Review: LeNet-1, LeNet-4, LeNet-5, Boosted LeNet-4 (Image Classification)

Sik-Ho Tsang Aug 8, 2018 · 6 min read

There are already plenty of literature review talk about the LeNet which is a classical image classification deep learning convolutional neural network. But most of them just review one of the versions, i.e. LeNet-5. LeNet-1, LeNet-4 and Boosted LeNet-4 are usually ignored. In this story, I will have an brief review on the network architectures of:

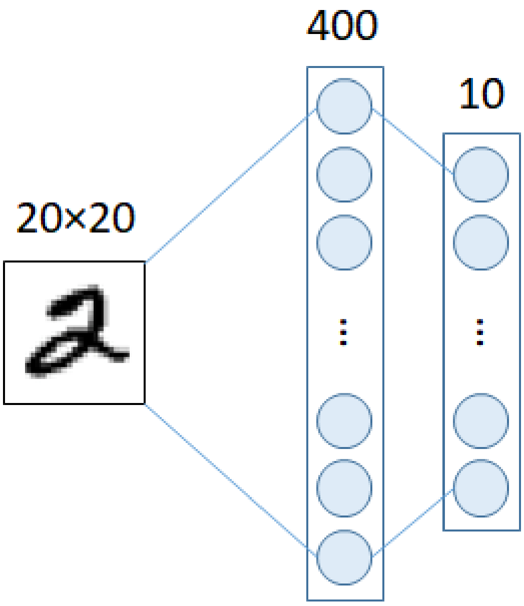
1. **Baseline Linear Classifier**
2. **One-Hidden-Layer Fully Connected Multilayer NN**
3. **Two-Hidden-Layer Fully Connected Multilayer NN**
4. **LeNet-1**
5. **LeNet-4**
6. **LeNet-5**
7. **Boosted LeNet-4**

The architectures and performances are presented in this story.

This is a **1998 Proceedings of the IEEE** journal with **about 14000 citations** when I was writing this story. Though it is from 1998, it starts by fundamentals of neural network which is a paper to have a good start to learn about deep learning. By looking at the development of neural network architecture and the error rate reduction, we can easily know the importance or influence of adding deep learning components onto the network. (Sik-Ho Tsang @ Medium)

It is assumed that we've acquired the basic knowledge about the deep learning components such as convolutional layer, pooling layer, fully connected layer, and activation function.

1. Baseline Linear Classifier

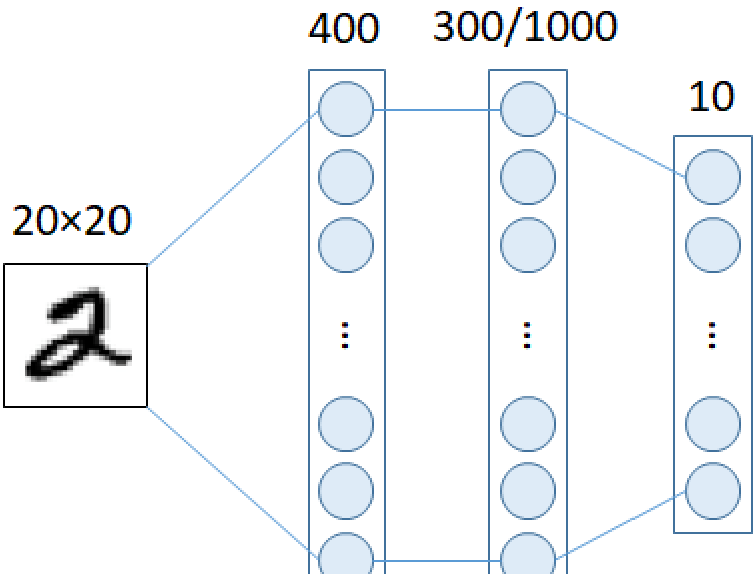


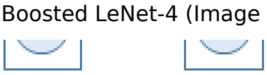
Baseline Linear Classifier

The simplest linear classifier. Each input pixel value contributes to a weighted sum for each output unit. The output unit with the highest sum indicates the class of the input character. Thus, as we can see, the image is treated as a 1D vector and connected to a 10-output vector.

