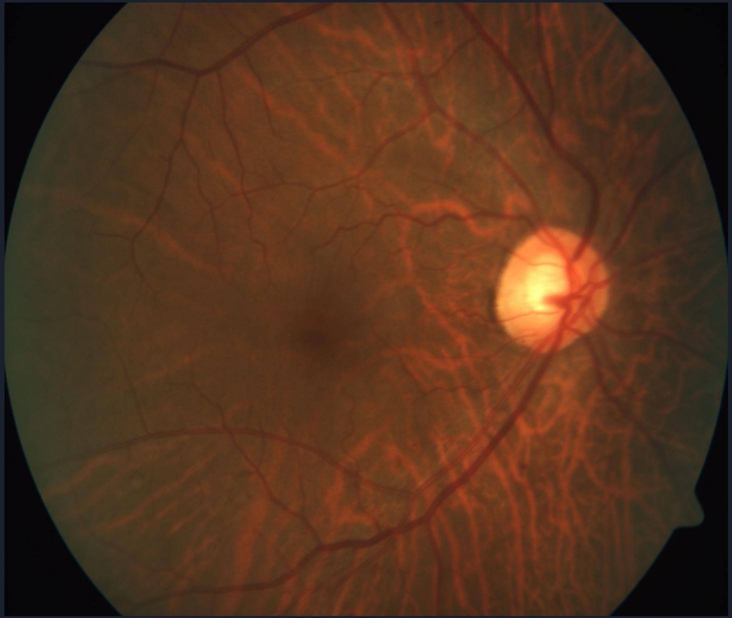
A clinician has rated each image for the severity of diabetic retinopathy on a scale of 0 to 4:

- **0 - No DR**
- **1 - Mild**
- **2 - Moderate**
- **3 - Severe**
- **4 - Proliferative DR**



Dataset

- APTOS 2019 Blindness Detection from Kaggle:
  - Train Images (.png)
  - Train Labels (.csv)
  - Test Images (.png)
  - Test Labels (.csv)
- 3662 Images:
  - 640 x 480
  - 3 Color Channels
- The images were gathered from multiple clinics using a variety of cameras over an extended period of time, which will introduce further variation.



Data Loading

- Used plyplot imread and numpy resize to read scale down from 640 x 480 pixels.
- Dropped the .png suffix using list comprehension to match the image names.
- Rescaled our images by dividing each one by 255.
- Train-test split
- Used utils.to_categorical to dummify our target variable.

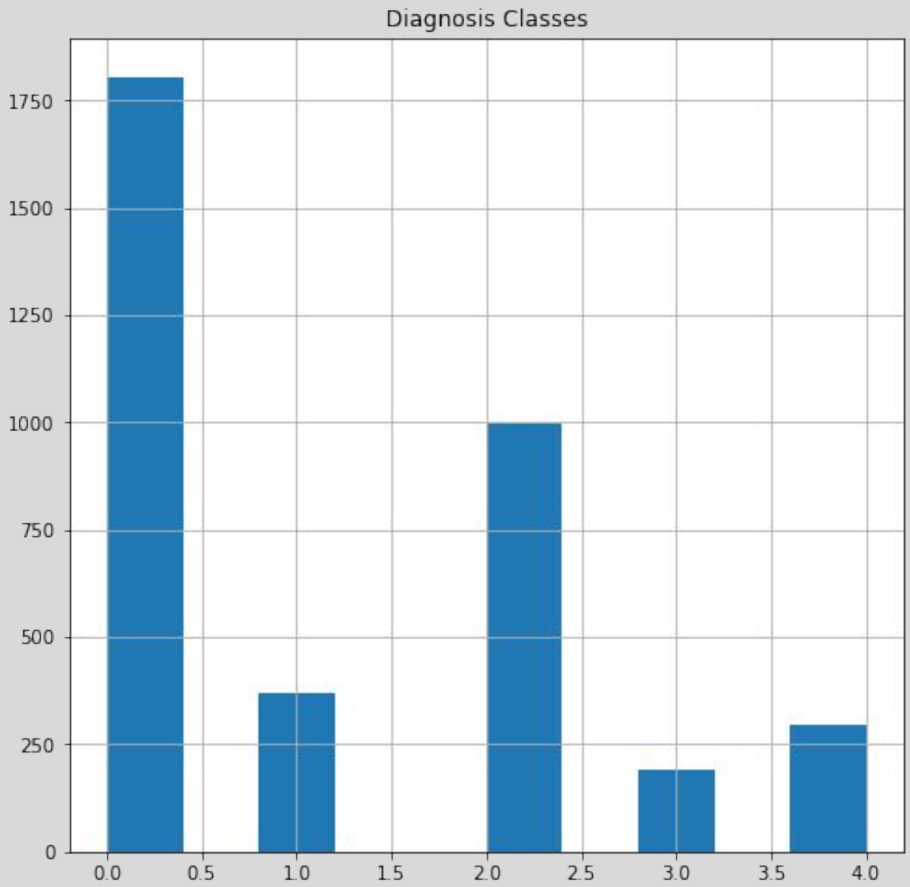
```
In [ ]:
%%time
folder = os.listdir('./alltrain/train/train/')

images = []
labels = []


for file in folder:

    filepath =  './alltrain/train/train/' + file
    image = cv2.imread(filepath)
    # read in the files and downscale to 25x25
    image = cv2.resize(image, (128, 128))
    images.append(image)
    # store images and labels
    labels.append(file)
```

