KULLIYYAH OF INFORMATION AND COMMUNICATION TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

FINAL YEAR PROJECT REPORT

AUTOMATED DIABETIC RETINOPATHY DETECTION USING DEEP LEARNING

MUHAMMAD LAZIEM BIN SHAFIE
1621781

AISHAH NABILAH BINTI MUHAMAD RIDZUAN
1625084

SUPERVISED BY

ASSOC. PROF. DR. AMELIA RITAHANI BINTI ISMAIL

AUGUST 2020
SEMESTER 2 2019/2020

FINAL YEAR PROJECT REPORT

Automated Diabetic Retinopathy Detection using Deep Learning

by

MUHAMMAD LAZIEM BIN SHAFIE

1621781

AISHAH NABILAH BINTI MUHAMAD RIDZUAN

1625084

SUPERVISED BY

ASSOC. PROF. DR. AMELIA RITAHANI BINTI ISMAIL

In partial fulfillment of the requirement for the

Bachelor of Computer Science

Department of Computer Science

Kulliyyah of Information and Communication Technology

International Islamic University Malaysia

AUGUST 2020

Semester 2 2019/2020

ACKNOWLEDGEMENTS

Praise and thanks to Allah first and foremost whose blessing enabled me to accomplish this project. We would like to express our special thanks and deepest appreciation of gratitude to our respected and beloved supervisor Assoc. Prof. Dr. Amelia Rithani Binti Ismail for giving us this golden opportunity to do such an amazing and interesting research on the topic Automated Diabetic Retinopathy Detection using Convolutional Neural Network, which also helped and guided us in doing this Research and model that made us knew about so many new things we are really thankful to her and her amazing personality. We wish to thank Asst. Prof. Dr. Norzariyah Binti Yahya, our Final Year Project coordinator, all lectures and staff of International Islamic University Malaysia that help and encourage us with warm hearts directly or indirectly in accomplishing this project. We also would like to express our gratitude to the website specifically kaggle.com and ilovescience (kaggle username) for preparing and providing open datasets and a good kernel so that we can experiment and do this project on it. We would also like to thank our families and friends who helped and support us and ease us in completing this project in time.

ABSTRACT

Diabetic Retinopathy (DR) is an eye 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 irreversible blindness; it is the most common eye disease among people with 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. 4 different experiments were conducted resulting in the best result of 0.82 accuracy, 0.80-0.85 precision, 0.8205 Cohen's kappa score with 0.5 threshold. The best model deployed in a web application.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
DR	Diabetic Retinopathy
DenseNet	Densely Connected Convolutional Networks
IRMA	Intraretinal Microvascular Abnormalities
NN	Neural Networks
GPU	Graphic Processing Unit
AlexNet	Alex Convolutional Neural Network
VGG16	Visual Geometry Group 16
GoogleNet	Google Convolutional Neural Network
vtDR	Vision threatening Diabetic Retinopathy
CE	Conformité Européenne
Ltd	Limited
AM-FM	Amplitude Modulation-Frequency Modulation
DR-RACS	Diabetic Retinopathy Royal Australasian College of Surgeons
SELENA	Singapore Eye Lesion Analyzer
AI	Artificial Intelligence
Colab	Colaboratory

HTML	Hypertext Markup Language
LaTeX	Layout Tex
TPU	Tensor Processing Unit
LTS	Long Term Support
VM	Virtual Machine
OpenCV	Open Source Computer Vision Library
APTOS	Asia Pacific Tele-Ophthalmology Society
BGR, RGB	Blue Green Red, Red Green Blue
val	Validation
jpg	Joint Photographic Experts Group
Restful API	Representational State Transfer Application Programming Interface

CHAPTER ONE

INTRODUCTION

1.1 Background

Diabetes mellitus is a disease that causes high blood glucose level over a prolonged period according to Wang, X., Lu, Y., Wang, Y., & Chen & W. B. (2018). According to Nentwich, M. M., & Ulbig, M. W. (2015), 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 . 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. Nentwich et al (2015) also mentioned 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. 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.

1.2 Problem Statement

The diagnosis of diabetic retinopathy(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.

1.3 Project Objective

1. To detect diabetic retinopathy level.
2. To construct diabetic retinopathy into image model
3. To evaluate and test the constructed model.
4. To develop an interface for detecting the diabetic retinopathy of patience.

1.4 Project Scope

This research aims to develop an automated diabetic retinopathy system that is able to detect the level (severity) of diabetic retinopathy.

1.5 Significance of The Project

Diabetic Retinopathy(DR) diagnosis currently using typical retinal fundus photography that requires and depends on a skilled reader for the manual DR assessment. However, according to Goh et al. (2016), this method opens to the inconsistency of the diagnosis. Therefore, Automated Diabetic Retinopathy Detection

aims to reduce the burden on ophthalmologists and mitigate diagnostic inconsistencies between manual readers.

CHAPTER TWO

LITERATURE REVIEW

This chapter includes literature review on diabetic retinopathy, reviews of other previous researchers and the techniques and methods they used in developing diabetic retinopathy detection models. These researches will help to grasp the knowledge to achieve the project's objectives.

2.1 Diabetic Retinopathy

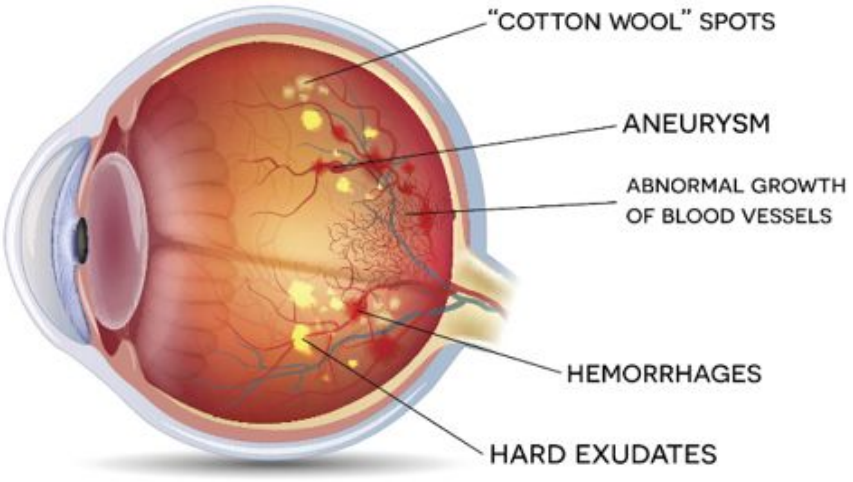
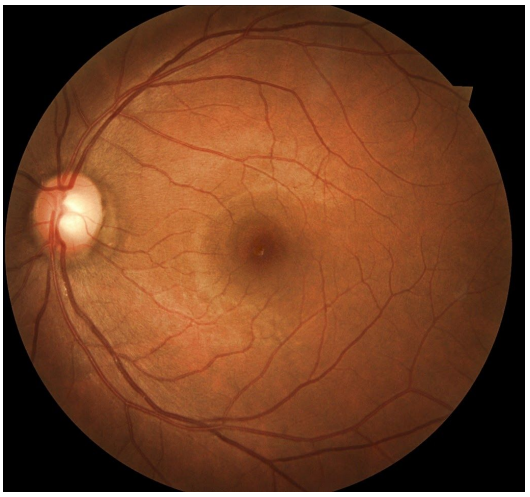
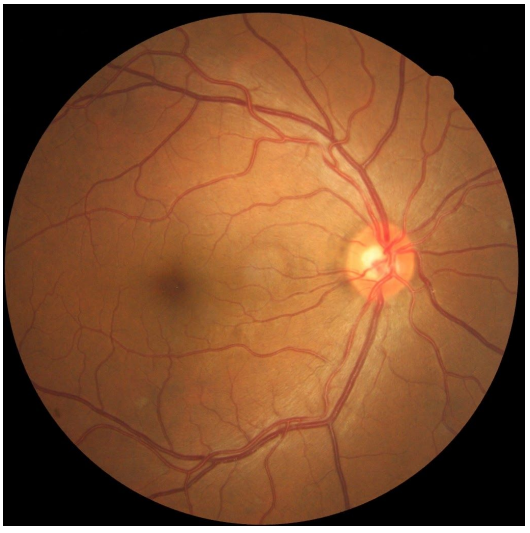


Figure 1: Retinal image. ***Source****: Getty Images*

According Wang et al (2018) DR mainly has been classified into five stages including No apparent retinopathy (Normal), Mild Nonproliferative Retinopathy, Moderate Nonproliferative Retinopathy, Severe Nonproliferative Retinopathy and Severe Nonproliferative Retinopathy depending on the changes happened in the retina as shown in figure 1.

