FAST NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES.

DEPARTMENT OF COMPUTER SCIENCE(CS)

SEMESTER SPRING 2023

BATCH 2020

Artificial Intelligence

PROJECT TOPIC:

MNIST Digit Classification with Neural Network

GROUP MEMBERS:

- Ayesha Zia (20k-0414)
- Robaisha Masood (20k-0390)
- Syeda Ravia Ejaz (20k-0246)

Project Motivation:

Our MNIST Digit Classification project aims to develop a neural network model capable of accurately identifying handwritten digits in the MNIST dataset. The dataset consists of 60,000 training images and 10,000 test images of handwritten digits from 0 to 9.

The project will follow a supervised learning approach, where the neural network will be trained using the labeled training data. The neural network architecture will consist of multiple layers, including an input layer, one or more hidden layers, and an output layer. The model will use backpropagation to optimize the weights and biases of the neurons in the network.

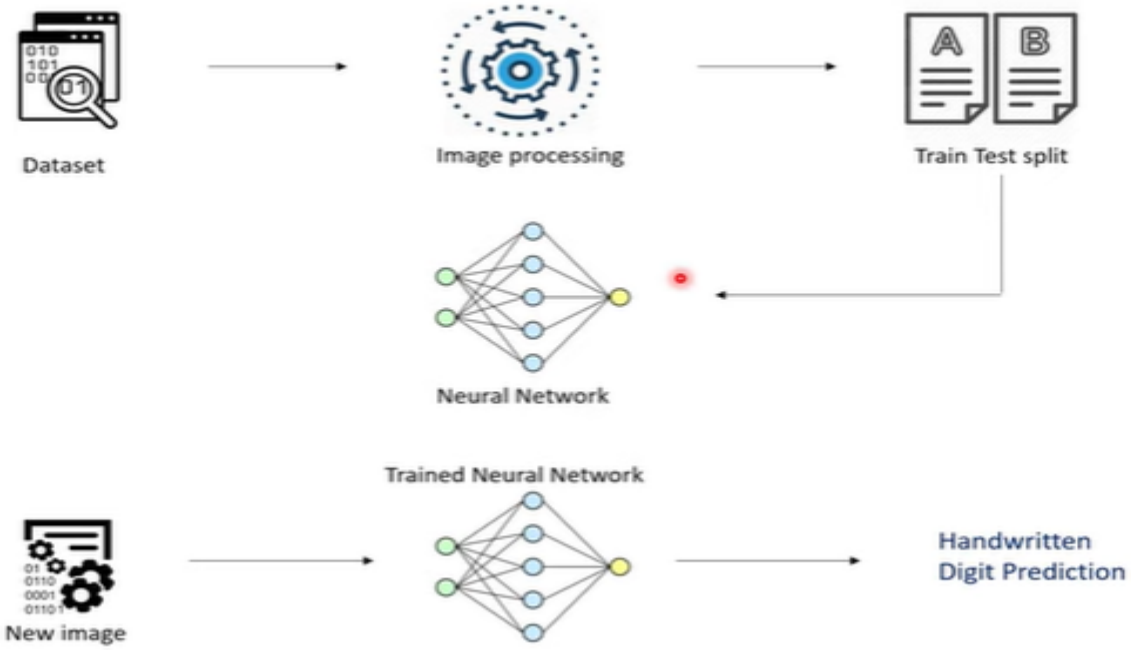
Various neural network architectures will be explored, including fully connected neural networks, convolutional neural networks (CNNs), and recurrent neural networks (RNNs), to determine the optimal architecture for the problem at hand. The model's performance will be evaluated using metrics such as accuracy, precision, recall, and F1 score.

Project Definition:

The project will be implemented using Python and the Keras library. The Keras library provides a user-friendly interface to create and train neural networks efficiently. The project will also make use of visualization tools such as Matplotlib and TensorBoard to visualize the neural network's performance. The project consists of a high-performing neural network model capable of accurately classifying handwritten digits. This model can be used in various applications, such as OCR systems and automated postal sorting.

The main objective of building this classifier is to train our neural network in such a way that it will be able to detect the input image. Our goal is to use these input images to develop AI-based approaches to predict and detect what particular digit is represented in the input image. Our input image will consist of digits from 0 to nine and our trained neural network will have to detect which digit is represented in the image. Our neural network is a single-digit detection neural network. If we are feeding a handwritten digit in this neural network, this trained neural network will successfully tell us what that digit is. Hence, once we train our neural network to detect the digit correctly, it is then tested with some test data. This whole process of training and testing will help us in building a single-digit detection classifier.

Model of Workflow:



Method followed:

