Automated Diabetic Retinopathy Detection using Deep Learning

Amelia Ritahani Ismail
International Islamic University Malaysia
amelia@iium.edu.my

Muhammad Laziem Shafie
International Islamic University Malaysia
muhdlaziem@gmail.com

Aishah Nabilah Muhamad Ridzuan
International Islamic University Malaysia
ainachmrj@gmail.com

Abstract

Diabetic retinopathy(DR) is an eye disease that occurs in patients with a long period history of diabetes. The diagnosis of DR currently using typical retinal fundus photography that requires and depends on a skilled reader for the manual DR assessment. However, this method opens to the inconsistency of the diagnosis. Thus, Automated Diabetic Retinopathy Detection aims to reduce the burden on ophthalmologists and mitigate diagnostic inconsistencies between manual readers by classifying DR stages using previous DR images with stages labels using Deep Neural Network. The accuracy of the initial model is 0.73 while the final validation accuracy is 0.74 The accuracy of binary prediction (Negative, Positive) is 0.7245 and 0.7268 for validation accuracy. Accuracy of the last model of binary prediction which combines (Mid + Moderate) and (Severe + Proliferative) is 0.8641 and the validation accuracy is 0.8005.

Keywords – Diabetic Retinopathy; machine learning; convolutional neural network; deep learning; multiclass; binary;

I. INTRODUCTION

Diabetes mellitus is a disease that causes high blood glucose level over a prolonged period[1]. It is also commonly known as diabetes and it is estimated about 440 million people in the age-group between 20 to 79 years will be suffering from diabetes mellitus,commonly known as diabetes, by the year 2030 [2]. DR is a disease that occurs in patients with a long period history of diabetes. It happens when high levels of blood sugar damage blood vessels in a part of the eye called the retina. This eye disease starts out with only mild vision problems such as blurriness, however the ignorance of diabetic retinopathy will eventually lead to blindness and it is the most common eye disease among people with diabetes[2]. However, the diagnosis of DR is difficult for patients in the early stage since it relies on the presence of microaneurysms, small saccular outpouching of capillaries, retinal hemorrhages, ruptured blood vessels on the fundoscopic images. It is also hard to diagnose DR requires a skilled reader and i's opens to the inconsistency of the diagnosis. Automated Diabetic Retinopathy Detection Using Deep Learning aims ; (i) to detect diabetic retinopathy level, (ii) to construct diabetic retinopathy into image model, (iii) to evaluate and test the constructed model and (iv) to develop an interface for detecting the diabetic retinopathy of patience.

The rest of the paper is organized as follows: In Section II, the details on diabetic retinopathy; Section III describes the machine learning and algorithm used; Section IV discusses the experimental setup; Section V discusses the finding; and finally, Section VI concludes.

II. DIABETIC RETINOPATHY

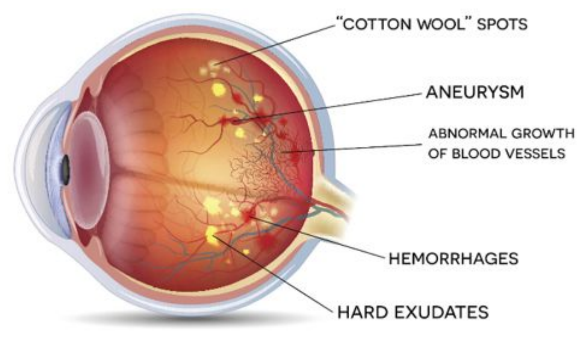

Figure 1: Retina image

***Source**: Getty Images*

DR mainly has been classified into live stages including No apparent retinopathy(Normal), Mild Nonproliferative Retinopathy, Moderate Nonproliferative Retinopathy, Severe Nonproliferative Retinopathy and Severe Nonproliferative

Retinopathy[1] depending on the changes happened in the retina as shown in figure 1.

