AI/ML Mini Project (Exp 9 & 10)

Topic: Alzheimer's Disease Detection

Team Members:

1. Hatim Sawai (2021300108)

2. Tathagat Sengupta (2021300110)

3. Sahil Shah (2021300115)

In [ ]:
```
import warnings
warnings.filterwarnings('ignore')
```

Importing Libraries

In [ ]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import tensorflow as tf

from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix

from tensorflow import keras
from keras import Input
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten, Conv2D, BatchNormalization, MaxPo
from keras.preprocessing.image import ImageDataGenerator as IDG
```

Importing Dataset

In [ ]:
```
BASE_DIR = "./data/"
TRAIN_DIR = BASE_DIR + 'train'
TEST_DIR = BASE_DIR + 'test'

CLASSES = [ 'NonDemented',
            'VeryMildDemented',
            'MildDemented',
            'ModerateDemented']

IMG_SIZE = 176
IMAGE_SIZE = [176, 176]
DIM = (IMG_SIZE, IMG_SIZE)
```

In [ ]:
```
datagen = IDG(
    rescale = 1./255,
```

```
    brightness_range=[0.8, 1.2],
    zoom_range=[.99, 1.01],
    data_format="channels_last",
    fill_mode='constant',
    horizontal_flip=True
)

train_data_gen = datagen.flow_from_directory(directory=TRAIN_DIR, target_size=DIM,
```

```
Found 5121 images belonging to 4 classes.
```

In [ ]:
```
#Retrieving the data from the ImageDataGenerator iterator
train_data, train_labels = train_data_gen.next()
#Getting to know the dimensions of our dataset
print(train_data.shape, train_labels.shape)
```

```
(5121, 176, 176, 3) (5121, 4)
```

Splitting Dataset into Training and Testing Sets (80:20)

In [ ]:
```
#Splitting the data into train, test, and validation sets
train_data, test_data, train_labels, test_labels = train_test_split(train_data, tra
train_data, val_data, train_labels, val_labels = train_test_split(train_data, train
```

Model Building

In [ ]:
```
def conv_block(filters, act='relu'):
    """Defining a Convolutional NN block for a Sequential CNN model. """

    block = Sequential()
    block.add(Conv2D(filters, 3, activation=act, padding='same'))
    block.add(Conv2D(filters, 3, activation=act, padding='same'))
    block.add(BatchNormalization())
    block.add(MaxPool2D())

    return block

def dense_block(units, dropout_rate, act='relu'):
    """Defining a Dense NN block for a Sequential CNN model. """

    block = Sequential()
    block.add(Dense(units, activation=act))
    block.add(BatchNormalization())
    block.add(Dropout(dropout_rate))

    return block

def construct_model(act='relu'):
    """Constructing a Sequential CNN architecture for performing the classification

    model = Sequential([
        Input(shape=(*IMAGE_SIZE, 3)),
        Conv2D(16, 3, activation=act, padding='same'),
        Conv2D(16, 3, activation=act, padding='same'),
        MaxPool2D(),
```

```
        conv_block(32),
        conv_block(64),
        conv_block(128),
        Dropout(0.2),
        conv_block(256),
        Dropout(0.2),
        Flatten(),
        dense_block(512, 0.7),
        dense_block(128, 0.5),
        dense_block(64, 0.3),
        Dense(4, activation='softmax')
    ], name = "cnn_model")

    return model
```

In [ ]:
```
class MyCallback(tf.keras.callbacks.Callback):
    def on_epoch_end(self, epoch, logs={}):
        if logs.get('val_acc') > 0.99:
            print("\nReached accuracy threshold! Terminating training.")
            self.model.stop_training = True

my_callback = MyCallback()
CALLBACKS = [my_callback]
```

In [ ]:
```
model = construct_model()

METRICS = [tf.keras.metrics.CategoricalAccuracy(name='acc'),
           tf.keras.metrics.AUC(name='auc')]

model.compile(optimizer='adam',
              loss=tf.keras.losses.CategoricalCrossentropy(),
              metrics=METRICS)

model.summary()
```

```
WARNING:tensorflow:From d:\SEM_5\HealthCure\venv\Lib\site-packages\keras\src\backend.py:1398: The name tf.executing_eagerly_outside_functions is deprecated. Please use tf.compat.v1.executing_eagerly_outside_functions instead.