By this mean, the **error rate** on test data is **8.4%**.

2. One-Hidden-Layer Fully Connected Multilayer NN

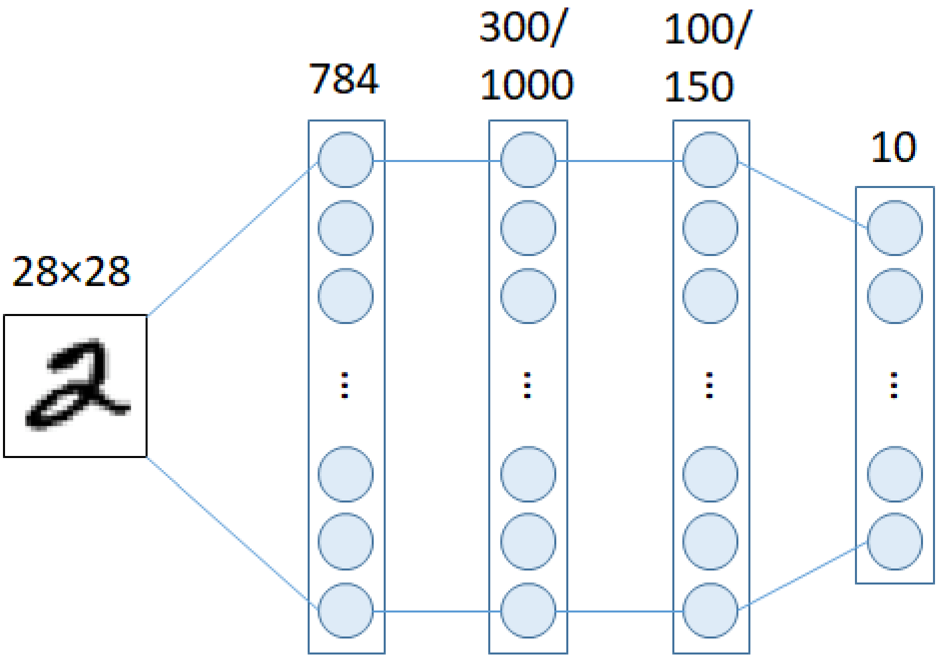




One-Hidden-Layer Fully Connected Multilayer NN

By adding one hidden layer in between the input layer and output layers, with 300 neurons at the hidden layer, i.e. **20×20–300–10 network**, the **error rate** on test data is **3.6%**. With 1000 neurons at the hidden layer, i.e. **20×20–1000–10 network**, the **error rate** on test data is **3.8%**.

