CNN on MNIST Dataset

Assignment Report

at

IIIT-A

By

Anurag Kumar (IIT2015143)

Shivam Kumar(IIT2015146)

(A University Established under sec.3 of UGC Act, 1956 Vide Notification No. F.9-4/99-U.3 Dated 04.08.2000 Of the Govt. Of India)
(A Centre of Excellence in Information Technology Established by Govt. of India)

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,
ALLAHABAD
(Deemed University)
(A Centre of excellence in IT, established by Ministry of HRD, Govt. of India)

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,
ALLAHABAD

(Deemed University)(A Centre of excellence in IT, established by Ministry of HRD, Govt. of India)

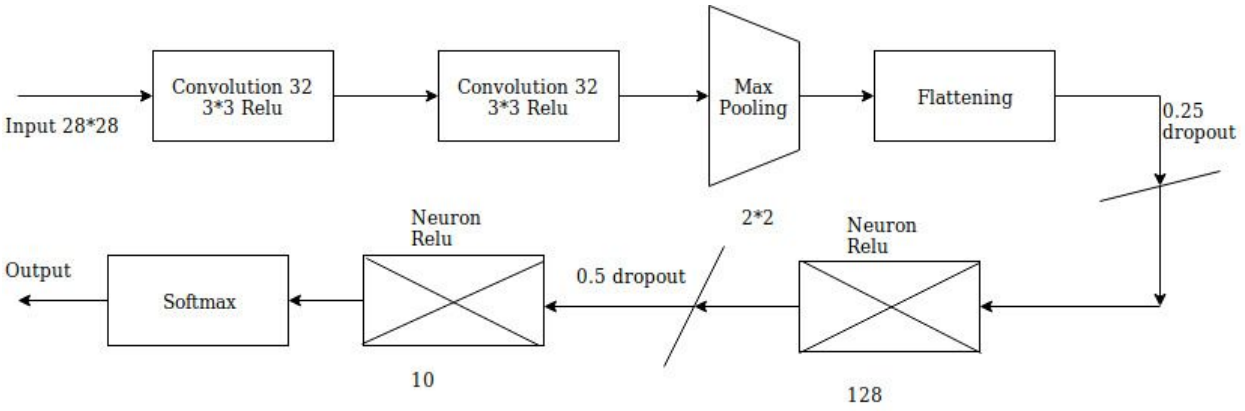
Table of Contents

INTRODUCTION

“Convolution Neural Network” , a buzz word and evidently a combination of biological and mathematical term along with computer science involved in it, is doing very well in terms of innovations. A lot of research papers have been published to define the beauty and to enhance pattern recognition capability of these networks.

The applications of CNNs are immense but here we will explore this network, while solving a problem based on MNIST dataset, a well known dataset in the field of Deep Learning. Here we will be examining a simple 2-D Convolutional Neural Network (CNN) model, designed using keras with tensorflow backend for MNIST digit recognition task.