*Figure 2: Retina image - **Mild** Nonproliferative Retinopathy stage*

During the **Mild** Nonproliferative Retinopathy stage as shown in figure 2, microaneurysms occur. Microaneurysms are the earliest changes of diabetic retinopathy that are clinically visible. They appear as small red dots, balloon-like swelling in the retina's tiny blood vessels which are often in clusters that may cause a leak of fluid into the retina. Dot and blot haemorrhages, flame-shaped haemorrhages and hard exudates also occur as the result of microaneurysms.

*Figure 3: Retina image - **Moderate** Nonproliferative Retinopathy stage*

In the **Moderate** Nonproliferative Retinopathy stage as shown in figure 3, blood vessels that supply blood to the retina swell and block. Cotton wool spots that appear as fluffy white patches on the retina will occur together with microaneurysms, nerve fiber layer haemorrhages (also called flame-shaped haemorrhages) and/or exudates during this stage.

*Figure 4: Retina image - **Severe** Nonproliferative Retinopathy stage*

Next, the stage of **Severe** Nonproliferative Retinopathy as shown in figure 4 occurs when many more blood vessels are blocked, damaging several areas of the retina with their blood supply. Then, Venous beading and *Intraretinal microvascular abnormalities **(IRMA)*** occur as the results of those areas of the retina send signals to the body to grow new blood vessels for blood supply. However, if the blood vessels close off completely, it can lead to blurry vision with dark spots that are often described as "floaters."

*Figure 4: Retina image - **Proliferative** Retinopathy stage*

The advanced stage of **Proliferative** Retinopathy as shown in figure 4, happens when the signals sent by the retina for blood supply trigger the growth of new blood vessels where the new blood vessels are in a form of abnormal and fragile. They will grow in the retina area and along the surface of the clear, vitreous gel that fills the inside of the eye. Luckily, these blood vessels would not lead to vision loss, but their thin and fragile walls may leak and will eventually lead to severe vision loss and even blindness. The disease severity level of the five diabetic retinopathy stages based on the findings in the retina is observable by the ophthalmoscopy which will be performed after dilation of the pupil [1] that allows visualization of the entire retina.

III. MACHINE LEARNING - CNN ALGORITHM

Computers have become more intelligent and able to optimize its performance automatically through experience with its ability to think by using Machine Learning. Machine learning has been used over the past two decades, from research to practical use, and has become the tools in developing practical software for computer vision, speech recognition, natural language processing, robot control, and other applications [3]. One of the algorithms of Machine Learning is Convolutional Neural Network (CNN) that has bloomed in many computer vision tasks.

CNN is partially inspired by neuroscience and is based on neural network architecture, the visual system of the brain, with extra layers which are made to process images [4]. The major difference between CNN and the usual NN, it takes input right away with images without flattening the images while NN needs to flatten the image first as the input. Thus, it reduces tons of parameters during the process.

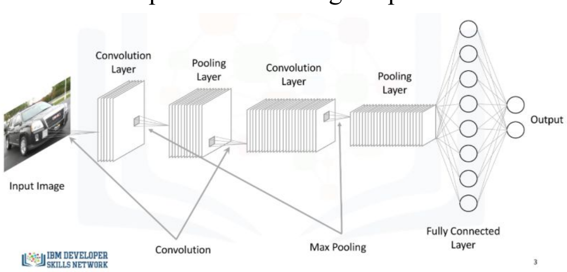

Figure 5: Basic architecture of CNN

Based on Figure 5. It has additional relu layer, convolutional layer and pooling layer. During the input layer, it takes input from (n x m x 1) for grayscale images or (n x m x 3) for color images. [5]

