Gradient-Based Learning Applied to Document Recognition

Team Number: 20 | Team Name: Story of SMAI Life

Team Members:

Abhijeeth Reddy Singam (2019101065)
Ainesh Sannidhi (2019101067)
Kunwar Maheep Singh (2019101075)
Rishabh Khanna (2019113025)

Github Link: https://github.com/inesane/Lenet

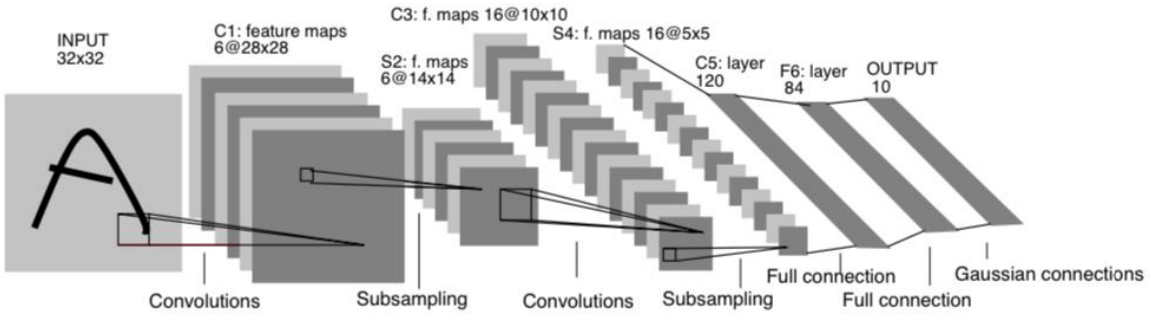
Problem Statement

Document recognition by virtue of character recognition is a problem that brings together the fields of computer vision, pattern recognition and machine learning. Character recognition is important in data extraction, especially from written text for the conversion of printed paper documents to machine-readable documents, such as bank cheques. Traditionally, machine learning techniques, especially neural networks, have been widely used when creating pattern recognition systems such as in optical character recognition and this paper aims to show how better pattern recognition systems can be made by focusing more on machine learning techniques as opposed to traditional hand-designed heuristics.

These techniques include using various Classifiers, Multilayer Neural Networks and Graph Transformer Networks which use Convolutional Neural Network character recognizers and global training techniques which give us a much better character recognition system than if we were to manually carry out the process of feature extraction.

Goals & Approach

The overall goal consists of implementing LeNet-5 from scratch.



Architecture of LeNet-5

LeNet-5 is a Convolutional Neural Network that consists of 7 different layers, other than the input. It takes 32x32 pixel images as input.

1. C1 - Layer C1 is a convolution layer with six convolution kernels of 5x5 and the size of feature mapping is 28x28. It contains 156 trainable parameters and 122,304 connections.
2. S2 - Layer S2 is the subsampling/pooling layer with size 2x2 that outputs 6 feature graphs of size 14x14. It has 12 trainable parameters and 5,880 connections.
3. C3 - Layer C3 is a convolution layer with 16 5x5 convolution kernels. It has 1,516 trainable parameters and 151,600 connections.
4. S4 - Layer S4 is a subsampling/pooling layer with size of 2x2 and output of 16 5x5 feature graphs. It has 32 trainable parameters and 151,600 connections.
5. C5 - Layer C5 is a convolution layer with 120 convolution kernels of size 5x5. It has 48,120 trainable connections.
6. F6 - Layer F6 is fully connected to C5, and 84 feature graphs are output. It has 10,164 trainable parameters.

Dataset

MNIST Dataset: http://yann.lecun.com/exdb/mnist/

Expected Deliverables

Implementing every aspect and component of LeNet-5 from scratch. This includes:

- Convolutions
- Kernels
- Pooling
- Subsampling
- Fully connected layers
    - Input layer
    - Hidden layers
    - Output layer
- Loss and Activation Functions
- Forward and Backward Pass

Milestones and Timeline

1st Nov - 10th Nov: Literature Review
10th Nov - 15th Nov: Implementing basic code layout without functionality, and implementing forward pass for all layers (Convolution, Pooling and Dense).
16th Nov : Prepare Presentation for mid-evaluation.
17th Nov - 20th Nov: Mid-Evaluation
21st Nov - 25th Nov: Implementing suggestions from Mid-Evaluation
25th Nov - 28th Nov: Implementing backward pass for all layers and testing.
29th Nov - 30th Nov: Create Final Presentation and Final Report
1st Dec - 4th Dec: Final Presentation
4th Dec: Final Report Submission

Work Distribution

All members will collectively work on the evaluation presentations/reports and general aspects of implementation such as testing and basic code layout.

Abhijeeth: Loss Function, Backward Pass in Convolution Layer
Ainesh: Forward Pass in Pooling Layer, Backward Pass in Dense Layer
Kunwar: Forward Pass in Dense Layer, Backward Pass in Pooling Layer
Rishabh: Activation Function, Forward Pass in Convolution Layer

References

http://vision.stanford.edu/cs598_spring07/papers/Lecun98.pdf