ARCHITECTURE

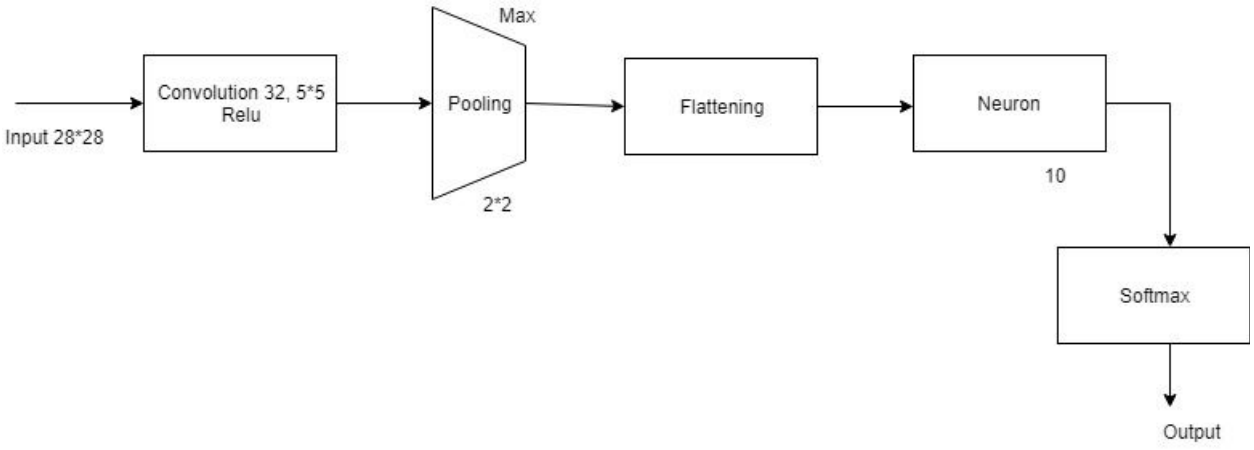


The architecture we have used is taking in input the 28 x 28 pixel image of the digits contained in MNIST dataset. First of all in convolution layer 32 filters of dimension 3 x 3 with RELU activation is applied two times. Then max pooling is done with dimension 2 x 2 and stride 2. Then we flatten the output to convert the output of previous layer to vector form so that it can be feed to a neural network. We perform 25% dropout and feed the input to a layer of 128 neurons with activation RELU. The output is again used as input to second layer after a drop out of 50%. The second layer has 10 neurons which after softmax give us the probability of out ten classes(the digits).

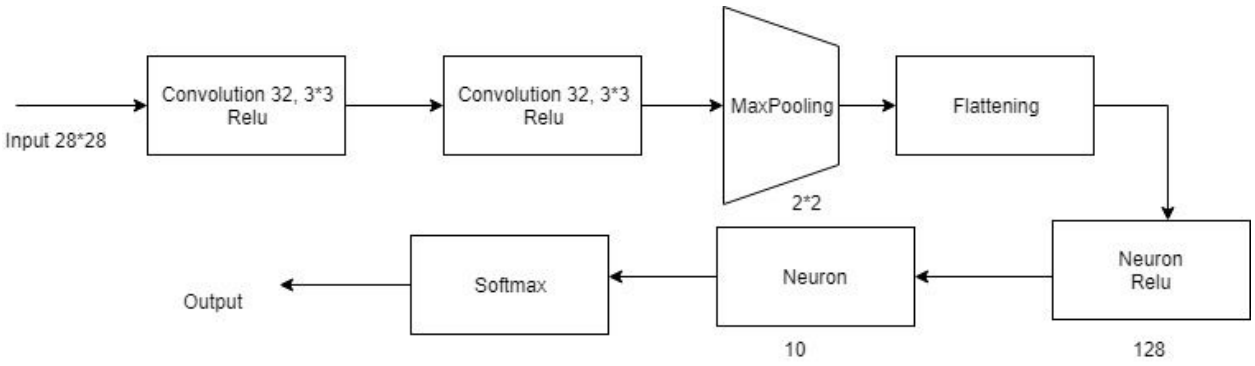
EXPERIMENTS

While doing the assignment we did experiment with several other architectures too which gave good results. But the last one with drop out was better than others so we stopped.

Several architecture we worked on were :



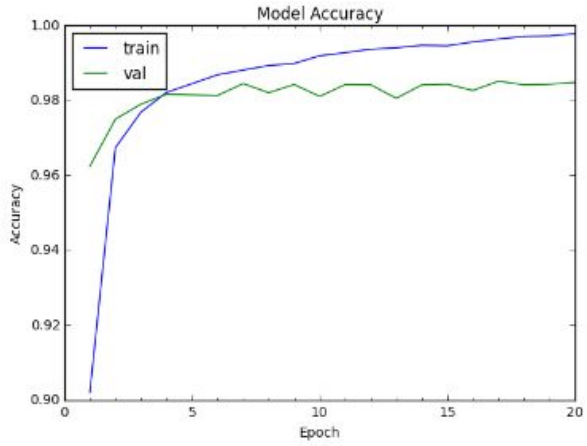
And

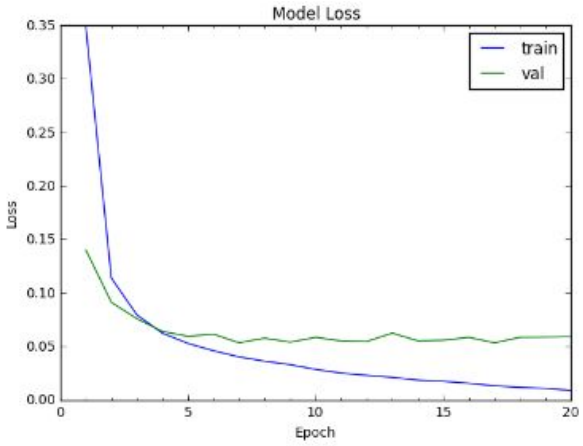


ANALYSIS

As in experiments section we tried two three networks and stopped at the last one. While tweaking the architecture we analysed that:

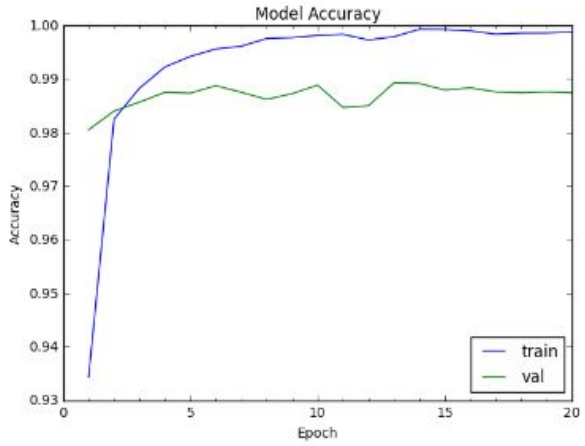
- Our first architecture where we did convolution followed with pooling followed with flattening and finally feed the output to a layer of 10 neurons which after softmax were producing the result, was giving good accuracy.

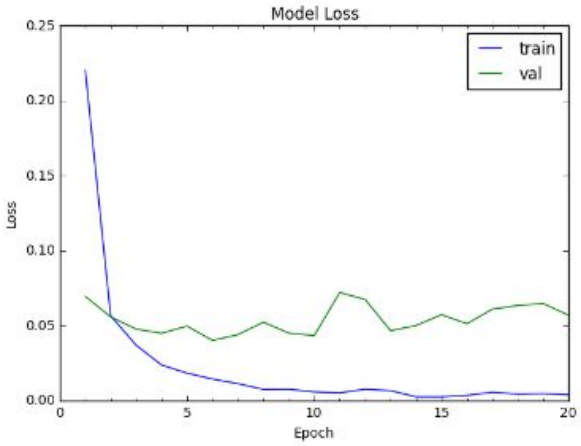





```
Model took 75.61 seconds to train
Accuracy on test data is: 98.68
```

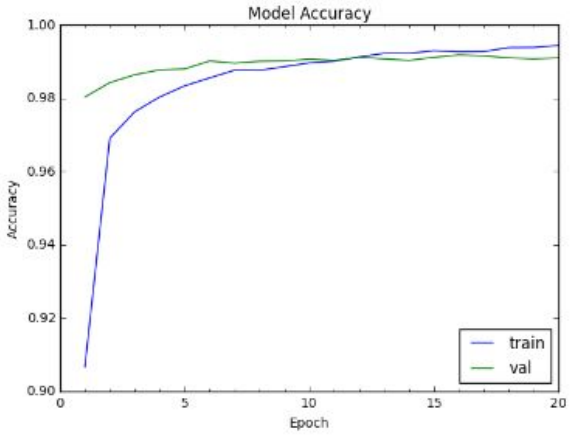
- When we deepened the convolution and neural layer the result was better than before. We just increased one of both layers and we had better output.

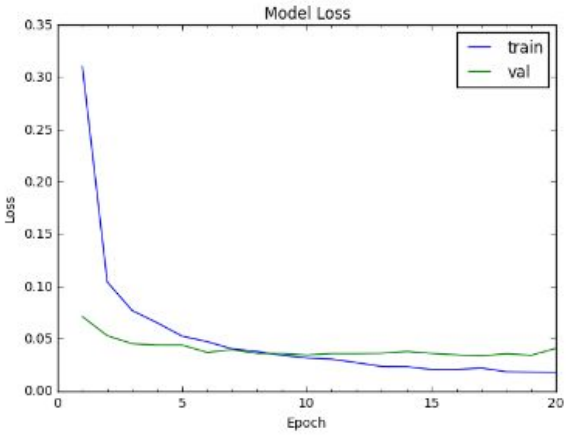





```
Model took 137.12 seconds to train
Accuracy on test data is: 98.78
```

- After performing dropout on the second architecture the result was modified again. We performed dropout after flattening and after the output of first neural layer and the result was surprising.






```
Model took 141.79 seconds to train
Accuracy on test data is: 99.21
```

CONCLUSION

The conclusion we drew is that as we deepen the CNN the output accuracy increases. Also dropout influences our CNN in a good way to achieve more accuracy.