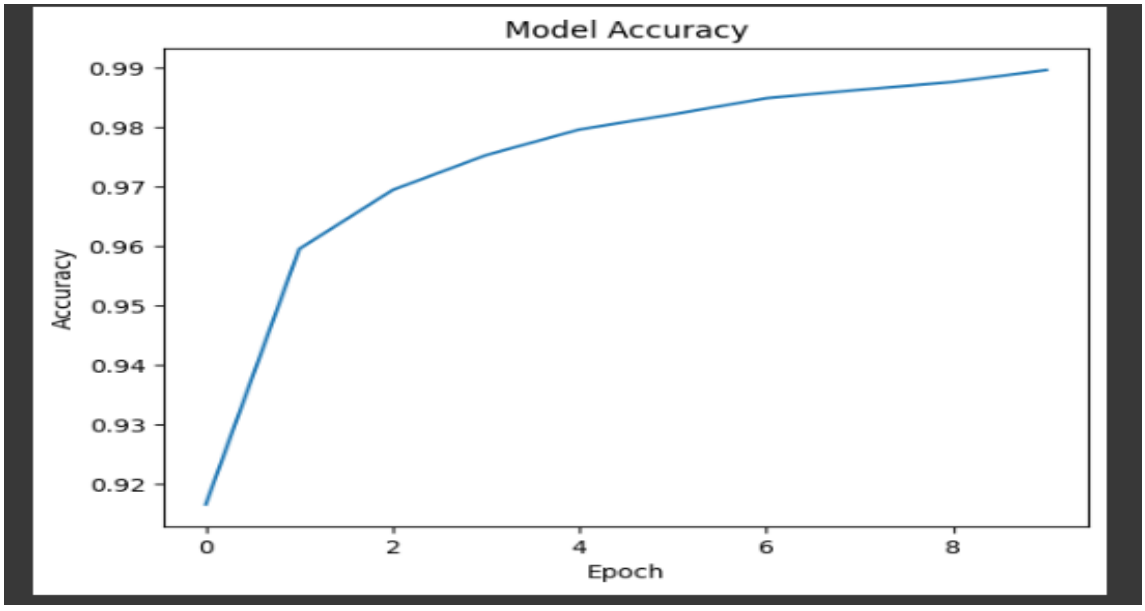
1. Data Collection
2. Image Processing
3. Importing Dependencies
4. Importing Deep Learning Libraries
5. Resize the Image
6. Scale the Data
7. Normalization
8. Setting up the Layers of Neural Network
9. Activation Function
10. Compiling the Neural Network
11. Optimizer
12. Accuracy
13. Training the Neural Network
14. Model Evaluation
15. Final Predictions
16. Building the Confusion Matrix
17. Build the Predictive System
18. Resize and Reshape the Image

Dependencies:

- The NumPy library will produce the array from the images
- Matplotlib, numpy arrays will be visualized using this library. Once we plot the array on matplotlib we can see which image we have.
- Seaborn is also used for visualization
- cv2 is an open computer vision library. One of the important libraries used for image recognition tasks.
- TensorFlow and Keras
- Randomseeds
- Python Imaging Library (PIL)
- Confusion matrix from TensorFlow.

Performance Measurement:

MNIST Digit Classification is a well-known problem in the field of machine learning, where the objective is to classify handwritten digits from the MNIST dataset. In this project, we implemented a neural network model using Python and TensorFlow library to classify the digits. The neural network model consists of multiple layers, including an input layer, one or more hidden layers, and an output layer using convolutional neural networks (CNNs), and recurrent neural networks (RNNs) architectures.



After training the model on the MNIST dataset, we achieved an accuracy of around 97.1%, which is an excellent result for this task. We also tested the model on a separate set of images and found that it performs well in classifying new images accurately.

Performance Checks:

In conclusion, we can say that the neural network model we implemented is effective in classifying the MNIST digits with high accuracy. The model can be used for various applications, such as automatic digit recognition in postal services, banks, and other fields where handwritten digit recognition is required. Although we achieved high accuracy, there is still room for improvement by fine-tuning the model's hyperparameters or by implementing more advanced neural network architectures. Overall, this project demonstrates the effectiveness of neural networks in image classification tasks and highlights the potential of using machine learning for real-world applications.

Project Results:

```
Enter the path of the image to be predicted: /content/img_62.jpg
```

```
1/1 [==============================] - 0s 27ms/step
The Handwritten Digit In Image Is Reconginized As:  5
```

```
Enter the path of the image to be predicted: /content/img_1.jpg
```

```
1/1 [==============================] - 0s 23ms/step
The Handwritten Digit In Image Is Reconginized As:  0
```

```
Enter the path of the image to be predicted: /content/img_2.jpg
```

```
1/1 [==============================] - 0s 23ms/step
The Handwritten Digit In Image Is Reconginized As:  1
```

```
Enter the path of the image to be predicted: /content/img_64.jpg
```

```
1/1 [==============================] - 0s 20ms/step
The Handwritten Digit In Image Is Reconginized As:  6
```

```
Enter the path of the image to be predicted: /content/img_24.jpg
```

```
1/1 [==============================] - 0s 33ms/step
The Handwritten Digit In Image Is Reconginized As:  2
```

```
Enter the path of the image to be predicted: /content/img_76.jpg
```

```
1/1 [==============================] - 0s 24ms/step
The Handwritten Digit In Image Is Reconginized As:  7
```

```
Enter the path of the image to be predicted: /content/img_13.jpg
```

```
1/1 [==============================] - 0s 24ms/step
The Handwritten Digit In Image Is Reconginized As:  3
```

```
Enter the path of the image to be predicted: /content/img_67.jpg
```

```
1/1 [==============================] - 0s 46ms/step
The Handwritten Digit In Image Is Reconginized As:  8
```

```
Enter the path of the image to be predicted: /content/img_88.jpg
```

```
1/1 [==============================] - 0s 24ms/step
The Handwritten Digit In Image Is Reconginized As:  9
```

```
Enter the path of the image to be predicted: /content/img_39.jpg
```

```
1/1 [==============================] - 0s 33ms/step
The Handwritten Digit In Image Is Reconginized As:  4
```