Diagnosis Classes For Test Data

0: 49.49%
1: 10.10%
2: 27.28%
3: 5.27%
4: 8.06%

Baseline: 49.49%

Classes are imbalanced



Modeling – Models Table

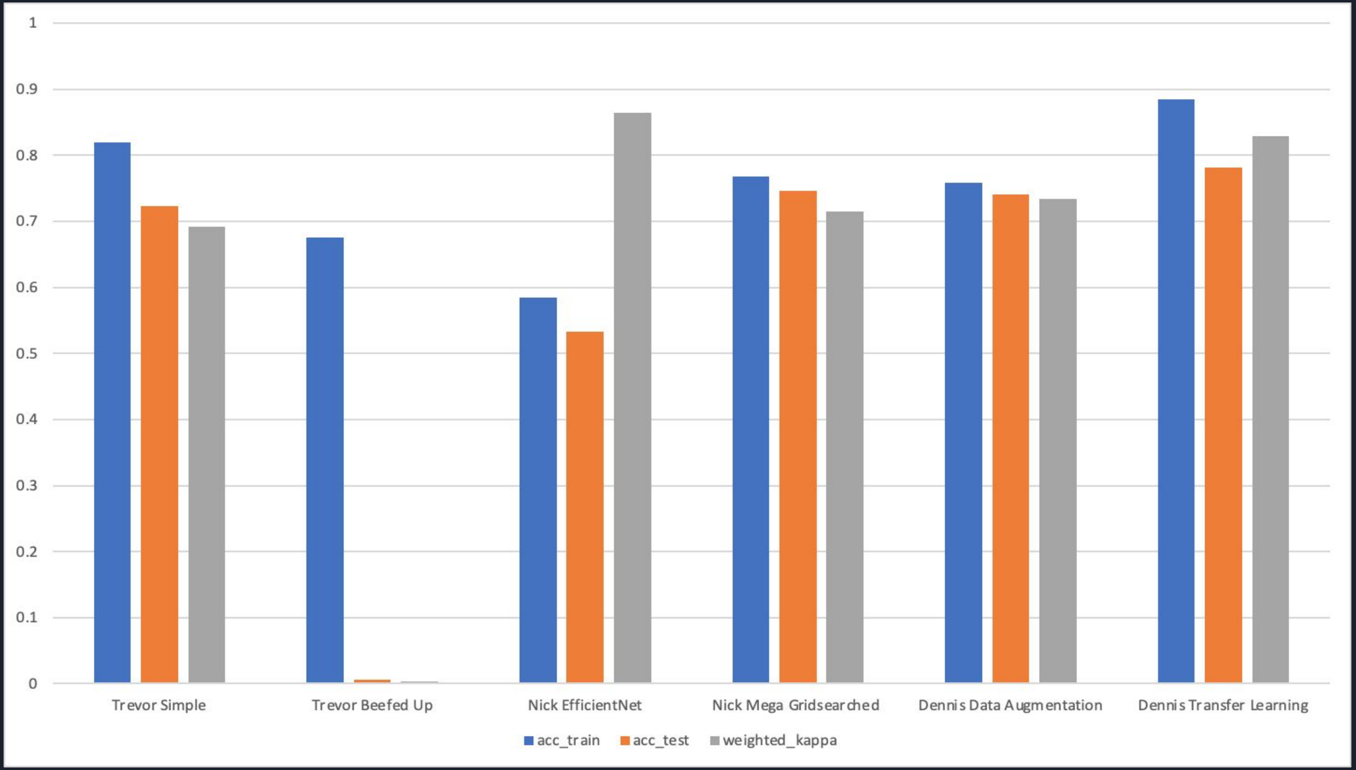
Name	Structure	Regularization	Score & difference	WQKC
Trevor Simple	2 conv layers 3 dense layers	Early Stop	0.8193 training, 0.7223 testing	0.6914
Trevor Beefed Up	4 conv layers 3 (large) dense layers	Early Stop, Dropout, Batch Normalization	0.6749 training, 0.0057 testing	0.0032
Nick Gridsearched CNN	2 conv layers Big filters/kernels Big dense layers	None	0.7677 training, 0.7467 testing	0.7186
Nick EfficientNet	32 layers No Hidden Small batch size	Early Stop, Dropout, Group Normalization	0.5841 training, 0.5328 testing	0.8643
Dennis Data Augmentation	1 Augmentation 4 Convolution No Hidden	Early Stop	0.7578 training, 0.7402 testing	0.7334
Dennis Transfer Learning	1 Model No Hidden	Early Stop	0.8849 training, 0.7817 testing	0.8291