WARNING:tensorflow:From d:\SEM_5\HealthCure\venv\Lib\site-packages\keras\src\layers\pooling\max_pooling2d.py:161: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From d:\SEM_5\HealthCure\venv\Lib\site-packages\keras\src\optimizers\__init__.py:309: The name tf.train.Optimizer is deprecated. Please use tf.compat.v1.train.Optimizer instead.

Model: "cnn_model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 176, 176, 16)      448

 conv2d_1 (Conv2D)           (None, 176, 176, 16)      2320

 max_pooling2d (MaxPooling2  (None, 88, 88, 16)        0
 D)

 sequential (Sequential)     (None, 44, 44, 32)        14016

 sequential_1 (Sequential)   (None, 22, 22, 64)        55680

 sequential_2 (Sequential)   (None, 11, 11, 128)       221952

 dropout (Dropout)           (None, 11, 11, 128)       0

 sequential_3 (Sequential)   (None, 5, 5, 256)         886272

 dropout_1 (Dropout)         (None, 5, 5, 256)         0

 flatten (Flatten)           (None, 6400)              0

 sequential_4 (Sequential)   (None, 512)               3279360

 sequential_5 (Sequential)   (None, 128)               66176

 sequential_6 (Sequential)   (None, 64)                8512

 dense_3 (Dense)             (None, 4)                 260

=================================================================
Total params: 4534996 (17.30 MB)
Trainable params: 4532628 (17.29 MB)
Non-trainable params: 2368 (9.25 KB)
_________________________________________________________________
```

Model Evaluation

In [ ]:
```
#Fit the training data to the model and validate it using the validation data
EPOCHS = 50
```

```
history = model.fit(train_data, train_labels, validation_data=(val_data, val_labels
model.save('alzheimer_model.h5')
```

```
Epoch 1/50
WARNING:tensorflow:From d:\SEM_5\HealthCure\venv\Lib\site-packages\keras\src\utils\t
f_utils.py:492: The name tf.ragged.RaggedTensorValue is deprecated. Please use tf.co
mpat.v1.ragged.RaggedTensorValue instead.

