

# CNN Architecture

## Assignment Questions



## TOPIC: Understanding Pooling and Padding in CNN

1. Describe the purpose and benefits of pooling in CNN.
2. Explain the difference between min pooling and max pooling.
3. Discuss the concept of padding in CNN and its significance.
4. Compare and contrast zero-padding and valid-padding in terms of their effects on the output feature map size.

## TOPIC: Exploring LeNet

1. Provide a brief overview of LeNet-5 architecture.
2. Describe the key components of LeNet-5 and their respective purposes.
3. Discuss the advantages and limitations of LeNet-5 in the context of image classification tasks.
4. Implement LeNet-5 using a deep learning framework of your choice (e.g., TensorFlow, PyTorch) and train it on a publicly available dataset (e.g., MNIST). Evaluate its performance and provide insights.

## TOPIC: Analyzing AlexNet

1. Present an overview of the AlexNet architecture.
2. Explain the architectural innovations introduced in AlexNet that contributed to its breakthrough performance.
3. Discuss the role of convolutional layers, pooling layers, and fully connected layers in AlexNet.
4. Implement AlexNet using a deep learning framework of your choice and evaluate its performance on a dataset of your choice.

## Submission Guidelines:

- Answer all the questions in a single Jupyter Notebook file (.ipynb).
- Include necessary code, comments, and explanations to support your answers and implementation.
- Ensure the notebook runs without errors and is well-organized.
- Create a GitHub repository to host your assignment files.
- Rename the Jupyter Notebook file using the format "date\_month\_topic.ipynb" (e.g., "12\_July\_CNN\_Basics.ipynb").
- Place the Jupyter Notebook file in the repository.
- Commit and push any additional files or resources required to run your code (if applicable) to the repository.
- Ensure the repository is publicly accessible.
- Submit the link to your GitHub repository as the assignment submission.

## Grading Criteria:

1. Understanding and completeness of answers: 40%
2. Clarity and depth of explanations: 25%
3. Correct implementation and evaluation of optimizer techniques: 15%
4. Analysis and comparison of different optimizers: 10%
5. Proper code implementation and organization: 10%

**Note:** Create your assignment in Jupyter notebook and upload it to GitHub & share that uploaded assignment file link through your dashboard. Make sure the repository is public.

