INTRO TO GEOMETRIC DEEP LEARNING FINAL PROJECT

Submission Deadline: October 15, 2024

Submission Format: A single ZIP file containing your code, report, and result files. Each question should be in a sperate folder with the report and code within this folder. The ZIP name should be Intro2GDL_hw1_hw2. The folders names should be Q1, Q2.

Grading: Projects will be graded on correctness, clarity, and creativity. Bonus points will be awarded for using methods that were not covered in class. Any attempt to cheat will result in serious consequences.

TASK 1: POINT CLOUD CLASSIFICATION

Dataset: ModelNet10 (accessible through PyTorch Geometric)

Objective: Train a point cloud classification model using a method of your choice. The model should classify objects from the ModelNet10 dataset, which contains 3D CAD models categorized into 10 classes.

INSTRUCTIONS:

1. Dataset Loading: Load the ModelNet10 dataset using PyTorch Geometric’s built-in functionality.

2. Model Selection: Choose or design a model to perform point cloud classification. You may use well-known architectures such as PointNet, PointNet++, DGCNN or any other relevant model.

3. Training: Train your model on the dataset, you must use `seed=42` for reproducibility.

4. Evaluation: Evaluate your model on the test set and report the accuracy.

5. Report: Submit a report explaining your method, architecture choice, training procedure, and results. Highlight any novel methods or modifications if they differ from those taught in class. The report should include images of the point clouds you successfully classified and for the ones you failed. Try to explain why.

6. Bonus: You will receive bonus points for implementing a method not covered in class.

DELIVERABLES:

- A Python script (`main.py`) that produce the entire training and evaluation predications.

- A report summarizing your approach and findings, report.pdf.

Note: Any result that cannot be reproduced using the submitted code will be graded zero. Do not include pre-trained models or pickled files. You must run on the allocated server.

TASK 2: GRAPH CLASSIFICATION

Dataset: The dataset is in the homework folder, named: processed_data.pt. This is a PyG data object. It contain train, test, validation sets where the test don't have labels.

Objective: Train a graph classification model to classify the graphs.

INSTRUCTIONS:

1. Dataset Loading: Load the dataset.

2. Model: Choose or design a graph neural network as you wish.

3. Training: Train your model on the provided dataset with `seed=42`.

4. Evaluation: Evaluate your model on the validation set and submit a file name predication.csv with coloums idx, label, containing the predicted classes for each graph in the test set.

5. Report: Provide a detailed explanation of your method, including your choice of model and hyperparameters. Discuss your results in the context of the dataset. You are required to achieve results higher than 75% on the validation.

6. Bonus: Implementing a model not discussed in the course will earn you additional points.

DELIVERABLES:

- A Python script (`main.py`) that trains and evaluates your model, producing a result file containing the predictions for the test set.

- The result file (`predictions.txt`) containing the class predictions for each test graph.

- A report detailing your approach and results, report.pdf.

Grading: Your grade will be determined in a tournament-style evaluation, where the highest classification accuracy receives the highest grade. Results must be fully reproducible.

IMPORTANT NOTES:

Reproducibility: Ensure that your code is fully reproducible by anyone running `main.py` with `seed=42`.

Server Usage: All models must be trained on the provided server environment. Training on personal machines will result in a.

Deadline: Late submissions will not be accepted. Ensure your project is submitted before October 15, 2024.

ADDITIONAL RESOURCES:

For guidance on working with the ModelNet10 dataset, refer to the tutorial available here.