LeNet-5 on MNIST (LeCun et al., 1998)

Members:

25M0808 Chittari Vineeth Babu
25M0807 G Jayavardhan
25M0802 R Ajay
25M0787 Pujith Sai Kumar Korlepara
25M0834 Harshith matta

1. Objective

The objective of this project is to reproduce the results of the classic paper "Gradient-Based Learning Applied to Document Recognition" (LeCun et al., 1998), which introduced the LeNet-5 convolutional neural network architecture for handwritten digit recognition on the MNIST dataset.

The project's objectives are as follows:

- To implement the LeNet-5 architecture from first principles, utilizing a contemporary deep learning framework such as PyTorch or TensorFlow.
- To replicate the performance metrics detailed in the original publication, specifically on the MNIST dataset.
- To conduct an analysis of how modifications to the architecture and training regimen influence model performance.
- To meticulously document the implementation specifics, challenges encountered, and insights acquired throughout the reproduction process.

2. Planned Approach

Step 1: Literature Review

Study the original LeCun et al. (1998) paper to understand the architecture, training procedure, and results. Review modern deep learning frameworks' implementations of CNNs for context.

Step 2: Dataset Preparation

Use the MNIST dataset (handwritten digits 0–9). Preprocess images (normalization, reshaping, etc.) according to LeNet-5 specifications.

Step 3: Model Implementation

Implement LeNet-5 using a modern framework such as PyTorch. Define all layers (convolutional, subsampling, fully connected). Match hyperparameters (learning rate, optimizer, loss function) as closely as possible to the paper.

Step 4: Training and Evaluation

Train the model using the training set (60,000 samples) and validate on the test set (10,000 samples). Measure accuracy, loss curves, and compare with reported paper results (~99% accuracy).

Step 5: Analysis and Visualization

Visualize feature maps and filters to interpret model learning. Compare results with other modern architectures (optional).

Step 6: Documentation and Report

Prepare a report describing methodology, experiments, and results. Include plots, training logs, and comparisons with the original paper.

3. Planned Timeline

Week	Duration	Tasks Planned
Week 1	Oct 5 – Oct 11	- Read and summarize the LeNet-5 paper - Review MNIST dataset and preprocessing pipeline - Finalize framework choice (PyTorch/TensorFlow)
Week 2	Oct 12 – Oct 18	- Implement LeNet-5 model architecture - Set up data loading and preprocessing code - Verify model forward pass
Week 3	Oct 19 – Oct 25	- Implement training loop (optimizer, loss function, scheduler) - Begin initial experiments on MNIST - Debug training process
Week 4	Oct 26 – Nov 1	- Fine-tune hyperparameters - Reproduce results close to the paper’s accuracy - Log training and validation metrics
Week 5	Nov 2 – Nov 8	- Analyze learned filters and feature maps - Visualize model’s intermediate outputs - Compare with modern CNNs (optional)
Week 6	Nov 9 – Nov 15	- Write report sections: Introduction, Methodology, and Experiments - Prepare tables, plots, and accuracy comparisons
Week 7	Nov 16 – Nov 22	- Finalize report and presentation - Review and polish results - Submit final documentation and code repository

4. Expected Outcome

- Successfully reproduced LeNet-5 achieving near-original accuracy (~99%) on MNIST.
- Comprehensive understanding of early CNN architectures and their influence on modern models.
- Well-documented codebase and project report for educational and reproducibility purposes.

5. References

- **Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner.**

  Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.

  http://yann.lecun.com/exdb/publis/pdf/lecun-98.pdf