VERZEO-MINOR PROJECT

Submitted By: Aleena Mathew

Submitted On: 21 -02-2022

E-mail: aleenamathew177@gmail.com

Project Name: Artificial Intelligence December Minor Project

Project Description: The goal of this project is to create a model that will be able to recognize and determine the handwritten digits from its image by using the concepts of Artificial Neural Network.

• Perform a digit classification to correctly identify digits from a dataset of tens of thousands of handwritten images from the MNIST dataset from keras.

• MNIST database of handwritten digits is used as dataset. It consists of a training set of 60,000 examples, and a test set of 10,000 examples. The digits have been size-normalized and centered in a fixed-size image of 28*28 pixels (784 pixels).

• also create a .hdf5 model Share your project as a .ipynb file (colab notebook)

Code:

```
import tensorflow as tf

mnist = tf.keras.datasets.mnist
(x_train, y_train),(x_test, y_test) = mnist.load_data() # used to load the dataset from keras.

x_train = tf.keras.utils.normalize(x_train, axis=1) # scaling data between 0 and 1
x_test = tf.keras.utils.normalize(x_test, axis=1)  # scaling data between 0 and 1

model = tf.keras.models.Sequential()
model.add(tf.keras.layers.Flatten())  # this takes 28x28 pixels and make it 1x784 pixels
model.add(tf.keras.layers.Dense(128, activation=tf.nn.relu))
model.add(tf.keras.layers.Dense(128, activation=tf.nn.relu))
model.add(tf.keras.layers.Dense(10, activation=tf.nn.softmax)) # output
 layer and softmax is used for probability distribution

model.compile(optimizer='adam',loss='sparse_categorical_crossentropy',
metrics=['accuracy'])

model.fit(x_train, y_train) # train the model

val_loss, val_acc = model.evaluate(x_test, y_test) # evaluate the sample data with model
print(val_loss) # error in model
print(val_acc) #  accuracy of the model
model.save('num_model1.hdf5')

new_model = tf.keras.models.load_model('num_model1.hdf5')

predictions = new_model.predict(x_test)
print(predictions)
```

```
import numpy as np
import matplotlib.pyplot as plt

print(np.argmax(predictions[11]))
plt.imshow(x_test[11],cmap=plt.cm.binary)
plt.show()
```

Output:

```
6
```