103/103 [==============================] - 190s 2s/step - loss: 1.7342 - acc: 0.3223
- auc: 0.5700 - val_loss: 1.3618 - val_acc: 0.4841 - val_auc: 0.6245
Epoch 2/50
103/103 [==============================] - 155s 2s/step - loss: 1.3533 - acc: 0.4158
- auc: 0.6823 - val_loss: 1.6111 - val_acc: 0.3768 - val_auc: 0.7032
Epoch 3/50
103/103 [==============================] - 144s 1s/step - loss: 1.1672 - acc: 0.4850
- auc: 0.7576 - val_loss: 1.0724 - val_acc: 0.4939 - val_auc: 0.7958
Epoch 4/50
103/103 [==============================] - 140s 1s/step - loss: 1.0802 - acc: 0.5079
- auc: 0.7902 - val_loss: 1.8122 - val_acc: 0.5000 - val_auc: 0.7820
Epoch 5/50
103/103 [==============================] - 140s 1s/step - loss: 1.0261 - acc: 0.5140
- auc: 0.8050 - val_loss: 1.1497 - val_acc: 0.4049 - val_auc: 0.7064
Epoch 6/50
103/103 [==============================] - 137s 1s/step - loss: 0.9732 - acc: 0.5394
- auc: 0.8223 - val_loss: 1.6266 - val_acc: 0.4854 - val_auc: 0.8108
Epoch 7/50
103/103 [==============================] - 140s 1s/step - loss: 0.9417 - acc: 0.5467
- auc: 0.8303 - val_loss: 3.4227 - val_acc: 0.1329 - val_auc: 0.3489
Epoch 8/50
103/103 [==============================] - 160s 2s/step - loss: 0.9257 - acc: 0.5491
- auc: 0.8354 - val_loss: 1.5009 - val_acc: 0.1329 - val_auc: 0.5646
Epoch 9/50
103/103 [==============================] - 146s 1s/step - loss: 0.9045 - acc: 0.5464
- auc: 0.8405 - val_loss: 0.8759 - val_acc: 0.5573 - val_auc: 0.8469
Epoch 10/50
103/103 [==============================] - 141s 1s/step - loss: 0.9095 - acc: 0.5534
- auc: 0.8377 - val_loss: 1.0357 - val_acc: 0.5073 - val_auc: 0.8266
Epoch 11/50
103/103 [==============================] - 144s 1s/step - loss: 0.8939 - acc: 0.5620
- auc: 0.8439 - val_loss: 10.4977 - val_acc: 0.4841 - val_auc: 0.6577
Epoch 12/50
103/103 [==============================] - 143s 1s/step - loss: 0.8820 - acc: 0.5708
- auc: 0.8481 - val_loss: 0.9539 - val_acc: 0.4890 - val_auc: 0.8127
Epoch 13/50
103/103 [==============================] - 145s 1s/step - loss: 0.8737 - acc: 0.5678
- auc: 0.8513 - val_loss: 1.2950 - val_acc: 0.2085 - val_auc: 0.6209
Epoch 14/50
103/103 [==============================] - 161s 2s/step - loss: 0.8616 - acc: 0.5800
- auc: 0.8556 - val_loss: 1.5987 - val_acc: 0.1976 - val_auc: 0.5873
Epoch 15/50
103/103 [==============================] - 155s 2s/step - loss: 0.8305 - acc: 0.5971
- auc: 0.8666 - val_loss: 4.2012 - val_acc: 0.1329 - val_auc: 0.3048
Epoch 16/50
103/103 [==============================] - 115s 1s/step - loss: 0.8441 - acc: 0.5897
- auc: 0.8627 - val_loss: 1.2286 - val_acc: 0.3720 - val_auc: 0.7280
Epoch 17/50
103/103 [==============================] - 110s 1s/step - loss: 0.8397 - acc: 0.5919
- auc: 0.8645 - val_loss: 4.2338 - val_acc: 0.4841 - val_auc: 0.7053
Epoch 18/50
```

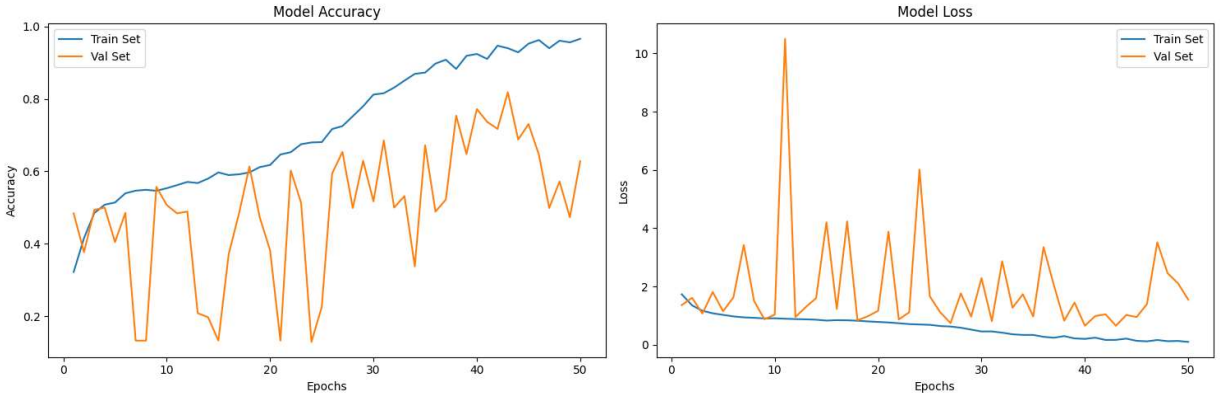
```
103/103 [==============================] - 105s 1s/step - loss: 0.8260 - acc: 0.5971
- auc: 0.8683 - val_loss: 0.8427 - val_acc: 0.6134 - val_auc: 0.8721
Epoch 19/50
