DR FUNDUS

Link Google Colab

https://colab.research.google.com/drive/1PT5XfguV9ua62dOxwvrIehxgqy5Vh_a0?usp=sharing

Code

#install lb

```
!pip install tf-nightly
```

#import lb

```
import matplotlib.pyplot as plt
import numpy as np
import os
import PIL
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.models import Sequential
from IPython.display import Image
```

#เชื่อมกับข้อมูลกับ googledrive

```
import pathlib
path = "/content/drive/MyDrive/DR/Train"
data_dir = pathlib.Path(path)

from google.colab import drive
drive.mount('/content/drive')
```

#เรียกดูตัวอย่างภาพ

```
Mild  = list(data_dir.glob('Mild DR/*'))
PIL.Image.open(str(Mild [1]))
```

#ขนาดการอ่านข้อมูลแต่ละรอบ

```
batch_size = 32
img_height = 150
img_width = 150
```

#ทำการ Train 80% และ validation 20%

```
train_ds = tf.keras.preprocessing.image_dataset_from_directory(
data_dir,
validation_split=0.2,
subset="training",
seed=123,
image_size=(img_height, img_width),
batch_size=batch_size)

val_ds = tf.keras.preprocessing.image_dataset_from_directory(
data_dir,
validation_split=0.2,
subset="validation",
seed=123,
image_size=(img_height, img_width),
batch_size=batch_size)
```

#เช็คชื่อ Class ของ Dataset

```
class_names = train_ds.class_names
print(class_names)
```

#สุ่มตัวอย่างรูปจากใน dataset

```
plt.figure(figsize=(7, 7))
for images, labels in train_ds.take(1):
 for i in range(9):
   ax = plt.subplot(3, 3, i + 1)
   plt.imshow(images[i].numpy().astype("uint8"))
   plt.title(class_names[labels[i]])
   plt.axis("off")
```

#ทำการ normalization แปลงค่าสีให้สอดคล้องกับการทำ CNN

```
normalization_layer = layers.experimental.preprocessing.Rescaling(1./255)
normalized_ds = train_ds.map(lambda x, y: (normalization_layer(x), y))
image_batch, labels_batch = next(iter(normalized_ds))
first_image = image_batch[0]
print(np.min(first_image), np.max(first_image))

num_classes =9
model = Sequential([
layers.experimental.preprocessing.Rescaling(1./255, input_shape=(img_heigh
t, img_width, 3)),
layers.Conv2D(16, 3, padding='same', activation='relu'),
layers.MaxPooling2D(),
layers.Conv2D(32, 3, padding='same', activation='relu'),
layers.MaxPooling2D(),
layers.Conv2D(64, 3, padding='same', activation='relu'),
layers.MaxPooling2D(),
layers.Flatten(),
layers.Dense(128, activation='relu'),
layers.Dense(num_classes)])
```

#แสดงค่าความแม่นยำในระหว่างการ Train

```
model.compile(optimizer='adam',
loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),metri
cs=['accuracy'])
model.summary()
```

#Train model

```
epochs=10
history = model.fit(
  train_ds,
  validation_data=val_ds,
  epochs=epochs
)
```

#แสดงผลลัพท์ของการ Train เป็นรูปแบบกราฟ

```
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

loss = history.history['loss']
val_loss = history.history['val_loss']

epochs_range = range(epochs)
```

#กราฟ Training and Validation Accuracy

```
plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')
```

#กราฟ Training and Validation Loss

```
plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```

#การทดสอบ

```
path = "/content/drive/MyDrive/DR/Test/Severe DR/Severe DR10606_right_0_85
66.jpeg"
img = keras.preprocessing.image.load_img(
path, target_size=(img_height, img_width)
)

img_array = keras.preprocessing.image.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = model.predict(img_array)
score = tf.nn.softmax(predictions[0])
display(Image(filename=path))
print(
"จากการประมวลผลภาพจอประสาทตาภาพนี้ เป็น โรคเบาหวานขึ้นจอตาระยะ {} มีความตรงกันกับต้นแบบ {:.2f} % "
.format(class_names[np.argmax(score)], 100 * np.max(score))
)
```