Metric: Quadratic Weighted Kappa

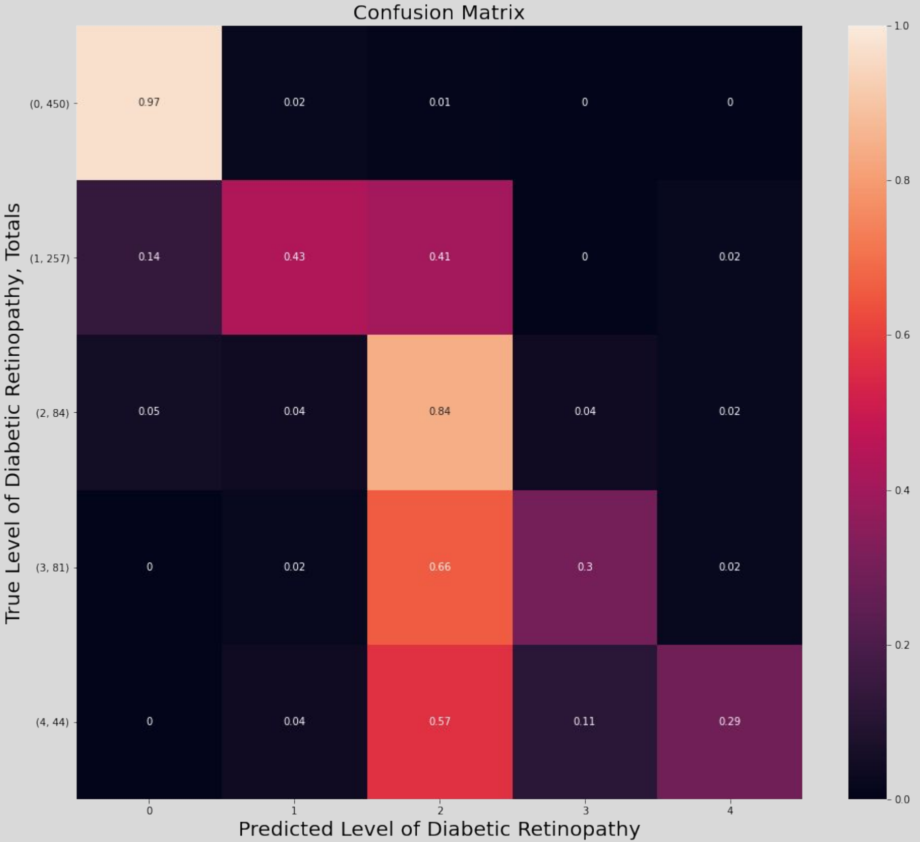
$$P_e = \sum_i \sum_j W_{ij} P_{i+} P_{+j}$$