Table 1 : Retinal image

Level	Retinal image	Abnormalities
0	 *Figure 2: No apparent retinopathy (Normal)*	- No
1	 *Figure 3 : **Mild** Nonproliferative Retinopathy stage*	- Microaneurysms - Dot and blot haemorrhages - Flame-shaped haemorrhages - hard exudates

2	 *Figure 4: **Moderate** Nonproliferative Retinopathy stage*	- Blood vessels in the retina swell and block - Cotton wool spots - Flame-shaped hemorrhage - Exudates
3	 *Figure 5: **Severe** Nonproliferative Retinopathy stage*	- *Intraretinal microvascular abnormalities (**IRMA**)* - Blood vessels are blocked
4	 *Figure 6: **Proliferative** Retinopathy stage*	- *New blood vessels(*abnormal and fragile) - Blurriness and vision loss

During the **Mild** Nonproliferative Retinopathy stage as shown in figure 3, 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.

In the **Moderate** Nonproliferative Retinopathy stage as shown in figure 4, 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.

Next, the stage of **Severe** Nonproliferative Retinopathy as shown in figure 5 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."

The advanced stage of **Proliferative** Retinopathy as shown in figure 6, 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. According Wang et al (2018) 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 that allows visualization of the entire retina.

2.2 Machine Learning

Computers have become more intelligent and able to optimize its performance automatically through experience with its ability to think by using Machine Learning. According to Jordan, M. I., & Mitchell, T. M. (2015), 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 . One of the algorithms of Machine Learning is Convolutional Neural Network (CNN) that has bloomed in many computer vision tasks

2.3 Convolutional Neural Network

Convolutional Neural Networks (CNN) are made up of neurons that have learned able weights and biases. It is very similar to ordinary Neural Networks. Liang, M., & Hu, X. (2015) mentioned, it 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. 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.

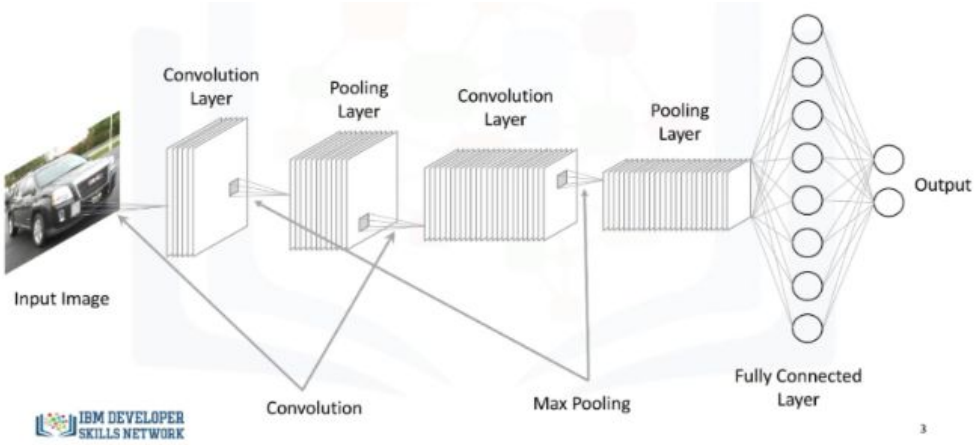


Figure 7: Basic architecture of CNN

Based on Figure 7. It has additional relu layer, convolutional layer and pooling layer. Mentioned by O'Shea, K., & Nash, R. (2015), During the input layer, it takes input from (n x m x 1) for grayscale images or (n x m x 3) for colour images.

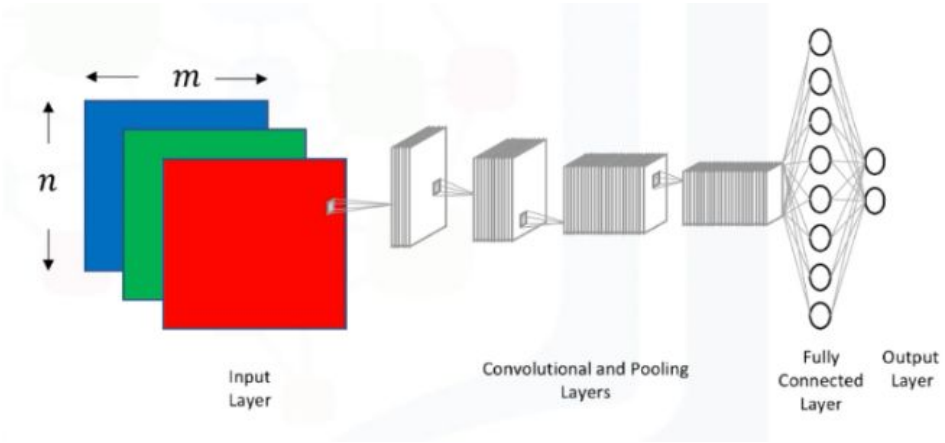


Figure 8: Dimension for colour image

In the convolutional layer, it filters the images by computing the dot product between the images and the filters as shown in figure 8. 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. O'Shea, K. et al (2015) mentioned again, during this layer, it also contains ReLU layers which filter the convolutional step which only allows positive values to be passed through forward propagation.

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. By referring to O'Shea, K. et al (2015), 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.

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 according to O'Shea, K. et al (2015).

2.4 Review of Previous Works

Doshi, Shenoy, Sidhpura, & Gharpure(2017) in their paper explained the automatic diagnosis of the disease into its several stages using deep learning. Doshi et al. presents the design and implementation of GPU accelerated deep convolutional neural networks to automatically diagnose and thereby classify

high-resolution retinal images into 5 stages of the disease based on severity. However, the single model that Doshi et al. created only produced accuracy of 0.386 on a quadratic weighted kappa metric and assembling three such similar models resulted in a score of 0.3996.

Lam, Yi, Guo, & Lindsey (2018) also use a deep convolutional neural network (CNN) but with a different approach of classification. They examined binary, 3-ary and 4-ary label classification to examine the weakness of CNN. They are using several CNN models, for example, AlexNet, VGG16 and GoogleNet. They found that binary models (normal or mild, moderate to end stage) achieved more accuracy and sensitivity than multiclass training. This might come from the noises from Kaggle images or low fidelity of labelling the images. These are the results:

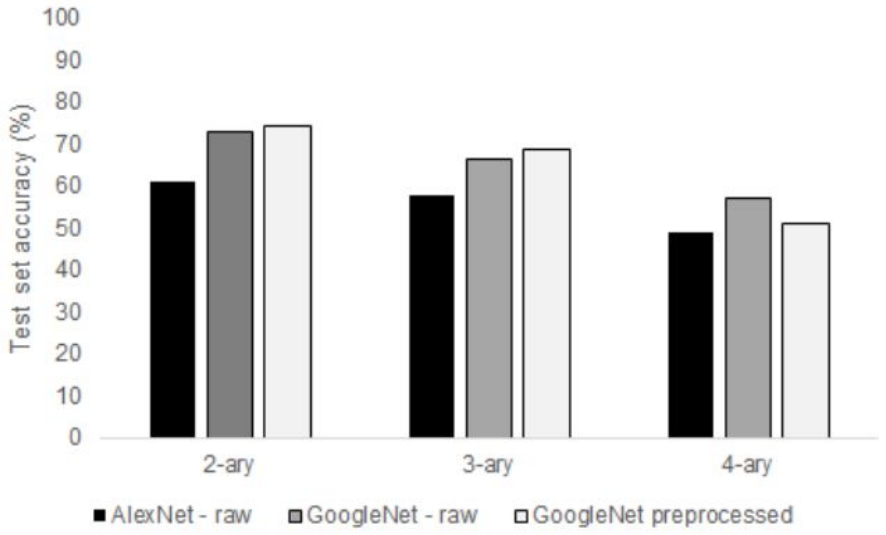


Figure 9 : Graph of accuracy based on model and number of classes

Abràmoff et al., (2016) doing “Improved Automated Detection of Diabetic Retinopathy on a Publicly Available Dataset Through Integration of Deep Learning”. They were working on three classes {No DR, Vision threatening DR (vtDR), Macular edema}. They used AlexNet CNN model to train and managed to get this results:

IDx Output For	Disease Level	Sensitivity (95% CI)	Specificity (95% CI)	Negative Predictive Value (95% CI)	Positive Predictive Value (95% CI)	AUC (95% CI)
rDR	rDR	96.8% (93.3%–98.8%)	87.0% (84.2%–89.4%)	99.0% (97.8%–99.6%)	67.4% (61.5%–72.9%)	0.980 (0.968, 0.992)
rDR	vtDR	100% (96.1%–100%)	N.A.	N.A.	N.A.	N.A.
rDR	ME	100% (95.6%–100%)	N.A.	N.A.	N.A.	N.A.
vtDR	vtDR	100.0% (96.1%–100.0%)	90.8% (88.5%–92.7%)	100.0% (99.5%–100.0%)	56.4% (48.4%–64.1%)	0.989 (0.984, 0.994)

N.A., not calculated.

Figure 10 : The Result of models

The results are good with the integration of CNN. A working medical device, iGradingM, had received its class 1 Conformité Européenne (CE) mark in 2013 and being used in Scottish Diabetic Retinopathy Screening Program (Sim et al., 2015). It is a product produced by Medalytix Group Ltd. However, the device could only detect diabetic retinopathy absent and diabetic retinopathy present instead of specifying the severity of the disease condition.

There are also a few companies that produce a system regarding this disease as shown in table 2 below, for example DR-RACS system is built based on Amplitude modulation-frequency modulation(AM-FM), k-means & partial least square classifier to predict the risk(low/high) for DR. Retmarker DR also produces a system that is able to detect the presence or absence of DR using longitudinal analysis by

comparing with baseline image and Singapore Eye Lesion Analyzer (SELENA) produces a system.

Table 2 : List of systems

SYSTEM	Algorithm & Strength	OUTCOME
DR-RACS	Amplitude modulation-frequency modulation(AM-FM), k-means & partial least square classifier	Low risk / high risk for DR
Retmarker DR	Longitudinal analysis by comparing with baseline image	Presence/Absence of DR : microaneurysm turnover
Singapore Eye Lesion Analyzer (SELENA)	Deep learning technology using CNN and region extraction	Grade of DR and referable / nonreferable

CHAPTER THREE

METHODOLOGY

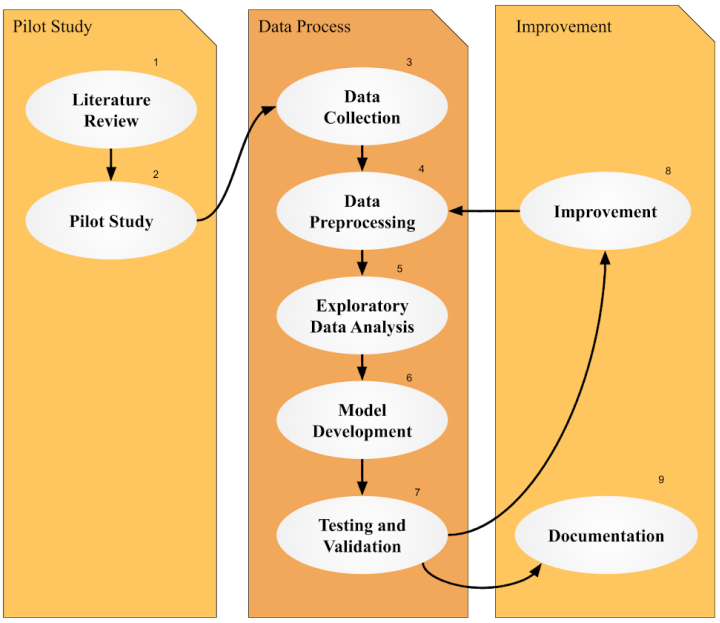


Figure 11 : Methodology

3.1 Environment

The whole environment is relying on Colaboratory or Colab in short. Colab is a platform mainly for students, data scientists or AI researchers. Colab notebook is basically a jupyter notebook that is hosted by Colab. It allows users to make executable code alongside rich text, images, HTML, LaTeX et cetera in one document. The difference between Colab and jupyter is Colab can easily be shared to others as it saves the document in Google Drive. The good things are Colab gives users free cloud GPU and cloud TPU with some limit of usages. It integrates with Google Cloud with a range of GPU types are NVIDIA K80, P100, P4, T4, and V100. So, Colab

basically will create a VM running Ubuntu 18.04.3 LTS alongside with proprietary drivers in it.

The product of this project is using the Flask web framework to create a Restful API with the integration of tensorflow, and OpenCV for deployment.

3.2 Dataset

The dataset of Diabetic Retinopathy was obtained from Kaggle, an open sources platform used by an online community of data scientists. The dataset is cropped and resized with maximum 1024x1024x3 pixels in size. The dataset contains 35108 unique images with 5 levels of Diabetic Retinopathy which are No apparent retinopathy (Normal), Mild Nonproliferative Retinopathy, Moderate Nonproliferative Retinopathy, Severe Nonproliferative Retinopathy and Severe Nonproliferative Retinopathy. Details distribution of datasets is explained in table below :

Table 3 : Dataset distribution details

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

Table 4 : Testing dataset distribution details

Level	Type of DR	No. of Sample
0	Normal	20
1	Moderate	20
2	Severe	20

3.3 Preprocessing

Useful information from images need to be extracted in order to train the model well. Since 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 1024x1024 pixels. The retinal images are then resized to 64x64 pixels using open-cv python to resize the retinal image.

a) Image Resizing

Due to resource limitation 320x320x3 pixels are chosen to fit in the model.

b) Ben's Image preprocessing

This technique is very popular in APTOS competition in which Ben Graham won the previous APTOS competition in 2019. The reasons of behind this preprocessing are:

- Enhance finer details (thin strokes in retinal images)
- To tackle different lighting or illumination backgrounds. (Background color differences)

The technique which is applied is first by converting BGR color to RGB color which is a common problem when importing an image with opencv, the color is reversed. Then, image smoothing is applied using Gaussian Blur.

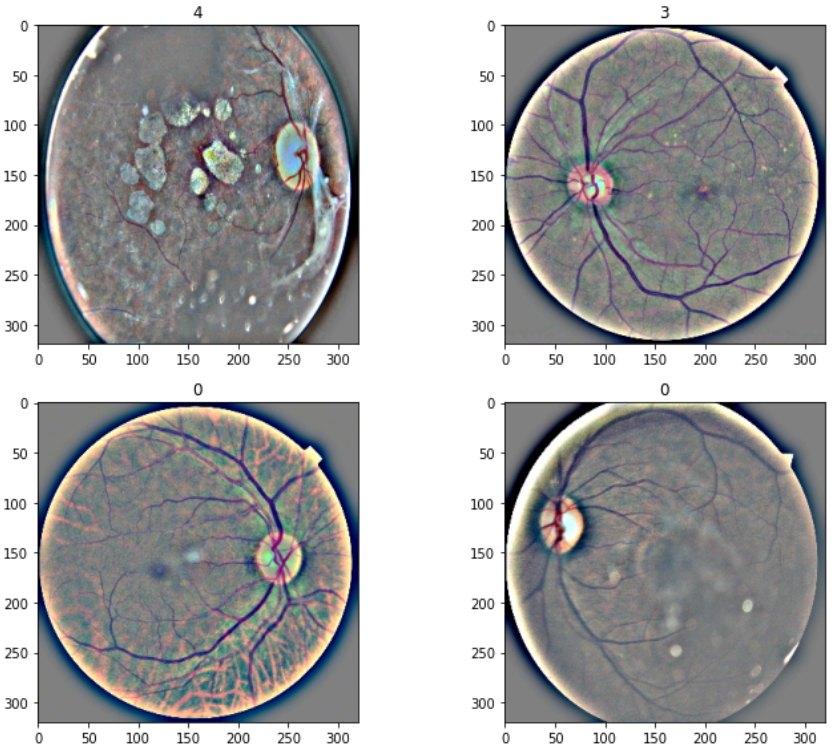


Figure 12 : Ben's Preprocessing and Resizing

c) Data Augmentation

Data augmentation will be applied in the training set to increase the amount and diversity of the images in the dataset. Example of data augmentation is flipping, rotating, mirroring, zooming et cetera the images.