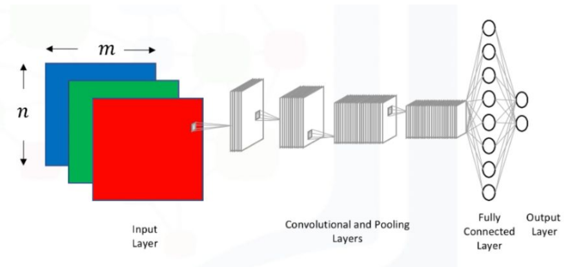

Figure 6: Dimension for colour image

In the convolutional layer, it filters the images by computing the dot product between the images and the filters as shown in figure 6. The more filters used, the more special dimension can be preserved. Convolution's way of handling images saves computing power rather than flatten the input which takes a massive amount of parameters. During this layer, it also contains ReLU layers which filter the convolutional step which only allows positive values to be passed through forward propagation [5].

In the Pooling layer, the main objective of this layer is to reduce the dimension of the images propagating to the network. It has two most popular pooling techniques which are max pooling and average pooling. It takes the maximum or average value for each section of the images and stores them to the desired pooling dimension. Max pooling technique provides spatial variance which enables the NN to recognize objects in an image even if the object does not resemble the original object [5].

After the last convolution and pooling layer, the outputs are flattened to the fully connected layer which connects every node of the current layer with every node of the next layer. It outputs an n-dimensional layer according to the number of classes to work on [5].

IV. EXPERIMENTAL SETUP

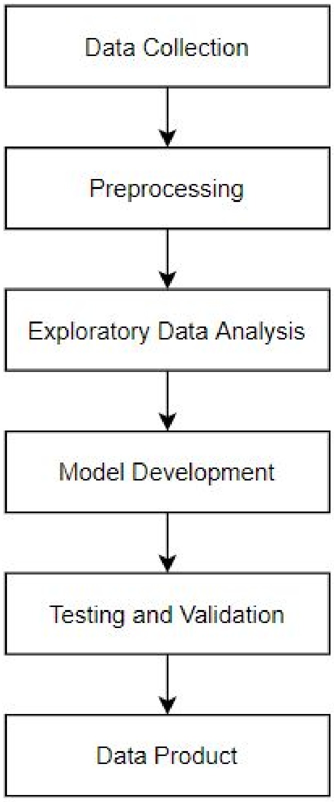

Figure 7: Methodology

A. Data Collection

The first phase of this research are by doing pilot study of Diabetic Retinopathy Detection using kaggle dataset {https://www.kaggle.com/tanlikesmath/diabetic-retinopathy-resized} to create and analyse a good model in preparation for actual data. There are two types of images which are cropped and uncropped images. The cropped images mean the images of DR have been cropped to remove the black area of the images:

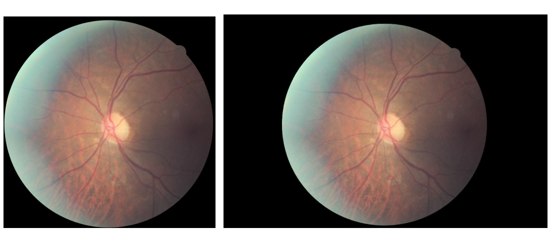

Figure 8 & 9: Cropped image and uncropped image

The left side is a cropped image and the right side an uncropped image.

Table 1: shows the details of datasets distribution classed by the type or level of DR

Level	Type Of DR	Number of Sample	Percentage
0	No	25802	73.49%
2	Mild	2438	6.94%
3	Moderate	5288	15.06%
4	Severe	872	2.48%
5	Proliferative	708	2.02%

B. Data Pre-Processing

Since the dataset provides cropped versions of retinal images. Cropped retinal images are taken to fit in the initial model. However, the size is too big to be fit in the training as to process high resolution images requires much computing power. Therefore, the images need to be resized. The actual size is approximately around 1024x1024x3 pixels. The retinal images are then resized to 64x64x3 pixels on the initial model and 256x256x3 pixels for the binary predictions. Open-cv python is used as a tool to resize the retinal image.

