Deep Learning – Assignment II

Prof. Chih-Chung Hsu

TA. Yi-Fan Wang, Wei-Hao Huang, Wo-Ting Liao, Jia-Ming Li

I. Dataset

- Dataset URL: https://cchsu.info/files/images.zip
- The mini-ImageNet dataset is a smaller version of the ImageNet dataset, created to reduce the computational complexity while retaining the essential properties of the larger dataset.

- **The data is organized as follows:**
  - train.txt: Contains the file paths and labels for the training images.
  - validation.txt: Contains the file paths and labels for the validation images.
  - test.txt: Contains the file paths and labels for the test images.

- **Validation Performance:** During training, you must evaluate your model using the validation set. Report the performance metrics (e.g., accuracy, loss) on the validation set to monitor the model's performance and to tune hyperparameters.

- **Final Performance**: After completing the training, evaluate your model using the test set. Report the final performance metrics based on the test set. This ensures that the reported performance reflects the model's ability to generalize to unseen data.

II. Task: Designing a Convolution Module for Variable Input Channels

- **Description:**

  Design a special convolutional module that is spatial size invariant and can handle an arbitrary number of input channels. You only need to design this special module, not every layer of the CNN. After designing, explain the design principles, references, additional costs (such as FLOPS or #PARAMS), and compare with naive models. To simulate the practicality of your method, use the ImageNet-mini dataset for training, and during the inference process, test images with various channel combinations (such as RGB, RG, GB, R, G, B, etc.) and compare the performance.

- **Detailed Example:**

1. Design Principles:

Dynamic Convolution: Design a dynamic convolution module that can adjust its weights based on the number of input channels. This can be achieved by learning a weight-generating network that takes the input channels as an input and generates corresponding convolution kernels.

2. Implementation Details:

- Convolution Module: Use a dynamic weight-generating network. This network receives the number of input channels and generates convolution kernels accordingly. These kernels can then be used for standard convolution operations.

- Reference: The concept of dynamic convolution can be referenced from the paper 'Dynamic Convolution: Attention over Convolution Kernels' by Wu et al., CVPR 2020. Cost Analysis: Compared to naive models, this design might increase additional computational costs (such as FLOPS) and parameters but can significantly reduce the need for training and storing models for different channel numbers.

3. Experiment Design:

Training and Inference: Use the ImageNet-mini dataset for training and test the model on images with various channel combinations during inference. Comparison: Compare the performance of the model using the dynamic convolution module with naive models across different input channel combinations, evaluating accuracy and computational cost.

III. Designing a Two-Layer Network for Image Classification

- **Description:**

  Design a two-layer CNN, Transformer, or RNN network that can achieve 90% performance of ResNet34 on ImageNet-mini (i.e., with no more than 10% performance loss). There are no restrictions on parameter count or FLOPS, but the maximum number of input and output layers is limited to 4. Explain the design principles, references, and provide experimental results.

- **Detailed Example:**

  1. Design Principles:

  Increasing Receptive Field: Use techniques such as RRDB, self-attention mechanisms, or Graph Convolutional Networks (GCN) to increase the network's receptive field and enhance feature extraction capabilities.

  2. Implementation Details:

  - RRDB Module: Implement a Residual-in-Residual Dense Block (RRDB) to increase the receptive field and enhance feature extraction.
  - Reference: The design concept can be referenced from 'ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks' by Wang et al., ECCV 2018.
  - Attention Mechanism: Incorporate self-attention mechanisms to improve the network's ability to capture global features.
  - Cost Analysis: This design might increase computational costs (such as FLOPS) and parameter count but can improve performance within the constraint of limited layers.

  3. Experiment Design:

  - Training and Inference: Use the ImageNet-mini dataset for training and evaluate the network's performance on image classification tasks.
  - Comparison: Compare the designed two-layer network with ResNet34 on ImageNet-mini, evaluating accuracy and computational cost.

- **Additional Tips:**

  1. Increasing Receptive Field: RRDB: Use RRDB to enhance the network's receptive field.
  2. Attention Mechanism: Incorporate self-attention mechanisms to improve the capture of global features.
  3. GCN: Use Graph Convolutional Networks to handle structured information in images and enhance feature extraction.

IV. Evaluation Criteria

- **Theoretical Justification (30%):**

  Provide a clear and coherent explanation of the proposed solution, highlighting how it combines the strengths of DDPM and DIP. Justify the design choices and assumptions made in the proposed approach. Discuss the potential benefits and limitations of the proposed solution compared to using DDPM or DIP alone.

- **Experimental Verification (40%):**

  Implement the proposed solution and conduct experiments to validate its effectiveness. Compare the performance of the proposed approach with standalone DDPM and DIP methods in terms of either image quality, generation speed, or both. Provide quantitative metrics to support the claims, such as PSNR, SSIM, FID, or generation time. Present qualitative results showcasing the visual quality of the generated or reconstructed images. Analyze the experimental results and discuss the observed improvements or trade-offs compared to the baseline methods.

- **Ablation Studies and Analysis (30%):**

  Conduct ablation studies to investigate the impact of different components or hyperparameters in the proposed solution. Vary the key parameters, such as noise levels, denoising schedules, or architectures, and evaluate their influence on the performance. Provide insights and interpretations based on the ablation studies, justifying the chosen configurations.

- **Note:**

  The focus of this assignment is on demonstrating the effectiveness of combining DDPM and DIP techniques, rather than achieving state-of-the-art performance. The proposed solution should show improvements in either image quality, generation speed, or both, compared to using DDPM or DIP individually. The claims made in the solution should be supported by theoretical justifications and experimental verification.

  The evaluation criteria emphasize the importance of providing a clear theoretical justification for the proposed approach, conducting thorough experiments to validate its effectiveness, and presenting the work in a well- organized and understandable manner. You are expected to provide quantitative and qualitative results, perform ablation studies to analyze the impact of different components, and discuss the observed improvements or trade- offs compared to the baseline methods.

  By meeting these evaluation criteria, you can demonstrate their understanding of the DDPM and DIP techniques, their ability to combine them effectively, and their skills in conducting rigorous experiments and analysis.

V. Submission Requirements

- **GitHub Repository (50%):**
  1. Create a GitHub repository to host your implementation and related files.
  2. Include well-documented and organized code for your proposed solution.
  3. Provide clear instructions in the README.md file on how to run the code and reproduce the experiments.
  4. Use appropriate git commit messages and branches to track the development progress.
  5. Ensure that the repository is accessible to the instructor and teaching assistants.

- **Report (50%):**
  1. Write a comprehensive report describing your proposed solution, experiments, and findings.
  2. The report should be in a format of your choice (e.g., PDF, Markdown, LaTeX) and can be written in any preferred language.

- **Submission Deadline:**
  1. The GitHub repository link and the report should be submitted via Moodle by 2024.6.13.
  2. Late submissions will be subject to the course's late submission policy.