AIN422 Introduction to Deep Learning Laboratory

Assignment 2: Convolutional Neural Network
Date Issued : 28.04.2023
Date Due : 12.05.2023

Aim of the Experiment

In this assignment, we will focus on Convolutional Neural Network (CNN), which is a type of neural network commonly used in image and video recognition, processing and analysis. First you will build a basic CNN model and then you will develop this model using different methods and techniques. The assignment should be implemented as a single Jupyter Notebook. Your notebook should be clearly documented, using comments and Markdown cells to explain the code and results. At the end of this exercise, you will become familiar with basic CNN pipeline.

Convolutional Neural Network

Convolutional Neural Network is a type of artificial neural network that is primarily used for image classification, segmentation, and object detection tasks. CNNs are specifically designed to process input data with a grid-like topology, such as images. Some of the key advantages of CNNs are their ability to automatically learn features from raw input data, their ability to handle input data of different sizes and shapes, and their effectiveness at handling high-dimensional data such as images. CNNs consist of multiple layers of convolutional and pooling operations, followed by fully connected layers. The key components of a CNN are:

- ***Convolutional Layers:*** Convolutional layers are the core building blocks of a CNN. These layers apply a set of learnable filters (also known as kernels) to the input image to produce a feature map. Each filter learns to detect a specific feature in the image, such as edges, corners, or textures. The feature maps produced by these filters are then passed on to the next layer in the network.

- ***Pooling Layers:*** Pooling layers are typically used after convolutional layers to reduce the spatial size of the feature maps and to make the network more computationally efficient. These layers downsample the feature maps by applying a pooling function (such as max pooling or average pooling) to non-overlapping patches of the feature maps.

- ***Activation Functions:*** Activation functions introduce non-linearity into the network and allow it to learn complex relationships between the input and output. Common activation functions used in CNNs include ReLU, sigmoid, and tanh.

- ***Fully Connected Layers:*** Fully connected layers are used at the end of a CNN to classify the input image. These layers take the flattened output from the previous layer and apply a set of learnable weights to produce a class score for each possible output class.

- ***Dropout:*** Dropout is a regularization technique that is often used in CNNs to prevent overfitting. It works by randomly dropping out (setting to zero) a percentage of the neurons in the network during each training iteration, forcing the network to learn more robust features.

Experiment

1. Download the MNIST (alternative link) dataset. Write your own data loader function and load data without using a library.
2. Perform preprocessing steps that may be necessary to clean or filter the data.
3. Analyze the dataset using tables and graphs. Clearly explain analysis results.
4. Do not split the data at different training and testing rates. Use default train and test data.
5. Design your own CNN model. Your CNN should consist of at least one convolutional layer, one pooling layer, and one fully connected layer. Also you must determine kernel-size, padding, stride parameters etc. Do not use any well-known architecture.
6. Present the classification results (classification accuracy, precision, recall, F1 measure and confusion matrix).
7. Your analysis must be on layer size, batch-size, learning rate, activation function. You should change the number and order of your layers and also use different batch size, learning rate, activation function. You should interpret your results. Try to find the best batch-size, learning rate and activation function.
8. Present the classification results (classification accuracy, precision, recall, F1 measure and confusion matrix).
9. Compare the performance of CNN models using tables and graphs.
10. In your report, you will summarize the results of your experiments and draw conclusions about the impact of the methods you used on the performance of the CNN model. You should also provide recommendations on which techniques are most effective and when they should be used.

11. You should submit your codes as .ipynb format (use Jupyter or Colab Notebook) and your report file as pdf format in a single zip file.

Background information

We provide with you some references.

- https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/
- https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/
- https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939

Grading

- Import dataset and Preprocessing (%20)
- Visualization (%10)
- Implementing methods (%40)
- Report (%30)

REMARKS:

- Submission format:
  - <studentID.zip>
    - studentID_name_surname_hw2.ipynb (Required)
    - studentID_name_surname_hw2_report.pdf (Required)

- Your submission should be matched with the format above**. 10 point** penalty will be applied on mismatched submissions.
- You will use an online submission system to submit your experiments.
- https://submit.cs.hacettepe.edu.tr/ Deadline is 23:59. No other submission method (such as; CD or email) will be accepted.
- Do not submit any file via email related to this assignment.
- The assignment must be original, INDIVIDUAL work. Duplicate or very similar assignments are both going to be punished. General discussion of the problem is allowed, but DO NOT SHARE answers, algorithms, or source codes.
- You can ask your questions through the course's Piazza group and you are supposed to be aware of everything discussed in the group.