AI and Deep Learning

End Course Test

edureka!

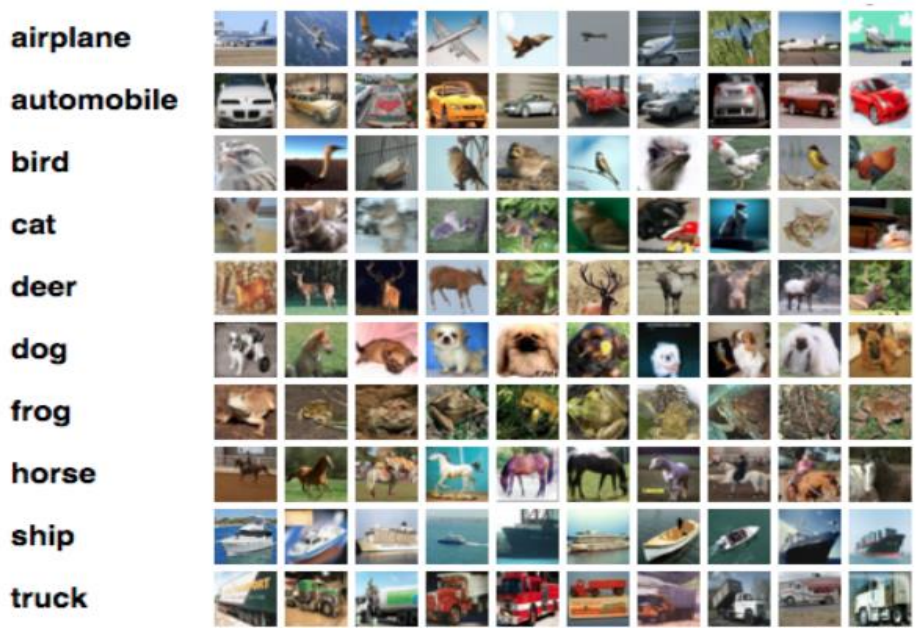
edureka!

© Brain4ce Education Solutions Pvt. Ltd.

End Course Test1

Description:

Classify the images using SLP, MLP and CNN techniques for the given CIFAR-10 data set.



Data Set:

Total Images: 60,000

Train Images: 50,000

Test Images: 10,000

Images/Class: 6,000

Image Size: 32 X 32

Different Classes:

Classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck.

Note: Please use Google Colaboratory – a free service by Google for AI developers – to work on this project. Also, make sure to opt for GPU under 'Hardware accelerator' while selecting Python 3 under 'Runtime type.'

https://medium.com/deep-learning-turkey/google-colab-free-gpu-tutorial-e113627b9f5d

Tasks to be Performed:

Question 1:

A. Load CIFAR-10 data from Keras Library and split it using the Train- and-Test method (90, 10). Allot the 90% data to Training and the rest 10% to Testing.

Output:

```
Downloading data from https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz
170500096/170498071 [==============================] - 11s 0us/step
```

B. Reshape and normalize the data:

Hint:

- Reshape the input image as demonstrated below:
  - train: 50000,32 * 32 * 3
  - test: 10000, 32 * 32*3
- Change the type of data to **float32**
- Normalize the data by dividing each with 255

Question 2:

Convert the **y_train** and **y_test** to **categorical** by using the function **tf.Keras.utilis_to_categorical**

QUESTION 3:

Create a single layer perceptron with

Hint:

- Single dense 2056 neurons
  - input shape being 3072
- Activation function ReLU.
- Softmax layer with 10 neurons as output
- loss- categorical_crossentropy
- Optimizer - Adam

- batch_size = 1000; epochs=50

Print the accuracy and loss of SLP model.

.

Question 4:

Create a Multi-Layer Perceptron with

Hint:

- Single dense layer 2056 neurons
- Input shape being 3072
- Activation function ReLU
- Second Dense Layer 1024 neurons and activation ReLU
- Third Dense Layer 512 neurons and activation ReLU
- Fourth Dense Layer 512 neurons and activation ReLU
- Softmax layer with 10 neurons as output
- loss- categorical_crossentropy
- Optimizer - Adam
- batch_size = 1000; epochs=50

Print the accuracy and loss of MLP model

Hint:

```
model_mlp = Sequential()

model_mlp.add(Dense(2056, input_shape=(3072, )))
```

Question 5:

Create a CNN Model using the hint mentioned below.

Hint:

- Load CIFAR10 data
- Convert Train and Test type to **float32**
- Normalize the data by dividing each by 255
- Convert **Y** to **categorical** encoding by using **to_categorical**

CNN Layer:

- Layer 1
  - two Conv2d Layer with 32 filter of (3,3) with input_shape= x_train.shape[1:]
  - maxpool 2d with 2,2 filter
  - activation ReLU
  - dropout 0.25

- Layer 2

  -two Conv2d Layer with 64 filter of (3,3)
  - maxpool 2d with 2,2 filter
  - activation ReLU
  - dropout 0.25

- Layer 3

  - Flatten Layer
  - Dense Layer with 512 Neuron with Relu
  - Dropout 0.25
  - Dense Layer with 10 Neuron with softmaz Activation

- Compile the model with RmsProp optimizer and categorical_crossentropy loss
- Fit model with batch_size=5000, epochs=20
- Print the accuracy and loss

Hint:

```
#layer 2
model.add(Conv2D(64, (3, 3), padding='same'))
model.add(Activation('relu'))
model.add(Conv2D(64, (3, 3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
```