C. Exploratory Data Analysis

Based on Table 1, the normal class of retinal images is way too much that the other classes. In figure 10 below, the different levels of retinal images can be seen. Some are very hard to differentiate and some can be distinguished, for example, the proliferative level of the retinal image, there is something like a blood coat covering the retinal image.

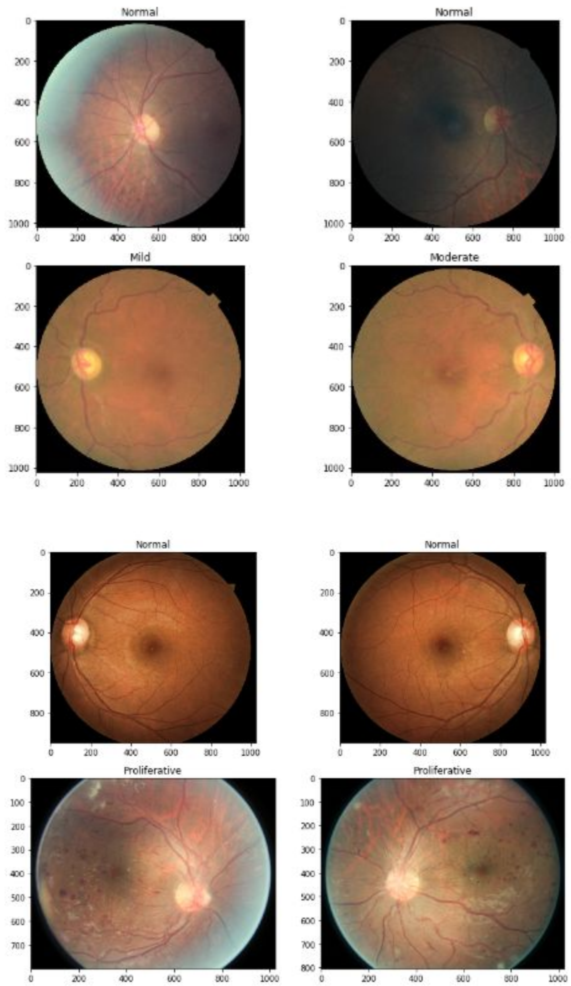

Figure 10: Different levels of retinal image

D. Modelling

The models are done using convolutional neural networks (CNN) which is a derivative of deep neural networks. There are two models tested. The first model is an initial model which classifies into 5 classes and the second one is the model that classifies into two classes which are between Negative and Positive. The positive classes are the combination of the 4 classes (Mild, Moderate, Severe, Proliferative). The 4 classes are then divided into parts which are (Mid + Moderate) and (Severe + Proliferative). The initial model and binary model architecture is as stated in Figure 8 and 9 below.

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 60, 60, 32)        2432
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 30, 30, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 26, 26, 64)        51264
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 13, 13, 64)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 10816)             0
_________________________________________________________________
dense_1 (Dense)              (None, 100)               1081700
_________________________________________________________________
dense_2 (Dense)              (None, 5)                 505
=================================================================
Total params: 1,135,901
Trainable params: 1,135,901
Non-trainable params: 0
```

Figure 11: Architecture of initial model (5 classes)

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 252, 252, 32)      2432
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 126, 126, 32)      0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 122, 122, 64)      51264
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 61, 61, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 57, 57, 128)       204928
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 29, 29, 128)       0
_________________________________________________________________
flatten (Flatten)            (None, 107648)            0
_________________________________________________________________
dense (Dense)                (None, 100)               10764900
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 202
=================================================================
Total params: 11,023,726
Trainable params: 11,023,726
Non-trainable params: 0
```

Figure 12: Architecture of binary model

The model is built using tensorflow, and keras library and with the help of google colab tool.

E. Testing and Validating

The dataset is split into two sets which are training, and testing set on ratio 8:2 for validating purposes.