3. Two-Hidden-Layer Fully Connected Multilayer NN

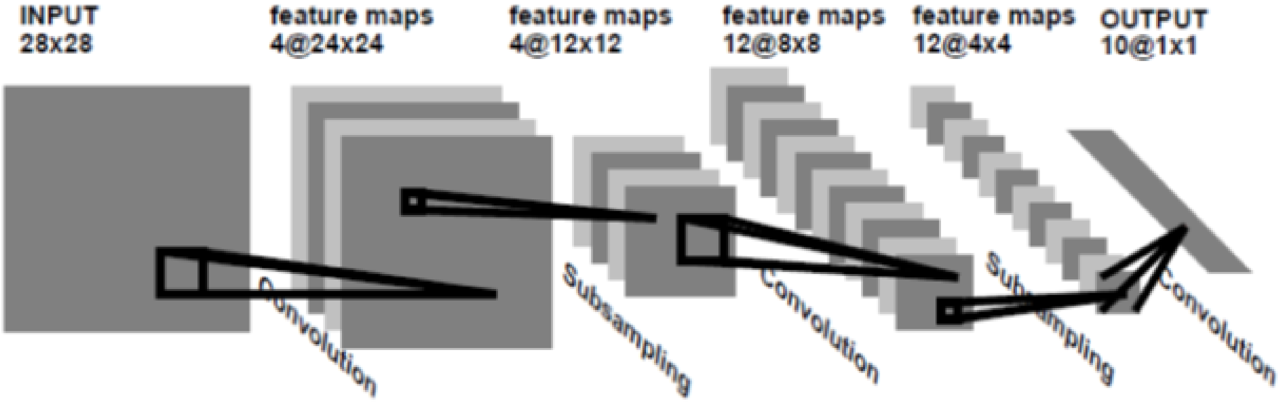


Two-Hidden-Layer Fully Connected Multilayer NN

By adding two hidden layers in between the input layer and output layers, **28×28–300–100-10 network**, the **error rate** on test data is **3.05%**. **28×28–1000–150–10 network**, the **error rate** on test data is **2.95%**.

We can observe that, by adding hidden layers, the error rate is getting smaller. But the improvement is also getting slow as well.

4. LeNet-1



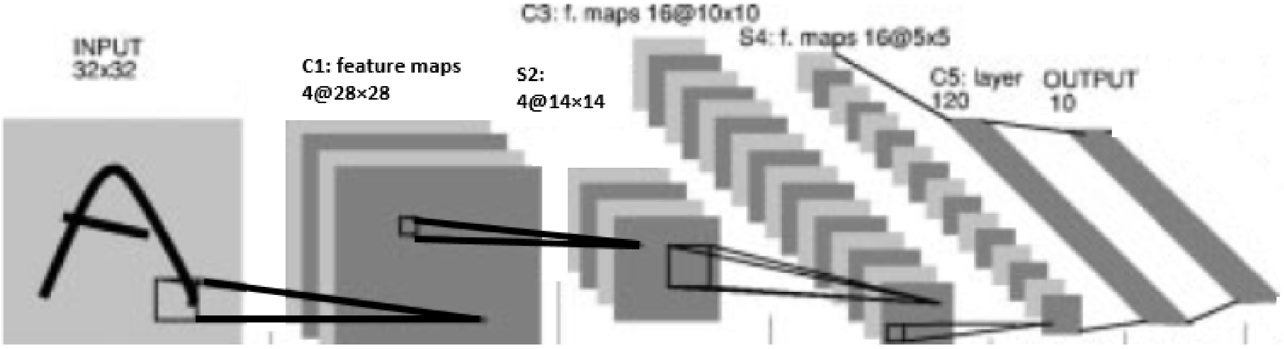
LeNet-1

In Lenet-1,
28×28 input image >
Four 24×24 feature maps convolutional layer (5×5 size) >
Average Pooling layers (2×2 size) >
Eight 12×12 feature maps convolutional layer (5×5 size) >
Average Pooling layers (2×2 size) >
Directly fully connected to the output

With convolutional and subsampling/pooling layers introduced, LeNet-1 got the **error rate** of **1.7%** on test data

It is noted that, at the moment authors invented the LeNet, they used average pooling layer, output the average values of 2×2 feature maps. Right now, many LeNet implementation use max pooling that only the maximum value from 2×2 feature maps is output, and it turns out that it can help for speeding up the training. As the strongest feature is chosen, larger gradient can be obtained during back-propagation.

