CAP 4630 Assignment 5: Neural Network

Instructor: Aakash Kumar

Overview

In this assignment, you will work with the MNIST dataset to build, train, evaluate, and comment on neural network models using TensorFlow and Keras. You will also submit a report in PDF format containing results and explanations as specified below along with jupyter notebook file.

Instructions

This assignment is divided into multiple parts. Follow each part closely and ensure you meet the requirements outlined for grading.

Important Note

All the code for data preprocessing, accuracy and loss graphs, classification report, confusion matrix, and sample predictions (qualitative results showing a selection of predictions vs. true labels) has been provided in the Jupyter notebook. **Your main task is to write the code for building the neural network model in the ”Build the Neural Network Model” section as specified below.**

Part 1: Code Commenting (40 Marks)

In the provided Jupyter notebook, you'll find sections marked with #Comment:. For each of these, provide detailed comments explaining the purpose, functionality, and significance of the lines of code immediately following the #Comment: tag.

Your comments should be insightful and comprehensive. This means explaining not only what the code does but why it's used in that specific context. **This section is worth 40 marks.**

Part 2: Building and Training a Neural Network Model (20 Marks)

1. **Define the Neural Network:** In the section titled ”Build the Neural Network Model,” define a neural network with the following architecture:

   - Input layer
   - One hidden layer with 8 neurons
   - Output layer with a size matching the number of classes in the dataset (10 for MNIST).

   Use the `Sequential` class from `tensorflow.keras.models` to define the layers.

2. **Training and Evaluation:** Train the network using the code provided in the notebook, which will generate:

   - Accuracy and loss graphs
   - Classification report
   - Confusion matrix
   - Sample predictions (visual representation of model predictions on test data)

3. **Reporting:** Save all results, including graphs and reports, in a PDF file.

This section is worth 20 marks.

Part 3: Experimenting with a Larger Hidden Layer (20 Marks)

Repeat the steps in Part 2, but this time, modify the neural network by increasing the hidden layer size to 128 neurons.

1. **Define the Neural Network:** Redefine the network to have one hidden layer with 128 neurons (the output layer remains the same as in Part 2).

2. **Training and Evaluation:** Train this modified network, and use the provided code to generate:

   - Accuracy and loss graphs
   - Classification report
   - Confusion matrix
   - Sample predictions (qualitative results showing a selection of predictions vs. true labels)

3. **Reporting:** Save all results, including graphs and reports, in a PDF file.

This section is worth 20 marks.

Part 4: Custom Neural Network for 99% average F1 Score (10 Marks)

1. **Model Design:** Design a neural network of your choice (any architecture or parameters) to achieve an average F1 score of 99% or higher on the MNIST dataset if possible.
2. **Allowed Adjustments:** You may adjust the number of epochs and batch size to optimize the model's performance.
3. **Training and Evaluation:** Use the provided code to generate the same results as in Parts 2 and 3, including accuracy and loss graphs, classification report, confusion matrix, and sample predictions.
4. **Reporting:** Document your process, results, and achieved F1 score in the PDF.

Marks Distribution:

- 5 marks for successful implementation.
- 5 marks for achieving a 99% F1 score or higher.

Part 5: Discussion Questions (10 Marks)

In your PDF report, answer the following questions with 2-3 sentences for each point:

1. **Epochs:**
   - Define the term "epoch" in the context of training neural networks.
   - Explain why epochs are used.
   - Describe what happens when you increase or decrease the number of epochs (3 points).
2. **Batch Size:**
   - Define "batch size" and discuss its role in training neural networks.
   - Describe the effects of increasing or decreasing batch size (3 points).
3. **Dropout:**
   - Explain what Dropout is and how it helps in training neural networks (4 points).

Ensure clarity and precision in your responses.

Submission Requirements

- Submit the final Jupyter notebook file with your code.
- Submit a separate PDF document containing:
  - Results and visualizations for Parts 2, 3, and 4.
  - Responses to the discussion questions in Part 5.

Grading Summary

- **Part 1 (Comments):** 40 marks
- **Part 2 (8 Neurons Hidden Layer):** 20 marks
- **Part 3 (128 Neurons Hidden Layer):** 20 marks
- **Part 4 (Custom Model for 99% F1):** 10 marks (5 marks for implementation, 5 for achieving 99% F1 score)
- **Part 5 (Discussion):** 10 marks

Total: 100 Marks

Good Luck!

By following these instructions and meeting the requirements for each part, you'll complete the assignment successfully. Good luck!