- Quadratic Weighted Kappa 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, the metric may go below 0.
- The quadratic weighted kappa is calculated between the scores which are expected/known and the predicted scores.

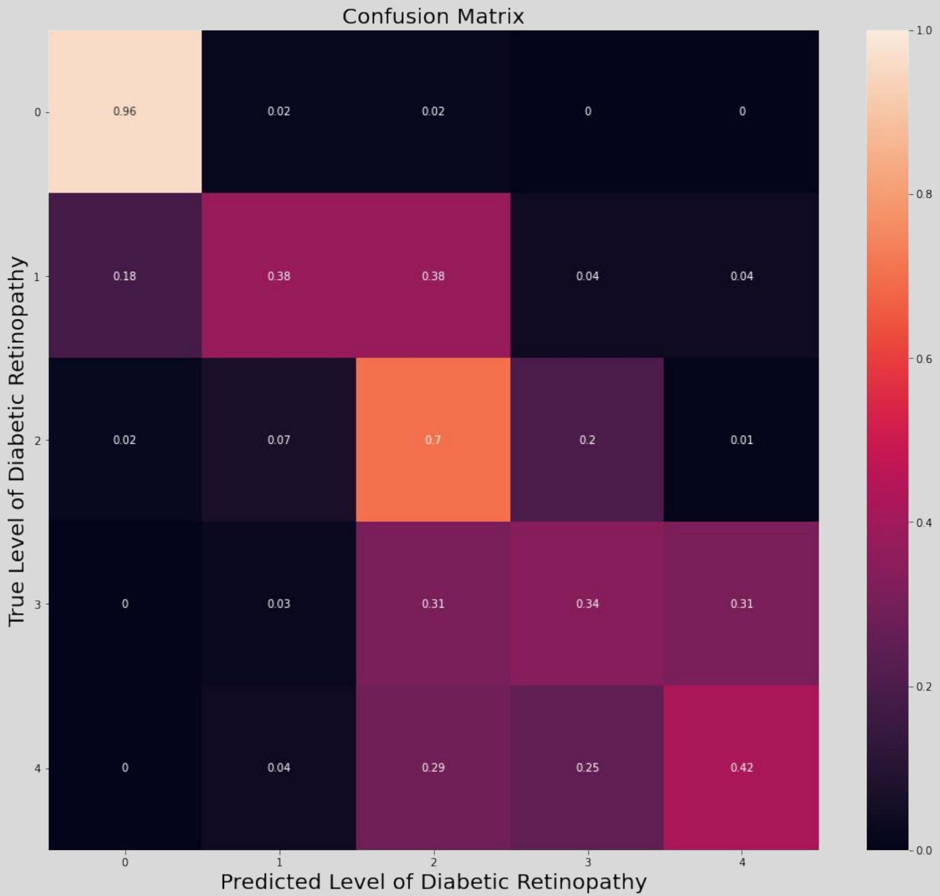
Model Metric Comparison



Transferred Learning Model Conf. Matrix)



Efficient Net Model Conf. Matrix)



Moving Forward

- Double Checking Data
- Getting More Data
- Bigger Image Sizes/Better computers
- KMeans Clustering

Thank you!

https://www.kaggle.com/c/aptos2019-blindness-detection

https://www.nhs.uk/conditions/diabetic-retinopathy/

https://www.kaggle.com/aroraaman/quadratic-kappa-metric-explained-in-5-simple-steps

https://www.kaggle.com/carlolepelaars/efficientnetb5-with-keras-aptos-2019