5. LeNet-4

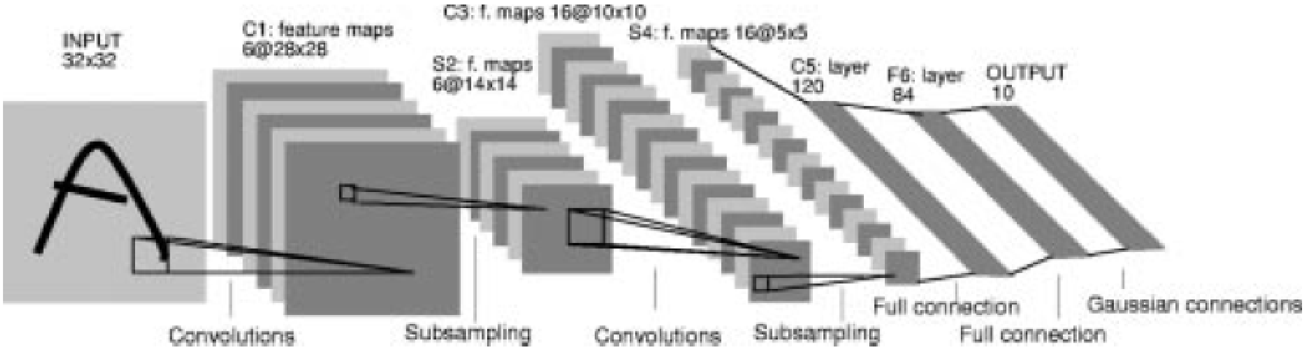


Lenet-4

In Lenet-4,
32×32 input image >
Four 28×28 feature maps convolutional layer (5×5 size) >
Average Pooling layers (2×2 size) >
Sixteen 10×10 feature maps convolutional layer (5×5 size) >
Average Pooling layers (2×2 size) >
Fully connected to 120 neurons >
Fully connected to 10 outputs

With more feature maps, and one more fully connected layers, **error rate** is **1.1%** on test data.

6. LeNet-5



LeNet-5

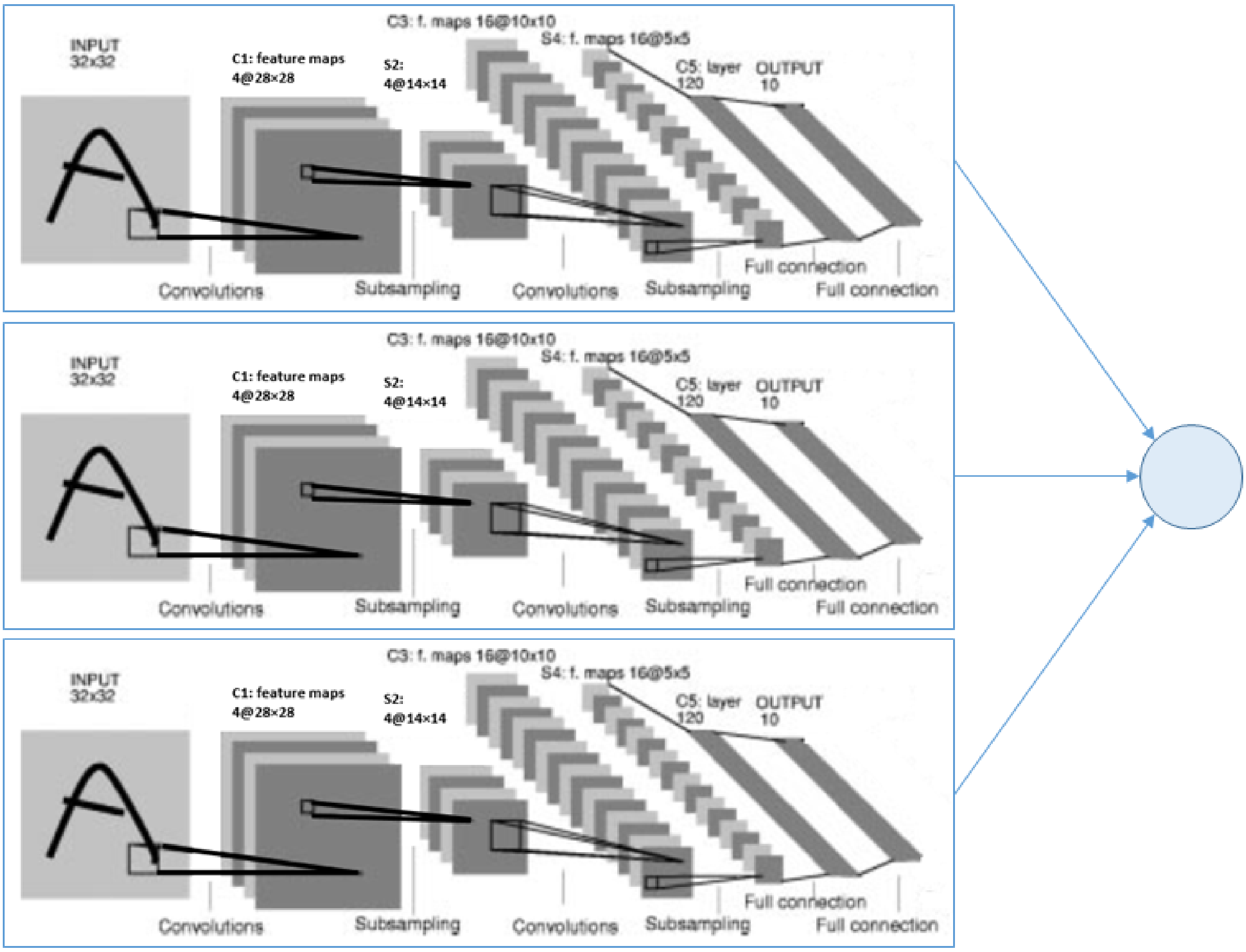
LeNet-5, the most popular LeNet people talked about, only has slight differences compared with LeNet-4.