Table 2 : Table of results

Type	Accuracy	Validation Accuracy	Epochs
5 classes prediction	0.73	0.74	10
2 classes (Negative, Positive)	0.7245	0.7268	10
2 classes (Mid + Moderate) and (Severe + Proliferative)	0.8641	0.8005	40

Based on Table 2, The final accuracy of the initial model is 0.73 while the final validation accuracy is 0.74. The initial model ran with 10 epochs in total. While the (Negative, Positive) accuracy is 0.7245 and validation accuracy is 0.7268 with 10 epochs. Lastly, the (Mid + Moderate) and (Severe + Proliferative) accuracy is 0.8641 and validation accuracy is 0.8005 with 40 epochs.

Performance Validation

Table 3: Confusion Matrix of the initial model

	Normal	Mil	Mod	Sev	Prol
Nor	5212	0	0	0	0
Mil	481	0	0	0	0
Mod	1023	0	0	0	0
Sev	159	0	0	0	0
Prol	147	0	0	0	0

Table 4: Confusion Matrix of the 2 classes (Negative, Positive)

	Actual Negative	Actual Positive
Predict Negative	4645	19
Predict Positive	1733	16

Table 5: Confusion Matrix of the 2 classes (Negative, Positive)

	Actual Negative	Actual Positive
Predict Negative	1372	71
Predict Positive	282	44

Based on table 3, it shows that the levels other than normal tend to learn normal levels rather than learn their own levels which is, the model is not very good enough for deliverability. This problem might occur because imbalance classes on datasets as shown in the distribution of classes on Table 1. Table 4 shows improvement as the distributions are slightly different as the positive class is the combination of the 4 classes as well as Table 5 are little bit outperform the initial model. The number of false positives are increased when using binary mode prediction.

F. Data Product(interface)

A platform(interface) for predicting the Diabetic Retinopathy was built using html, CSS, JavaScript, flask web framework and runs on local hosts.

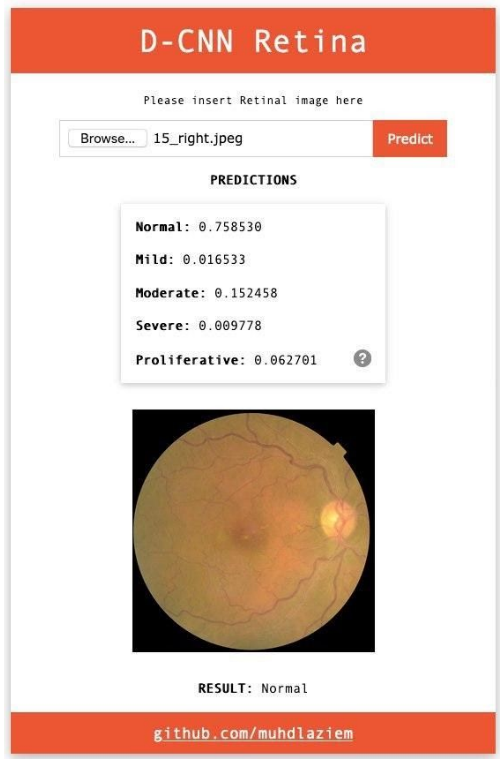

Figure 13: The interface of the system

Figure 13 shows the interface of the model. It works by retrieving a picture of the retina and predicting the level of DR of the image using the model. The image sample shows that it has the highest probability of a normal class with 0.758538 chance. Thus, the model predicted the picture as a normal class.

G. Experiment results

Comparison of results of machine learning models are elaborated in section V.

V. FINDING

The initial result is 0.73 which indicates the model only learns the highest distribution of classes which is Normal. Assuming Mild, Moderate, Severe and Proliferative are in one class, the number of false negatives (FN) is too high. This is a common problem which appears in the healthcare AI industry. A rare but significant healthcare problem or disease contributes huge unstructured datasets which include imbalance classes [6]. The reason this problem persists is because of the rareness of the diseases. Thus, a positive class data is very rare to get. However, this data is very crucial to improve the industry of healthcare.

