**Convolutional Neural Network Modeling to Classify Handwritten digits using MNIST data**

Sri Charan Bodduna

Email: [sricharan0246@gmail.com](mailto:sricharan0246@gmail.com)

Table of Contents

[Part 1: One Convolution Layer 3](#_Toc159798616)

[Block diagram 3](#_Toc159798617)

[Results 4](#_Toc159798618)

[Part 2: One convolution layer with 3000 steps 5](#_Toc159798619)

[Code 5](#_Toc159798620)

[Results 5](#_Toc159798621)

[Part 3: Two Convolution layer with 5000 steps 6](#_Toc159798622)

[Block diagram 6](#_Toc159798623)

[Results 6](#_Toc159798624)

## Part 1: One Convolution Layer

### Block diagram

A diagram of a network

Description automatically generated

A screenshot of a computer

Description automatically generated

Image showing step size as 5001 to consider 5000th step I have taken 5001 as the step count.

### Results

A screenshot of a computer

Description automatically generated

In this section, the have redesigned the previous model to have only 1 convolution layer. The output of the CONV layer is applied to flattening layer then I have connected it to fully connected layer with SoftMax function and dropout function.

After training the model with 5000 steps, the model has resulted the test accuracy as **0.986.**

## Part 2: One convolution layer with 3000 steps

### Code

**A screenshot of a computer

Description automatically generated**

Image showing step size as 3001 to consider 3000th step I have taken 5001 as the step count.

### Results

A screenshot of a computer

Description automatically generated

After re-training the previous model with 3000 steps, the above results shows that accuracy is **0.9798** at 3000th step.

## Part 3: Two Convolution layer with 5000 steps

### Block diagram

A screenshot of a computer

Description automatically generated

### Results

A screenshot of a computer

Description automatically generated

In this section, I have created 2 layered convolutions, ReLU with 2 levels of pooling. I have used max pooling in this model. Later the output of 2nd layer pooling is flattened and provided to fully connected layer with SoftMax activation function. I have also implemented dropout with 0.5 probability. Later, I collected the output at the respective node from 0 to 9.

I have trained this model with 5000 steps. Finally evaluated the model on test dataset. The model has provided the test accuracy as 0.9884 (98.84%). Which is good for prediction.