d) Sampling and Undersampling

The sampling itself is a basic sampling which randomly selects from a group population, in this case a group of images. While undersampling is one of the techniques to solve imbalance in the dataset. According to table 3, it is clear the

dataset is very unbalanced which Normal level is approximately 73% of the dataset. Undersampling is a technique where the majority levels will be removed to balance with the minority level which Proliferative level. In fact,undersampling is one of the ways to solve unbalanced dataset. However, important features might be lost, the number of samples is decreasing.

3.4 Keras

An open source neural network library written in Python which runs on top of tensorflow. It is designed to be able to do fast experimentation deep learning models.

3.5 Algorithm

The main algorithm used in this project is DenseNet-121. According to Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017) in their paper mentioned DenseNet or Densely Connected Convolutional Networks is an algorithm which connects each layer to every layer in the algorithm during the feed-forward process. Traditional convolutional neural networks have L layers with L connections while DenseNet has L(L+1)/2 direct networks. Below are some of advantages of the DenseNet algorithm according to Huang G. et al (2017):

- Alleviate vanishing gradient problem
- strengthen feature propagation
- encourage feature reuse
- reduce number of parameters

The simple illustration of DenseNet can be shown in Figure 13

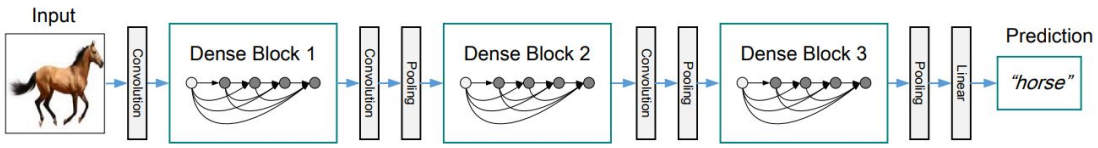


Figure 13 : Simple Illustration of DenseNet retrieved from https://www.kaggle.com/pytorch/densenet121

DenseNet-121 is just another derivation of DenseNet. The 121 notation represents the number of layers. The full description of differences in DenseNet architecture can be shown in Table 5

Table 5 : Architecture of DenseNet retrieved from https://www.kaggle.com/pytorch/densenet121

Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112 × 112	7 × 7 conv, stride 2			
Pooling	56 × 56	3 × 3 max pool, stride 2			
Dense Block (1)	56 × 56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56 × 56	1 × 1 conv			
	28 × 28	2 × 2 average pool, stride 2			
Dense Block (2)	28 × 28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28 × 28	1 × 1 conv			
	14 × 14	2 × 2 average pool, stride 2			
Dense Block (3)	14 × 14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	14 × 14	1 × 1 conv			
	7 × 7	2 × 2 average pool, stride 2			
Dense Block (4)	7 × 7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	1 × 1	7 × 7 global average pool			
		1000D fully-connected, softmax			

This project will layer until Dense Block 4 only. The classification layer is customized according to the total number of classes in this project which is 5.

a) Cohen's Kappa Score

According to Pykes, K (2020) in his article, Cohen's kappa measures the agreement between two raters who each classify N items into a number of mutually exclusive categories. The range is between 0 - 1.

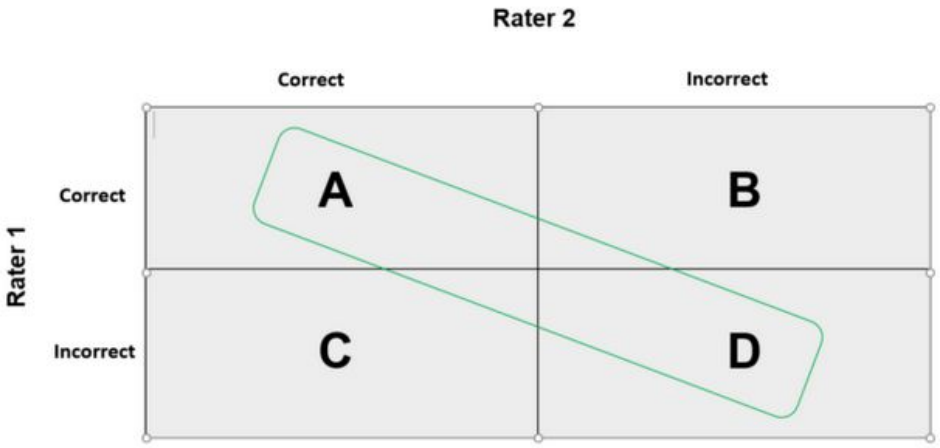


Figure 14 : N X N grid used to evaluate results of raters retrieved from Pykes, K (2020)

Given the grid in figure 14. The explanation A, B, C, D are as follows:

- A - The number of votes that the raters say correct
- B - The number of votes that the rater 1 says incorrect but rater 2 says correct
- C - The vice versa of B, the number of votes that the rater 1 says correct but rater 2 says incorrect
- D - the vice versa of A, The number of votes that the raters say incorrect

$$K = P_o - P_e / 1 - P_e$$

Figure 15 : Cohen's Kappa coefficient formula retrieved from Pykes, K (2020)

$$P_o = \text{Number in Agreement / Total}$$

Figure 16 : Probability of agreement retrieved from Pykes, K (2020)

$$P_{(correct)} = (A + B/ A + B + C + D) * (A + C/ A + B + C + D)$$

$$P_{(incorrect)} = (C + D/ A + B + C + D) * (B + D/ A + B + C + D)$$

$$P_e = P_{(correct)} + P_{(incorrect)}$$

Figure 17 : Formula to service probability of random agreement retrieved from Pykes, K (2020)

Figure 15 to 17 shows the formula how to calculate Cohen's Kappa score. It can be seen as a figure 14 is the same as a confusion matrix. The project will be evaluating each model created using this coefficient.

b) Sigmoid

Sigmoid activation function will be used in the neural network. The derivation sigmoid function is "**sigmoid(x) = 1 / (1 + exp(-x))"** where exp is exponential. The main reason this project is using sigmoid activation function is as sigmoid function is outputting 0 - 1 in range of probabilities.

c) Binary Cross Entropy

Since this project is using multiple binary classification to predict multilabel which will be explained in 3.6. This function will be used to calculate the loss of the predictions. The formula of binary cross entropy function is shown in figure 18

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i))$$

Figure 18 : Binary Cross-Entropy retrieved from Godoy, D. (2019)

In figure 18, y is the actual label and p(y) is the predicted probability.

3.6 Multi Labels

A usual one hot encoding label of level 4 (Proliferative) is [0, 0 , 0, 1]. Multi labels will change the format to [1, 1, 1, 1, 1] for level 4. A sigmoid function will be applied to predict these multi labels. So, in conclusion, multiple binary classification will be done and produce a probability for each label to be 1 within a range of 0 and 1. Then, a threshold will be applied to determine whether it is 0 or 1 for each one hot encoding explained above. For example, the output is [0.9678, 0.876, 0.677, 0.123, 0.34] and the threshold is 0.5. The result will be level 2 (Moderate). Rather than predicting only one label, this technique does multiple binary classification. This technique is inspired from Lex's kernel .

3.7 Training and Validation Phase

The dataset is splitted into two parts which is training and validation with a ratio of 7:3 .