Based on Table 4 and 5, the number of true positives might be better when using binary prediction method. However, this solution is not enough to counter imbalance classes as the observations appear, the number of false negatives still higher. This is not a good result as in the healthcare industry, false negative is a crucial error. A patient might lose an early prevention because of false negative prediction.

There are several results pertaining the same problems on kaggle which the results as follows:

Table 6: Public Score of Kernels in Kaggle

Kernel	Public Score
https://www.kaggle.com/xwxw2929/starter-kernel-for-0-79	0.798305
https://www.kaggle.com/tanlikesmath/training-on-previous-dataset-for-aptos	0.618139
https://www.kaggle.com/fanconic/efficientnetb3-train-keras	0.743

These public scores show the same range of accuracy of the three types of models. These results can be improved with different techniques which are:

- Over / undersampling
- Synthetic sampling (SMOTE)
- Cost-sensitive learning

Undersampling is a technique which randomly deletes the data that have sufficient number of distributions [7]. In this case, deletes the normal, mild, moderate and severe classes until distribution is in range with proliferative class. This technique might lose important features from the deleted images.

Oversampling is by increasing the imbalance classes randomly by copying the existing samples [7]. This approach can be improved by using data augmentation to prevent duplicate samples. Duplicate samples may lead to overconfident or overfitting models.

Synthetic sampling (SMOTE) is similar to nearest neighbors classification which synthetically produce observations of unbalanced classes [7]. This method has a challenge when the number of observations to an extremely rare class [7]. For example, when the rare class has only one image.

Cost-sensitive learning is taking the loss of model or costs into consideration. Which means, the misclassification (false negative and false positive) cost is also important and taken into the consideration when evaluating the model [8].

These techniques will be included in future works to avoid the model learning only normal / negative class.

VI. CONCLUSION

It is crucial to take good care and treatment regarding this disease as it can cause more and more damage to the eyes. However, there are a lot of weaknesses on the initial model that can be identified. Afterall, that is the reason initial models are conducted, to find what are essential to create a sophisticated model. Imbalance class contributed a lot of drawbacks of models, which made the class other than normal learn normal class instead of their own class. One of the future works on creating a complete model is, to have two separate models, which is classifying 'Normal' and 'DR' first. If it has DR, proceeds to classify the classes of DR.

Further improvement of this research on machine learning models is to test several optimization algorithms to optimize the predictive model. Plus, the techniques of solving imbalance classes will be tested in future. In development of machine learning models, it is crucial that the automated analysis outperform radiologist to radiologist agreement (~80%). Plus, the number of misclassifications need to be reduced, especially false negatives (FN).

VII. REFERENCE

[1] Wang, X., Lu, Y., Wang, Y., & Chen, W. B. (2018, July). Diabetic retinopathy stage classification using convolutional neural networks. In *2018 IEEE International Conference on Information Reuse and Integration (IRI)* (pp. 465-471). IEEE.

[2] Nentwich, M. M., & Ulbig, M. W. (2015). Diabetic retinopathy-ocular complications of diabetes mellitus. *World journal of diabetes*, *6*(3), 489.

[3] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255-260.

[4] Liang, M., & Hu, X. (2015). Recurrent convolutional neural network for object recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3367-3375).

[5] O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.

[6] Zhao, Y., Wong, Z. S. Y., & Tsui, K. L. (2018). A framework of rebalancing imbalanced healthcare data for rare events' classification: a case of look-alike sound-alike mix-up incident detection. Journal of healthcare engineering, 2018.

[7] Chatterjee, S. (2018, August 01). Deep learning unbalanced training data?Solve it like this. Retrieved July 17, 2020, from https://towardsdatascience.com/deep-learning-unbalanced-training-data-solve-it-like-this-6c528e9efea6

[8] Ling, C. X., & Sheng, V. S. (2008). Cost-sensitive learning and the class imbalance problem. Encyclopedia of machine learning, 2011, 231-235.