USN NUMBER:1RVU22CSE004
NAME: Abrar Hussain

Ex No: 4 **Date: 28-08-2024**	**Handwritten Digit Classifier**

Objective:

The objective is to develop a Convolutional Neural Network (CNN) for the classification of handwritten digits using the MNIST dataset. The model will involve multiple convolutional layers followed by fully connected layers, utilizing ReLU as the activation function. The project will also include calculating categorical cross-entropy loss and optimizing the model using the Adadelta optimizer.

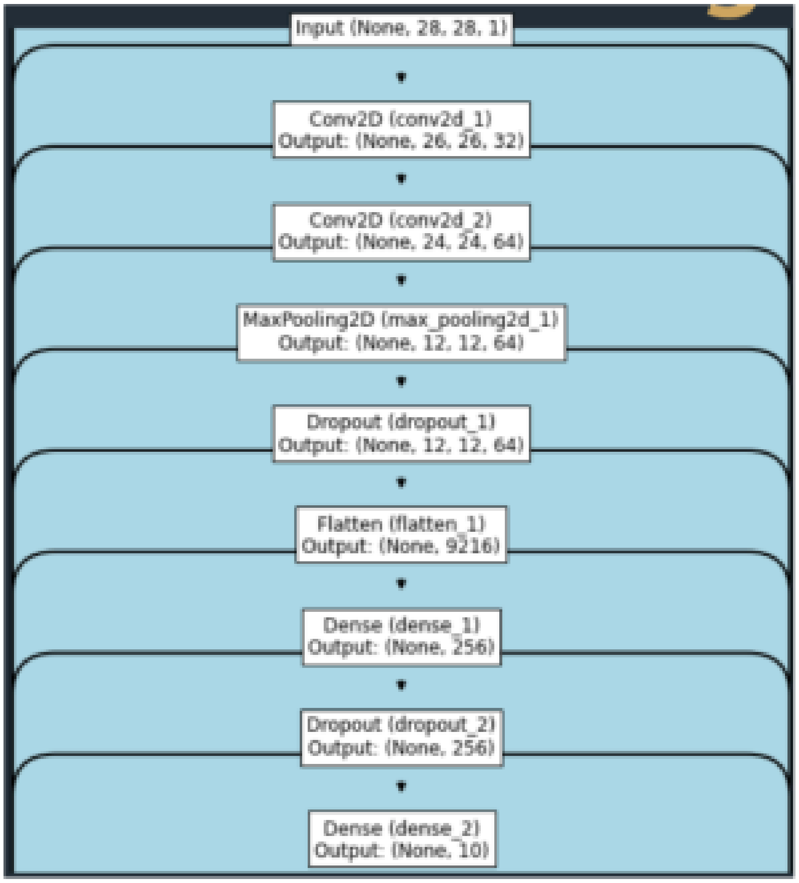
Descriptions:

The project focuses on building a CNN for the MNIST dataset, which consists of 28x28 grayscale images of handwritten digits from 0 to 9. The model will classify these images into one of ten classes (digits 0–9). CNNs are a powerful type of neural network that excel at capturing spatial hierarchies in image data, making them ideal for image classification tasks.

The network architecture involves two convolutional layers, each followed by a ReLU activation function to introduce non-linearity and a MaxPooling layer to down-sample the spatial dimensions. The final output from the convolutional layers is then flattened and passed through a fully connected (dense) layer, followed by a dropout layer to prevent overfitting. The last layer is a softmax layer, which outputs the probability distribution over the ten classes.

The model is trained using categorical cross-entropy loss, which measures the difference between the predicted class probabilities and the true labels. The Adadelta optimizer is used to update the model's weights, as it adapts the learning rate based on a moving window of gradient updates, enhancing training efficiency.

USN NUMBER:1RVU22CSE096
NAME:Maanya Manjunath



Steps to Build the Model:

Load and Preprocess Data:

Load the MNIST dataset, and reshape the images to include a channel dimension.
Normalize the pixel values to the range [0, 1] and one-hot encode the labels.
Define Model Architecture:

Input Layer: Shape (28, 28, 1) corresponding to the input images.
First Convolutional Layer: Apply 32 filters of size (3, 3) with ReLU activation. **Second Convolutional Layer:** Apply 64 filters of size (3, 3) with ReLU activation. **MaxPooling Layer:** Apply pooling with a (2, 2) filter size to down-sample the feature maps.
Dropout Layer: Apply dropout with a rate of 0.25 to prevent overfitting. **Flatten Layer:** Flatten the output from the convolutional layers to a 1D vector. **Fully Connected (Dense) Layer:** Apply 256 units with ReLU activation. **Second Dropout Layer:** Apply dropout with a rate of 0.5 to further prevent overfitting. **Output Layer:** Apply 10 units with softmax activation to output the probability distribution over 10 classes.

USN NUMBER:1RVU22CSE096

NAME:Maanya Manjunath

Compile the Model:

Use categorical_crossentropy as the loss function, which is suitable for multi-class classification.
Use the Adadelta optimizer to adaptively adjust the learning rate during training.
Evaluate the model's performance using the accuracy metric.
Train the Model:

Fit the model on the training data for a defined number of epochs, using a batch size of 128.
Validate the model on the test data after each epoch.
Evaluate the Model:

Evaluate the model's performance on the test set to obtain the final test loss and accuracy.
Summarize the model's architecture, including the number of layers, output shapes, and total parameters.
Save the Model:

Save the trained model as mnist.h5 for future use.

GitHub Link:
https://github.com/abraaaar/RVU_BtechHons/tree/main/Deep%20Learning/Lab4