3.8 Testing Phase

A testing set is prepared separately with 60 retinal images in it. Every image is labelled within 3 levels of Diabetic Retinopathy which are Normal , Moderate and Severe.

CHAPTER FOUR

RESULTS AND ANALYSIS

Four different experiments are done to compare the result. The experiments are as stated in Table 6.

Table 6 : Details of experiment setup

Experiment	Method	No classes
1	15000 random sampling from the dataset	5
2		3
3	Undersampling technique	5
4		3

The experiment was done by searching the best Cohen's kappa score during training and validating phase. When Cohen's kappa score is increased in a particular epoch, the model will be saved. The training phase is divided into 8 buckets. Each bucket has a number of epochs which are explained in Table 7

Table 7 : Number of Epochs in each bucket

Bucket	Number of Epoch
1	5
2	5
3	10
4	15
5	15
6	20
7	20
8	30

This bucket approach is inspired by Federico Raimondi's Kernel.

4.1 Experiment 1

Experiment 1 was done by random sampling 15000 retinal images from the dataset with 5 classes resulting in the distribution in Table 8.

Table 8 : Distribution for Experiment 1

Level	Total
0	11052
1	1011
2	2251
3	390
4	296

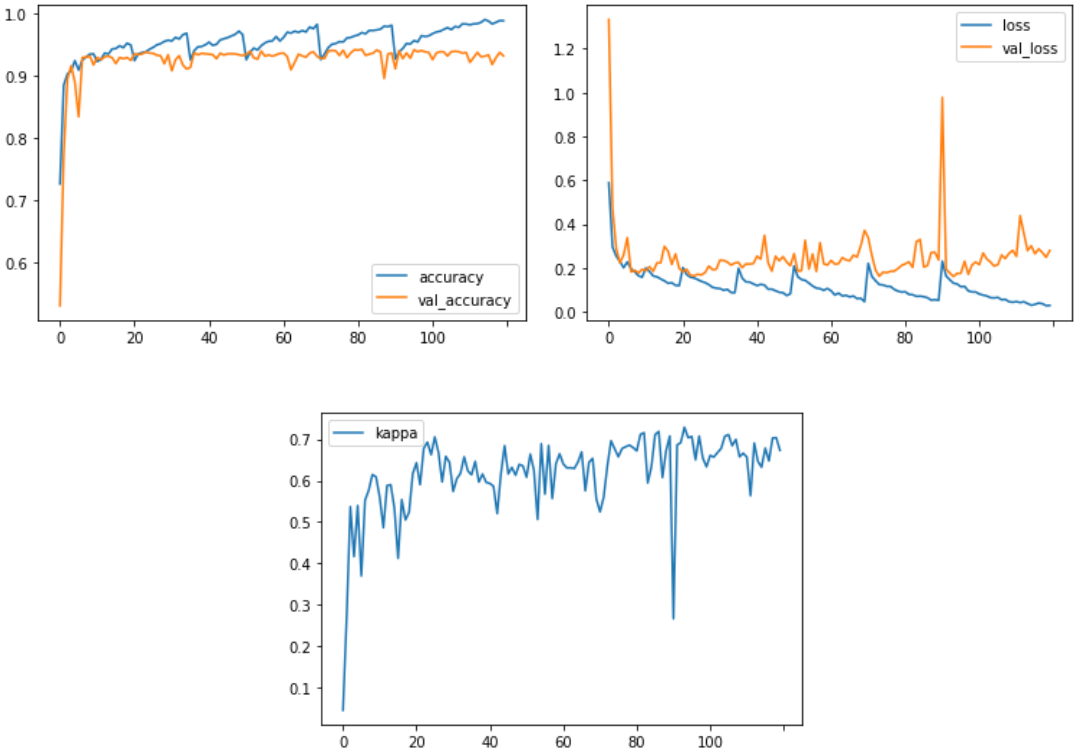


Figure 19 : Graphs of Multiple Binary Classification in Experiment 1

Graph 19 showed the result of Multiple Binary Classification for Experiment 1. The accuracy and val_accuray is increasing over the epochs and in contrary, loss and val _loss is decreasing over the epochs. All the graphs are somehow fluctuated when entering a new bucket and increasing through the new bucket. However, as mentioned Chapter 3.6, this is a multiple binary classification. So the real result is in the Validation test and Testing test.

Table 9 : Confusion Matrix for Validation Set in Experiment 1

		Actual				
		0	1	2	3	4
Prediction	0	3092	231	191	8	5
	1	101	45	48	1	0
	2	110	29	307	37	19
	3	5	1	90	63	15
	4	10	1	22	20	49

Table 10 : Precision for each level for Validation Set and the accuracy in Experiment 1

Level	Precision
0	0.93
1	0.15
2	0.47
3	0.49
4	0.56
Accuracy	**0.79**
Best Cohen's Kappa Score	**0.7333**
Best Threshold	**0.5271**

Based on Table 9, the red box indicates the false negative area. It is quite high in Mild class and decreasing to Proliferative class with a total of 435 out of 3948(class 1 to 4), which is 11.02% of retinal images. Furthermore, in Table 10, it clears that

level 0 has the highest precision which the model predicts level 0 better than other levels. The accuracy is high naturally because the distribution of normal class is approximately 73.49% from 15000 retinal images. However, Cohen's kappa score is quite high which is 0.7333, which is the probability of the false prediction to be true prediction is high. The best threshold is above the average which is over 0.5.

Table 11 : Confusion Matrix for Testing Set in Experiment 1

		Actual		
		0	1	2
Prediction	0	16	5	0
	1	4	14	4
	2	0	1	16

Table 12 : Precision for each level for Testing Set and the accuracy in Experiment 1

Level	Precision
0	0.80
1	0.70
2	0.80
Accuracy	**0.77**
Best Cohen's Kappa Score	**0.8205**
Best Threshold	**0.5063**

It was noted that in Experiment 1 the model was trained with 5 classes. So, during the testing phase, the result of Moderate level will be added to Mild level and

the result of Proliferative will be added to Severe level which will be 3 levels prediction.

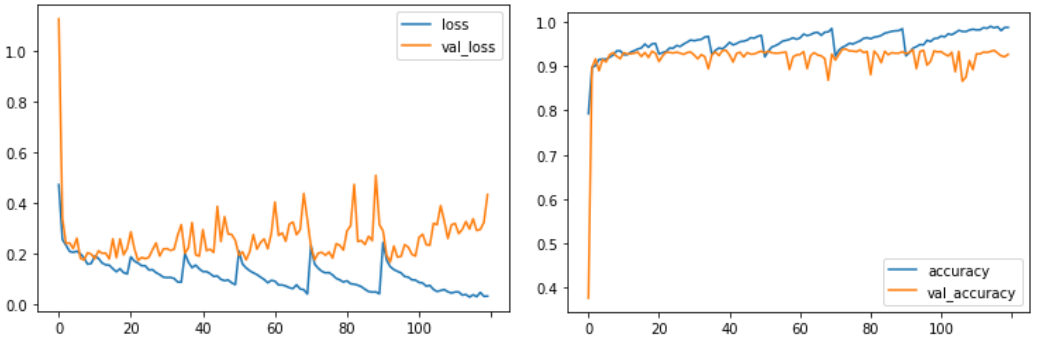
Based on Table 11 , 5 out of 40(class 1 and 2) retinal images are false negative. On Table 12, it can be concluded each level has very good precision from 0.7 - 0.8 in range. It means, the model predicts well on each level. The accuracy of the testing set is lower than the validation set, however the best Cohen's kappa score is 0.8205 is a bit higher than the validation set. The best threshold for the testing set is a bit lower than the validation set.

