Department Of Aerospace Engineering
Indian Institute Of Technology Madras

CS5830: Big Data Laboratory
Prof. Sudarsun

Lab Assignment 6:
Build a FastAPI for MNIST digit prediction

Sanket Pramod Bhure (*AE20B108*)

April 25, 2024

Contents

1 Introduction

The project focuses on developing a FastAPI-based REST API for digit classification using the MNIST dataset. The API allows users to upload images of handwritten digits and receive predictions about the digit represented in the image. It utilizes a pre-trained model to make accurate predictions, and users can interact with the API through Swagger UI. The assignment is available in the repository Sanky18/MNIST-Digit-Classification-API-with-FastAPI on GitHub.

2 Best Model

The best model used in this project for digit classification is trained on the MNIST dataset, which consists of grayscale images of handwritten digits (0-9). Here are the details of the best model:

- **Architecture**: The model architecture comprises a sequential neural network with multiple dense layers.
- **Input Layer**: The input layer consists of 256 nodes, which accept the flattened **28x28** pixel values of the input image.
- **Hidden Layer**: There is one hidden layer with **128** nodes, which helps capture complex patterns in the input data.
- **Output Layer**: The output layer consists of nodes corresponding to the 10 digit classes (0-9). It uses the softmax activation function to produce probability distributions over these classes.
- **Activation Functions**: Sigmoid activation functions are used in the hidden layers to introduce non-linearity, while the output layer uses softmax to produce probability scores for each class.
- **Optimizer**: The model is optimized using the **Adam optimizer**, which adapts the learning rate during training based on the gradients of the loss function.
- **Loss Function**: **Categorical cross-entropy** loss function is used to compute the loss between the predicted probabilities and the true labels.
- **Training Configuration**: The model is trained using a batch size of [32], and [10] epochs. The training data is shuffled and split into training and validation sets to monitor model performance during training.
- **Evaluation Metric**: The **accuracy metric** is used to evaluate the model's performance on both the training and validation sets.

This model configuration has been determined to be the best based on its performance on the test data from previous experiments. It has high accuracy with **0.97** in digit classification task and serves as the prediction model for the digit classification API implemented in this assignment.

3 Procedure

- **Model Loading**: Load a pre-trained MNIST digit classification model from a specified path on the disk.
- **Digit Prediction**: Predict the digit in an uploaded image using the loaded model.
- **Image Formatting**: Resize uploaded images to 28x28 grayscale and create a serialized array of 784 elements for prediction.
- **API Endpoint**: Expose a POST endpoint at `/predict` to accept image uploads, preprocess the data, and return the predicted digit.
- **Swagger UI Integration**: Accessible via `<api endpoint>/docs`, allowing users to test the API using Swagger UI.
- **Performance Evaluation**: Test the API with hand-drawn digit images and record observations.

4 Experiments

We have conducted 10 experiments, each corresponding to a hand-drawn digit created in MS Paint. These experiments were evaluated by uploading the images to the FastAPI.

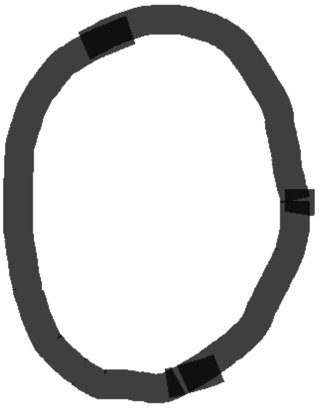

Figure 1: Experiment-1: Ground truth = 0 and Predicted = 0

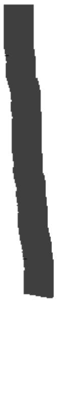

Figure 2: Experiment-2: Ground truth = 1 and Predicted = 5

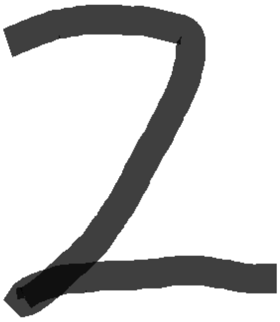

Figure 3: Experiment-3: Ground truth = 2 and Predicted = 2

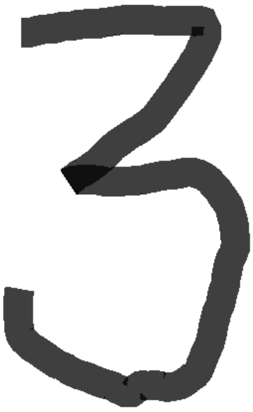

Figure 4: Experiment-4: Ground truth = 3 and Predicted = 3

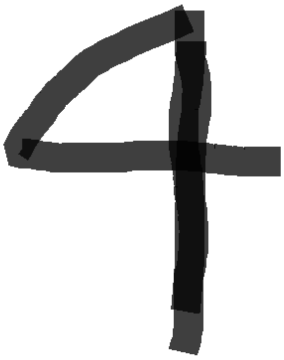

Figure 5: Experiment-5: Ground truth = 4 and Predicted = 9

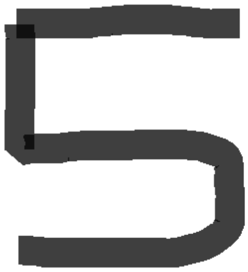

Figure 6: Experiment-6: Ground truth = 5 and Predicted = 6

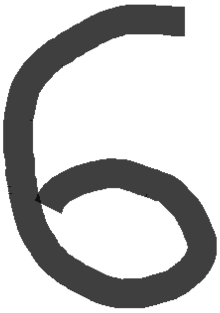

Figure 7: Experiment-7: Ground truth = 6 and Predicted = 6

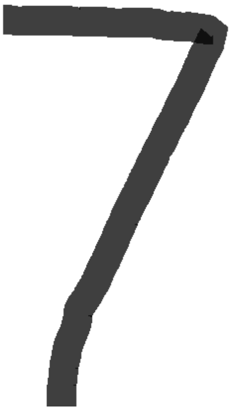

Figure 8: Experiment-8: Ground truth = 7 and Predicted = 7

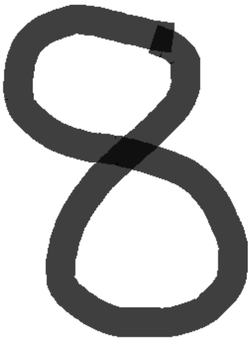

Figure 9: Experiment-9: Ground truth = 8 and Predicted = 8

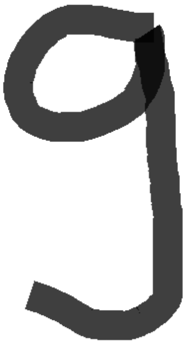

Figure 10: Experiment-10: Ground truth = 9 and Predicted = 2

Observations:

- **Dataset Discrepancy:** Our model, trained on the MNIST dataset, primarily consists of images with a black background. When tested with hand-drawn images from MS Paint, the model exhibited some misclassifications.
- **Potential Solution:** Augmenting the MNIST dataset with these hand-drawn images could enhance the model's ability to generalize to diverse data types.
- **Misclassifications:** Misclassifications include:
  - Predicting 0 as 5
  - Misidentifying 4 as 9
  - Confusing 5 with 6
  - Erroneously labeling 9 as 2
- **Insight:** These misclassifications underscore the importance of training models on diverse datasets that encompass various data distributions to improve robustness and generalization capabilities.