32×32 input image >
Six 28×28 feature maps convolutional layer (5×5 size) >
Average Pooling layers (2×2 size) >
Sixteen 10×10 feature maps convolutional layer (5×5 size) >
Average Pooling layers (2×2 size) >

Fully connected to 120 neurons >
Fully connected to 84 neurons >
Fully connected to 10 outputs

With more feature maps, and one more fully connected layers, **error rate** is **0.95%** on test data.

7. Boosted LeNet-4



Boosted LeNet-4

Boosting is a technique to combine the results from several/many weak classifiers to get a more accurate results. In LeNet-4, the outputs of three LeNet-4 are simply added together, the one with maximum value would be the predicted classification class. And there is an enhancement that when the first net has a high confidence answer, the other nets would not be called.

With boosting, the **error rate** on test data is **0.7%** which is even smaller than that of LeNet-5.

This boosting technique has been used for years, until now.

8. Summary on Error Rate

1. **Baseline Linear Classifier: 8.4%**
2. **One-Hidden-Layer Fully Connected Multilayer NN: 3.6% to 3.8%**
3. **Two-Hidden-Layer Fully Connected Multilayer NN: 2.95% to 3.05%**
4. **LeNet-1: 1.7%**
5. **LeNet-4: 1.1%**
6. **LeNet-5: 0.95%**
7. **Boosted LeNet-4: 0.7%**

We can see that, the error rate is reducing while adding more deep learning components or some machine learning techniques.

9. Discussions

In these few papers [1–3], actually, there are many basic techniques about deep learning described in details. Also, different versions of LeNet also are even compared with other conventional approaches such as PCA, k-NN, SVM.

Something we need to note about is that:

9.1. Activation Function

Tanh is used as activation function except at the output.
Sigmoid is used as activation function at the output.
ReLU was NOT used in those years.

[Nowadays] And later on, ReLU is found to be a better activation function to accelerate the convergence during training.

9.2. Pooling Layer

As mentioned, LeNet used **average pooling instead of max pooling**.

[Nowadays] Max pooling is much common, or even no pooling layers.

9.3. Hidden Layers

At the old time, the number of hidden layers is few and the performance cannot be boosted up too much by adding more layers.

[Nowadays] It can be hundreds or thousands of hidden layers.

9.4. Training Time

It took days for training at the old time.

[Nowadays] But right now, it is just a small network with GPU acceleration.

10. Conclusions

To conclude, LeNet papers are really worth reading especially for the beginners in deep learning.

If interested, there is also a tutorial about LeNet-5 quick setup using Nvidia-Docker and Caffe [4].

References

1. [1989 NIPS] [LeNet-1]
   Handwritten Digit Recognition with a Back-Propagation Network

2. [1995 ICANN] [LeNet-1, LeNet-4, LeNet-5, Boosted LeNet-4]
   Comparison of Learning Algorithms for Handwritten Digit Recognition

3. [1998 Proc. IEEE] [LeNet-1, LeNet-4, LeNet-5, Boosted LeNet-4]
   Gradient-Based Learning Applied to Document Recognition

4. VERY QUICK SETUP of LeNet-5 for Handwritten Digit Classification Using Nvidia-Docker 2.0 + CUDA + CuDNN + Jupyter Notebook + Caffe