4.2 Experiment 2

Experiment 2 was done using random sampling with 15000 retinal images from the dataset containing 3 classes resulting in the distribution in Table 13

Table 13 : Distribution for Experiment 2

Level	Total
0	11052
1	3262
2	686



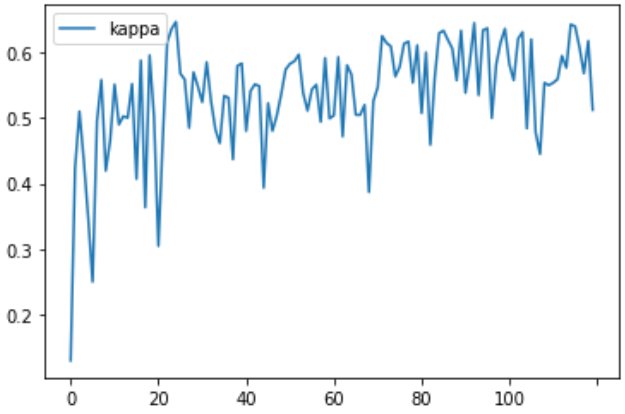


Figure 20 : Graphs of Multiple Binary Classification in Experiment 2

Figure 20 showed the result of Multiple Binary Classification for Experiment 2. The accuracy and val_accuray is increasing over the epochs and in contrary, loss and val _loss is decreasing over the epochs. Plus, fluctuation entering a new bucket happened.

Table 14 : Confusion Matrix for Validation Set in Experiment 2

		Actual		
		0	1	2
Prediction	0	3084	499	17
	1	228	371	54
	2	6	95	146

Table 15 : Precision for each level for Validation Set and the accuracy in Experiment 2

Level	Precision
0	0.93
1	0.38
2	0.67
Accuracy	**0.80**
Best Cohen's Kappa Score	**0.6473**
Best Threshold	**0.5359**

Based on Table 14, the number of false negatives is quite high in Moderate class, together with severe class makes a total of 516 out of 3948(class 1 and 2), which is 13.07% of the retinal images. Furthermore, in Table 15, it shows that level 0 has the highest precision which the model predicts level 0 better compared to other levels.

Same with experiment 1, the accuracy is high naturally because the distribution of normal class is approximately 73.49% from 15000 retinal images. However, when Cohen's kappa score is quite low which is 0.6473, the probability of the false prediction to be true prediction is low. The best threshold is above the average which is over 0.5.

Table 16 : Confusion Matrix for Testing Set in Experiment 2

		Actual		
		0	1	2
Prediction	0	17	3	1
	1	3	16	3
	2	0	1	16

Table 17 : Precision for each level for Testing Set and the accuracy in Experiment 2

Level	Precision
0	0.85
1	0.80
2	0.80
Accuracy	**0.82**
Best Cohen's Kappa Score	**0.8205**
Best Threshold	**0.5**

Based on Table 16, 4 out of 40(class 1 and 2) retinal images are false negative which are 3 from the Moderate class and 1 from the Severe class. On Table 17, it can be concluded each level has very good precision in range 0f 0.8 - 0.85. It means, the model predicts well on each level. The accuracy of the testing set is slightly higher than the validation set, and the best Cohen's kappa score is 0.8205 which is higher than the validation set. The best threshold for the testing set is a bit lower than the validation set.

4.3 Experiment 3

Experiment was done using undersampling technique which reduces the majority class to be levelled with other classes as mentioned in 3.3.d. The new distribution of the dataset is in Table 18.

Table 18 : Distribution for Experiment 3

Level	Total
0	800
1	753
2	739
3	710
4	708
Total	**3710**

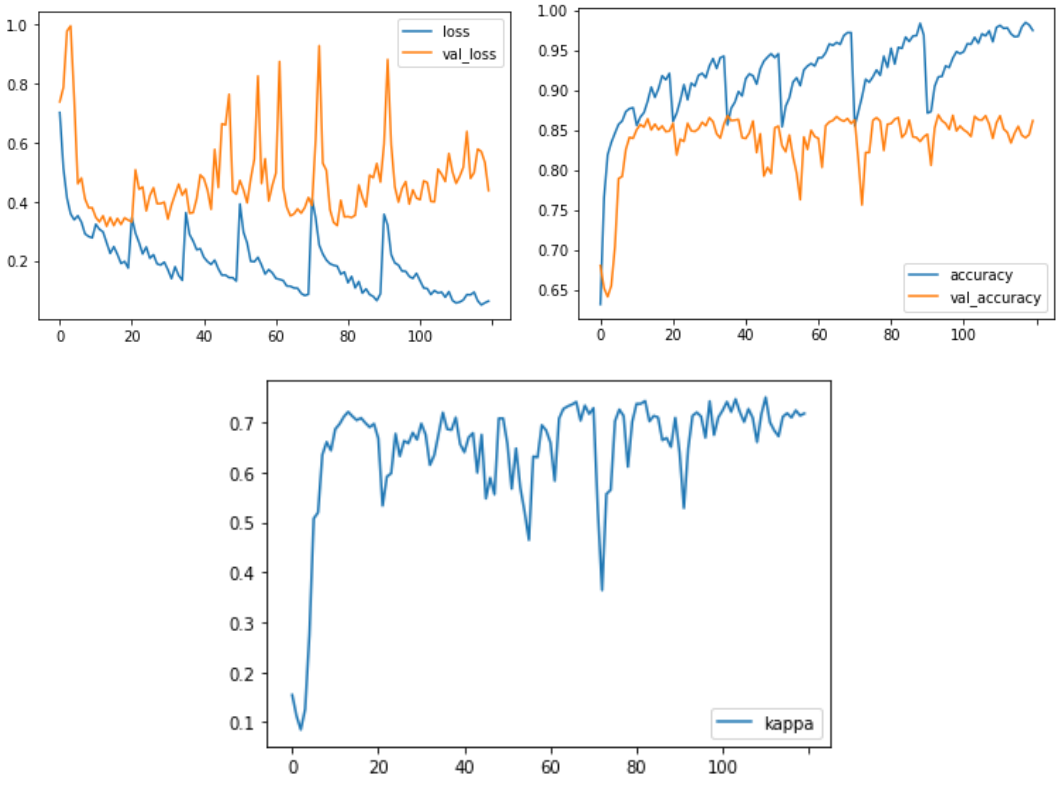


Figure 21 : Graphs of Multiple Binary Classification in Experiment 3

Figure 21 showed that all the results from multiple binary classification during training phase in experiment 3. Same from the previous experiment, the graphs are a bit fluctuated when entering a new bucket. It can be seen, at the end of the last bucket, the accuracy is near 1.00. This is just experimental as this project is focussing on Cohen's kappa score. The real results always depend on the validation set and testing set.

Table 19: Confusion Matrix for Validation Set in Experiment 3

		Actual				
		0	1	2	3	4
Prediction	0	123	114	32	6	1
	1	95	111	73	20	8
	2	9	11	47	14	15
	3	6	5	60	142	79
	4	5	2	9	11	115

Table 20 : Precision for each level for Validation Set and the accuracy in Experiment 3

Level	Precision
0	0.52
1	0.46
2	0.21
3	0.74
4	0.53
Accuracy	**0.48**
Best Cohen's Kappa Score	**0.7523**
Best Threshold	**0.5125**

Table 19 and table 20 show the result from the validation set. 153 out of 2910 (5.258%) are false negative which is a bit low. However, the accuracy of the validation set is very low which is only 0.48. Plus, the precision of each level is very low, only level 3 is a bit high which is 0.74. Nevertheless, the best Cohen's kappa score is quite promising which is 0.7523. While the best threshold is still within the range of the normal. threshold.

Table 21 : Confusion Matrix for Testing Set in Experiment 3

		Actual		
		0	1	2
Prediction	0	1	0	0
	1	16	14	1
	2	3	6	19

Table 22 : Precision for each level for Testing Set and the accuracy in Experiment 3

Level	Precision
0	0.05
1	0.70
2	0.95
Accuracy	**0.57**
Best Cohen's Kappa Score	**0.4928**
Best Threshold	**0.525**

Same in experiment 1, since the model was trained with 5 classes, the same method is applied in experiment 1 which is by merging the result of Mild with Moderate and Severe with Proliferative.