103/103 [==============================] - 110s 1s/step - loss: 0.8035 - acc: 0.6117
- auc: 0.8770 - val_loss: 0.9784 - val_acc: 0.4732 - val_auc: 0.8015
Epoch 20/50
103/103 [==============================] - 112s 1s/step - loss: 0.7839 - acc: 0.6175
- auc: 0.8830 - val_loss: 1.1682 - val_acc: 0.3817 - val_auc: 0.7312
Epoch 21/50
103/103 [==============================] - 104s 1s/step - loss: 0.7657 - acc: 0.6465
- auc: 0.8908 - val_loss: 3.8818 - val_acc: 0.1329 - val_auc: 0.3022
Epoch 22/50
103/103 [==============================] - 106s 1s/step - loss: 0.7396 - acc: 0.6529
- auc: 0.8975 - val_loss: 0.8690 - val_acc: 0.6024 - val_auc: 0.8673
Epoch 23/50
103/103 [==============================] - 103s 996ms/step - loss: 0.7098 - acc: 0.6
749 - auc: 0.9060 - val_loss: 1.1138 - val_acc: 0.5134 - val_auc: 0.8431
Epoch 24/50
103/103 [==============================] - 102s 990ms/step - loss: 0.6974 - acc: 0.6
798 - auc: 0.9096 - val_loss: 6.0118 - val_acc: 0.1293 - val_auc: 0.2597
Epoch 25/50
103/103 [==============================] - 101s 986ms/step - loss: 0.6850 - acc: 0.6
807 - auc: 0.9125 - val_loss: 1.6620 - val_acc: 0.2256 - val_auc: 0.6314
Epoch 26/50
103/103 [==============================] - 101s 983ms/step - loss: 0.6432 - acc: 0.7
167 - auc: 0.9239 - val_loss: 1.1192 - val_acc: 0.5939 - val_auc: 0.8642
Epoch 27/50
103/103 [==============================] - 100s 971ms/step - loss: 0.6274 - acc: 0.7
250 - auc: 0.9281 - val_loss: 0.7429 - val_acc: 0.6537 - val_auc: 0.8950
Epoch 28/50
103/103 [==============================] - 103s 998ms/step - loss: 0.5849 - acc: 0.7
521 - auc: 0.9377 - val_loss: 1.7615 - val_acc: 0.4988 - val_auc: 0.8230
Epoch 29/50
103/103 [==============================] - 102s 992ms/step - loss: 0.5204 - acc: 0.7
796 - auc: 0.9506 - val_loss: 0.9678 - val_acc: 0.6293 - val_auc: 0.8838
Epoch 30/50
103/103 [==============================] - 100s 965ms/step - loss: 0.4571 - acc: 0.8
120 - auc: 0.9619 - val_loss: 2.2887 - val_acc: 0.5171 - val_auc: 0.7925
Epoch 31/50
103/103 [==============================] - 99s 959ms/step - loss: 0.4591 - acc: 0.81
56 - auc: 0.9617 - val_loss: 0.8103 - val_acc: 0.6854 - val_auc: 0.9100
Epoch 32/50
103/103 [==============================] - 99s 965ms/step - loss: 0.4158 - acc: 0.83
12 - auc: 0.9685 - val_loss: 2.8675 - val_acc: 0.5000 - val_auc: 0.7631
Epoch 33/50
103/103 [==============================] - 98s 952ms/step - loss: 0.3597 - acc: 0.85
04 - auc: 0.9764 - val_loss: 1.2714 - val_acc: 0.5317 - val_auc: 0.8221
Epoch 34/50
103/103 [==============================] - 101s 981ms/step - loss: 0.3383 - acc: 0.8
690 - auc: 0.9786 - val_loss: 1.7347 - val_acc: 0.3378 - val_auc: 0.6890
Epoch 35/50
103/103 [==============================] - 101s 979ms/step - loss: 0.3383 - acc: 0.8
727 - auc: 0.9781 - val_loss: 0.9707 - val_acc: 0.6720 - val_auc: 0.8964
Epoch 36/50
103/103 [==============================] - 110s 1s/step - loss: 0.2731 - acc: 0.8974
- auc: 0.9861 - val_loss: 3.3510 - val_acc: 0.4890 - val_auc: 0.7240
```