Table 21 and table 22 show the result from the testing set in experiment 3. The result technically has 0% of false negative which is quite good as the false negative needs to be reduced as much as it can since this is concerning the healthcare area. Plus the accuracy is a bit disappointing as well as the best Cohen's kappa score which are 0.57 and 0.4928 respectively. In addition to that, although the false negative is 0, the amount of false positive is way too high which 19 out of 20 normal labelled images are predicted as positive. It brings the result for only 0.05 precision for level 0. The rest precisions are excellent which are 0.70 for level 1 and 0.95 for level 2.

4.4 Experiment 4

Experiment 4 was done using undersampling with 15000 retinal images from the dataset containing 3 classes resulting in the distribution in Table 23.

Table 23 : Distribution for Experiment 4

Level	Total
0	1600
1	1599
2	1580
Total	**4779**

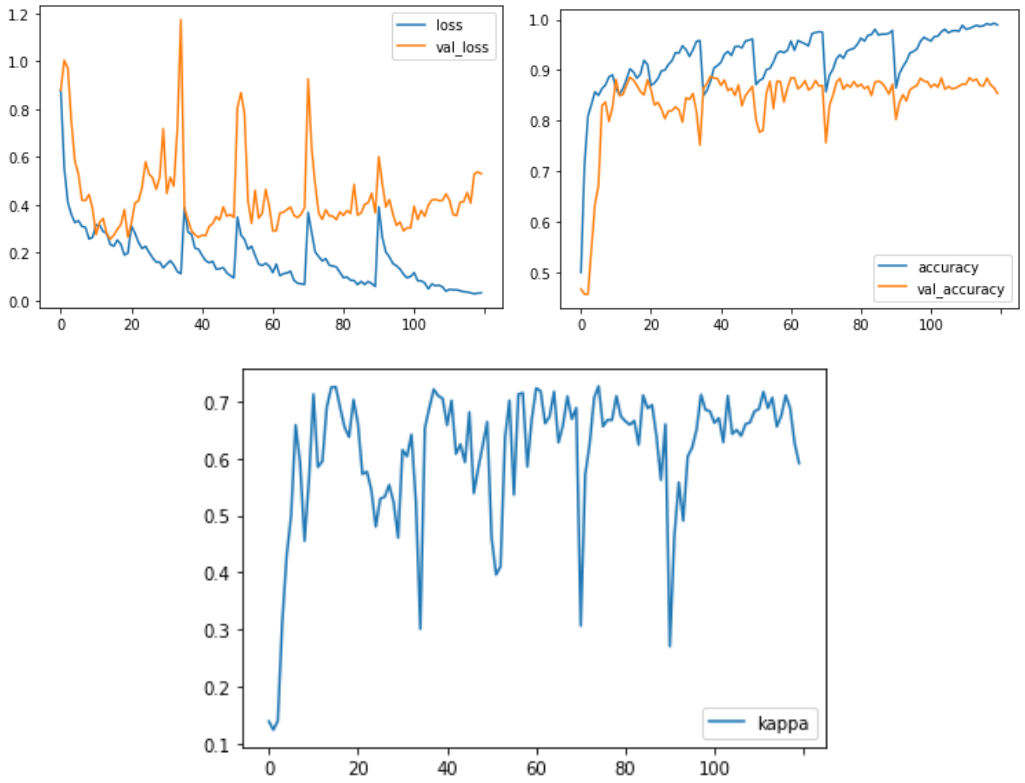


Figure 22 : Graphs of Multiple Binary Classification in Experiment 4

Figure 22 showed the result of Multiple Binary Classification for Experiment 4. The accuracy and val_accuray is increasing over the epochs and in contrary, loss and val _loss is decreasing with and high fluctuation entering the new bucket.

Table 24 : Confusion Matrix for Validation Set in Experiment 4

		Actual		
		0	1	2
Prediction	0	388	189	10
	1	96	205	77
	2	9	66	394

Table 25 : Precision for each level for Validation Set and the accuracy in Experiment 4

Level	Precision
0	0.79
1	0.45
2	0.82
Accuracy	**0.69**
Best Cohen's Kappa Score	**0.7515**
Best Threshold	**0.2375**

Based on Table 24, the number false negative is quite low in Moderate class, together with severe class makes a total of 199 out of 3199(class 1 and 2), which is 6.22% of the retinal images. Furthermore, in Table 25, it shows that level 2 has the highest precision which the model predicts level 2 better compared to other levels.

The accuracy is not so good but Cohen's kappa score is quite promising with 0.7515. The best threshold is very low 0.2375 which means the model needs to use a low threshold in order to get good results.

Table 26: Confusion Matrix for Testing Set in Experiment 4

		Actual		
		0	1	2
Prediction	0	7	1	0
	1	13	13	3
	2	0	6	17

Table 27: Precision for each level for Testing Set and the accuracy in Experiment 4

Level	Precision
0	0.35
1	0.65
2	0.85
Accuracy	**0.62**
Best Cohen's Kappa Score	**0.6761**
Best Threshold	**0.525**

Based on Table 26, only 1 out of 40(class 1 and 2) retinal images are false negative which is from the Moderate class. On Table 27, it can be concluded each level has a variety of precision scores in the range of 0.35 - 0.85. The model predicts well on level 2 but it predicts badly on level 0. The accuracy of the testing set is slightly lower than the validation set, and the best Cohen's kappa score is 0.6761

which is lower than the validation set. The best threshold for the testing set is a bit higher compared to the validation set.

4.5 Summary of Experiments

Table 28 : Summary of Result During Training Phase

Experiments	Multiple Binary Classification				Best Cohen's Kappa Score	Validation test accuracy	Best Threshold
	Accuracy	loss	Validation Accuracy	Validation loss			
1	0.9524	0.1313	0.9372	0.1614	0.7333	0.79	0.5271
2	0.9403	0.1542	0.9296	0.1805	0.6473	0.80	0.5359
3	0.9814	0.0675	0.8679	0.4613	0.7523	0.48	0.5125
4	0.9306	0.1658	0.8838	0.3392	0.7515	0.69	0.2375

Table 29 : Result of Testing Phase

Experiment	Best Cohen's Kappa Score	Best Threshold	Testing Set Accuracy	False Negative	False Positive
1	0.8205	0.5063	0.77	5	4
2	0.8205	0.5	0.82	4	3
3	0.4928	0.525	0.57	0	19
4	0.6761	0.525	0.62	1	13

4.6 Deployment

The best model was selected and deployed in a website application. This deployment contains a restful API that will predict the image when the image is selected.

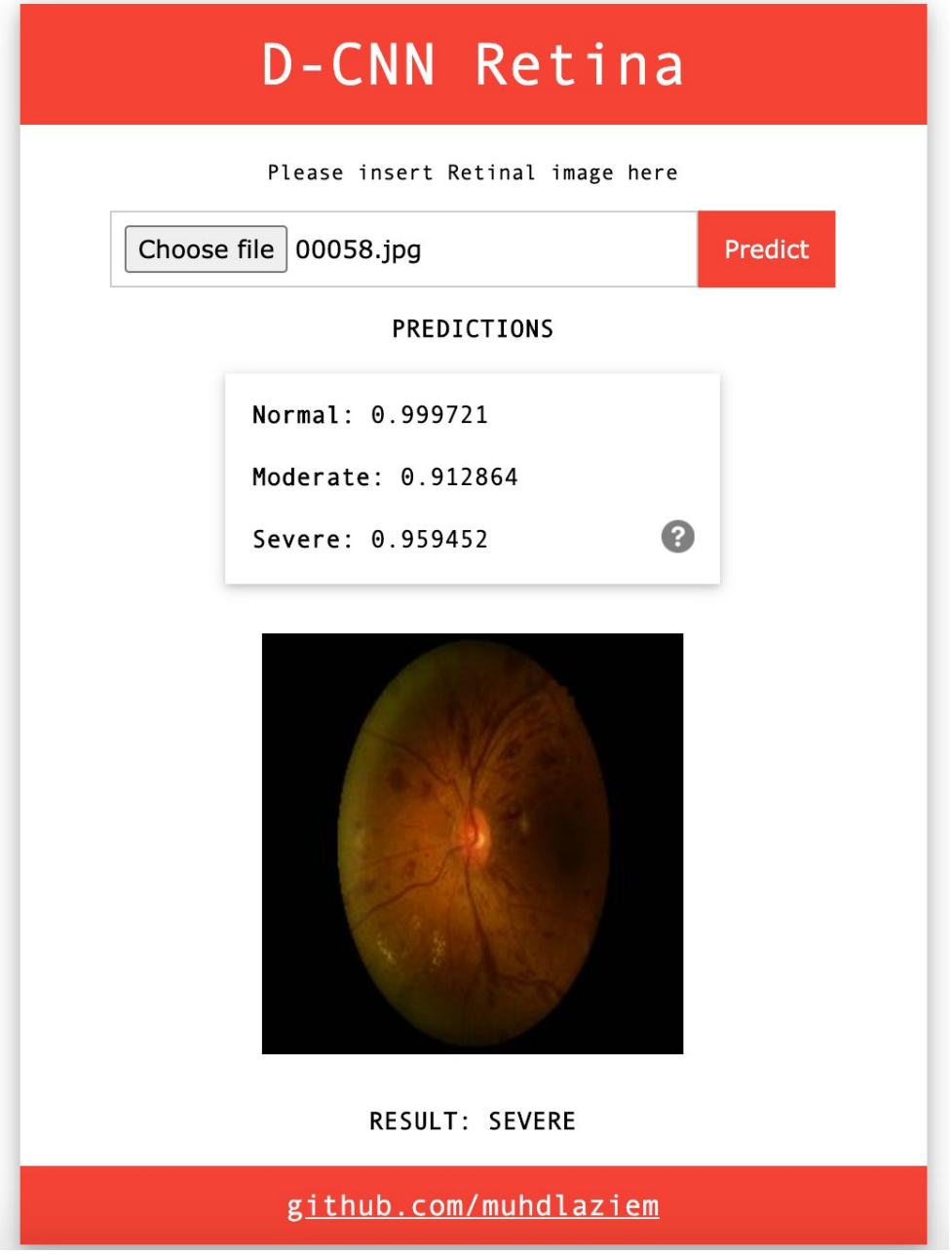


Figure 23 : The interface of Data Product

CHAPTER FIVE

CONCLUSION AND FUTURE WORKS

The existence of unbalanced dataset in the healthcare industry is real as the data are more towards negative classes no matter what diseases are. In this project, undersampling was applied as one of the ways to solve unbalanced dataset. However, the results are not as expected. The undersampling technique may reduce the amount of false negatives but the accuracies are a little disappointing as explained in 4.3 and 4.4. Nevertheless, this shows the ability of the DenseNet121 algorithm to solve unbalanced dataset.

In addition, it can be seen that 3-ary or 3 classes classification is better than 5-ary or 5 classes classification. In the testing set, experiment 2 is better than experiment 1 and experiment 4 is better than 3 which experiment 2 and 4 are both 3-ary classification. To conclude, the best result lies in experiment 2 applying the random sampling and 3-ary classification with result of 0.82 accuracy and 0.80-0.85 precision for all classes. Plus, Cohen's kappa score is also amongst the best which is 0.8205 with 0.5 threshold. This model is practically an improvement from all models in the experimental setup. Hence, the best model will be deployed to the web application.

For future works, it will be focussing on training a large scale of dataset rather than taking only 15000 retinal images. Plus, improving the deep learning model and better image preprocessing will be added in future.

REFERENCES

A.M. Martinez, & R. Benavente. (1998). The AR Face Database (No. 24).

Abdul Muthalib, A., Abdelsatar, A., Salameh, M., & Dalle, J. (2011). Making learning ubiquitous with Mobile Translator using optical character recognition (OCR). In 2011 International Conference on Advanced Computer Science and Information System (ICACSIS) (pp. 233–236).

Albus, J. S. (1975). A New Approach to Manipulator Control: The Cerebellar Model Articulation Controller (CMAC). Journal of Dynamic Systems, Measurement, and Control, 97(3), 220. doi:10.1115/1.3426922

Albus, J. S., Branch, D. T., Donald, C., & Perkel, H. (1971). A Theory of Cerebellar Function.

Alias, S., & Muhammad, S. K. (2013). Sequential pattern based multi document summarization #x2014; An exploratory approach. In 2013 International Conference on Research and Innovation in Information Systems (ICRIIS) (pp. 85–90). doi:10.1109/ICRIIS.2013.6716690

Godoy, D. (2019, February 07). Understanding binary cross-entropy / log loss: A visual explanation. Retrieved August 16, 2020, from https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).

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

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).

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

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

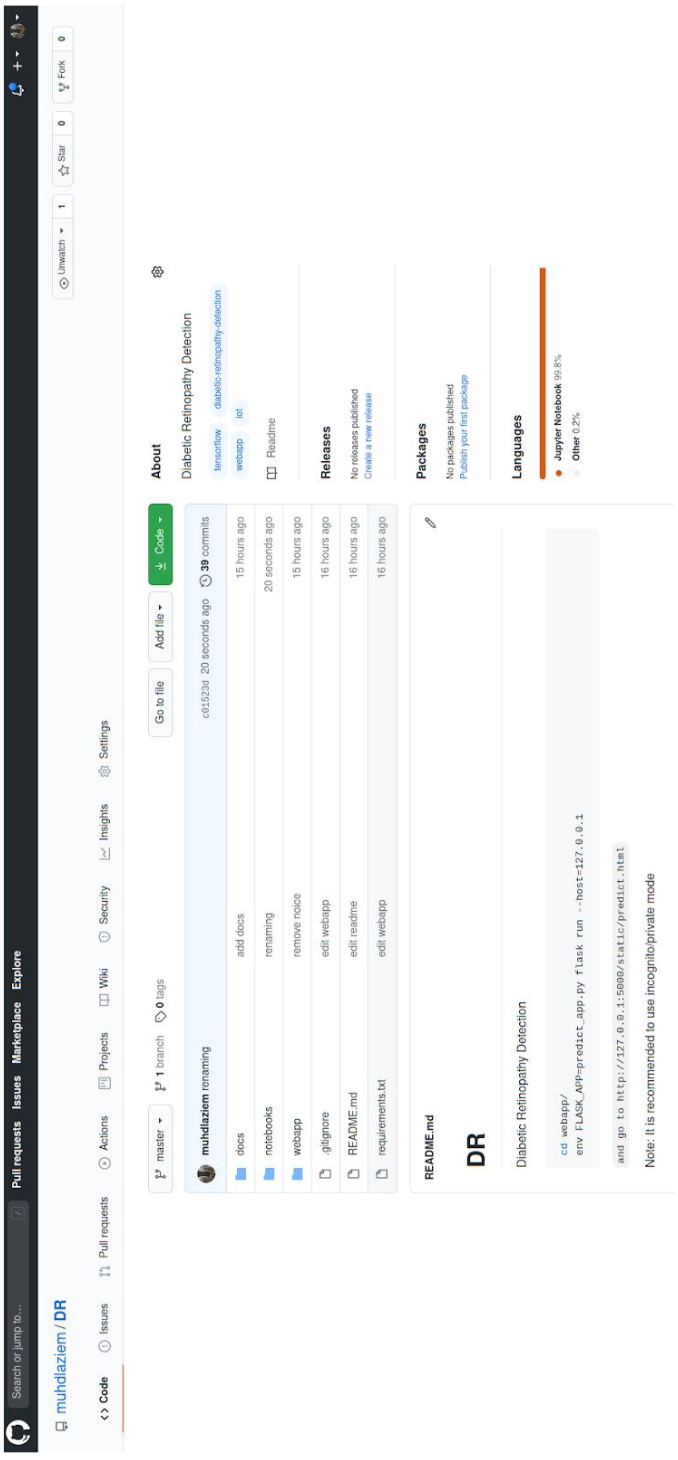
Pykes, K. (2020, February 27). Cohen's Kappa. Retrieved August 16, 2020, from https://towardsdatascience.com/cohens-kappa-9786ceceab58

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.

APPENDIX I

The entire project source code and documentation are available in github repository

: https://github.com/muhdlaziem/DR



APPENDIX II

Gantt Chart for this project.