```
Epoch 37/50
103/103 [==============================] - 106s 1s/step - loss: 0.2418 - acc: 0.9081
- auc: 0.9886 - val_loss: 2.0417 - val_acc: 0.5220 - val_auc: 0.8255
Epoch 38/50
103/103 [==============================] - 104s 1s/step - loss: 0.2984 - acc: 0.8828
- auc: 0.9832 - val_loss: 0.8282 - val_acc: 0.7537 - val_auc: 0.9256
Epoch 39/50
103/103 [==============================] - 108s 1s/step - loss: 0.2191 - acc: 0.9188
- auc: 0.9907 - val_loss: 1.4488 - val_acc: 0.6476 - val_auc: 0.8636
Epoch 40/50
103/103 [==============================] - 124s 1s/step - loss: 0.2027 - acc: 0.9240
- auc: 0.9916 - val_loss: 0.6542 - val_acc: 0.7720 - val_auc: 0.9380
Epoch 41/50
103/103 [==============================] - 118s 1s/step - loss: 0.2441 - acc: 0.9103
- auc: 0.9880 - val_loss: 0.9865 - val_acc: 0.7366 - val_auc: 0.9112
Epoch 42/50
103/103 [==============================] - 109s 1s/step - loss: 0.1660 - acc: 0.9469
- auc: 0.9939 - val_loss: 1.0448 - val_acc: 0.7171 - val_auc: 0.9083
Epoch 43/50
103/103 [==============================] - 108s 1s/step - loss: 0.1684 - acc: 0.9399
- auc: 0.9931 - val_loss: 0.6538 - val_acc: 0.8183 - val_auc: 0.9480
Epoch 44/50
103/103 [==============================] - 113s 1s/step - loss: 0.2120 - acc: 0.9286
- auc: 0.9906 - val_loss: 1.0232 - val_acc: 0.6878 - val_auc: 0.8931
Epoch 45/50
103/103 [==============================] - 114s 1s/step - loss: 0.1366 - acc: 0.9524
- auc: 0.9963 - val_loss: 0.9505 - val_acc: 0.7305 - val_auc: 0.9142
Epoch 46/50
103/103 [==============================] - 113s 1s/step - loss: 0.1185 - acc: 0.9625
- auc: 0.9967 - val_loss: 1.3944 - val_acc: 0.6463 - val_auc: 0.8717
Epoch 47/50
103/103 [==============================] - 103s 999ms/step - loss: 0.1648 - acc: 0.9
399 - auc: 0.9944 - val_loss: 3.5149 - val_acc: 0.4988 - val_auc: 0.7186
Epoch 48/50
103/103 [==============================] - 103s 1s/step - loss: 0.1236 - acc: 0.9606
- auc: 0.9958 - val_loss: 2.4566 - val_acc: 0.5720 - val_auc: 0.7948
Epoch 49/50
103/103 [==============================] - 107s 1s/step - loss: 0.1319 - acc: 0.9560
- auc: 0.9952 - val_loss: 2.1127 - val_acc: 0.4732 - val_auc: 0.7555
Epoch 50/50
103/103 [==============================] - 109s 1s/step - loss: 0.0986 - acc: 0.9658
- auc: 0.9975 - val_loss: 1.5492 - val_acc: 0.6280 - val_auc: 0.8657
```

In [ ]:

```
# plot model performance
acc = history.history['acc']
val_acc = history.history['val_acc']
loss = history.history['loss']
val_loss = history.history['val_loss']
epochs_range = range(1, len(history.epoch) + 1)

plt.figure(figsize=(15,5))

plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Train Set')
plt.plot(epochs_range, val_acc, label='Val Set')
plt.legend(loc="best")
```

```
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.title('Model Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Train Set')
plt.plot(epochs_range, val_loss, label='Val Set')
plt.legend(loc="best")
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.title('Model Loss')

plt.tight_layout()
plt.show()
```



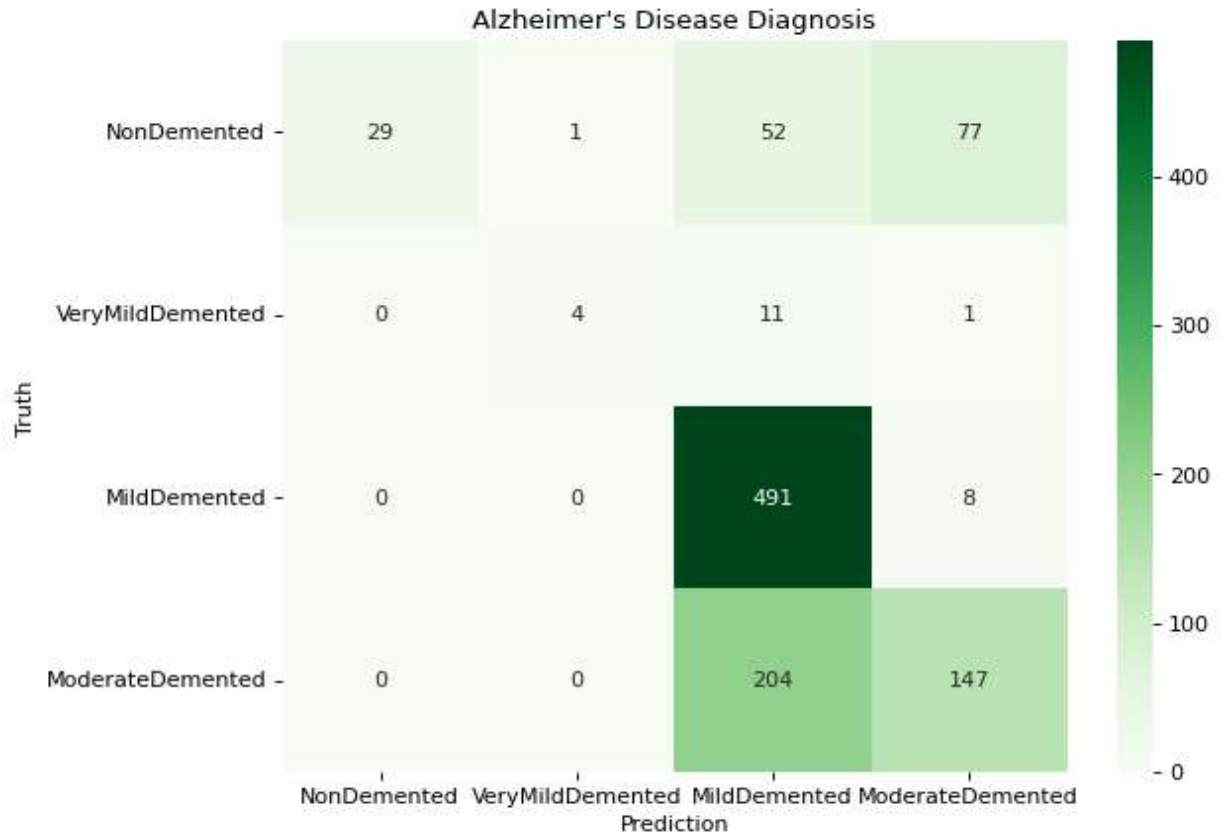
Model Testing

In [ ]:
```
#Predicting the test data

pred_labels = model.predict(test_data)
```

```
33/33 [==============================] - 12s 224ms/step
```

In [ ]:
```
#Plot the confusion matrix to understand the classification in detail
pred_ls = np.argmax(pred_labels, axis=1)
test_ls = np.argmax(test_labels, axis=1)

conf_arr = confusion_matrix(test_ls, pred_ls)
print(conf_arr)

plt.figure(figsize=(8, 6), dpi=80, facecolor='w', edgecolor='k')
ax = sns.heatmap(conf_arr, cmap='Greens', annot=True, fmt='d', xticklabels=CLASSES,
plt.title('Alzheimer\'s Disease Diagnosis')
plt.xlabel('Prediction')
plt.ylabel('Truth')
plt.show(ax)
```

```
[[ 29   1  52  77]
 [  0   4  11   1]
 [  0   0 491   8]
 [  0   0 204 147]]
```



Model Demo

In [ ]:
```
# test the model with an image stored in data_preview
from keras.models import load_model
from keras.preprocessing import image
import numpy as np

model = load_model('alzheimer_model.h5')
img = image.load_img('./data_preview/alzheimer/VeryMildDemented/verymildDem1.jpg',
img = image.img_to_array(img)
img = np.expand_dims(img, axis=0)
img = img/255

pred = model.predict(img)
index = pred[0].argmax()
outputs = ['Mild Demented', 'Moderate Demented', 'Non Demented', 'Very Mild Demente
print("Prediction: ", outputs[index])
```

```
1/1 [==============================] - 0s 395ms/step
Prediction:  Very Mild Demented
1/1 [==============================] - 0s 395ms/step
Prediction:  Very Mild Demented
```