

Traffic Sign Classifier Report

I did the following to classify Traffic Signs.

Init

1. Load the Training set using pickle.
2. Split the Validation Set from the training set.
3. Visualize the dataset.
4. Shuffle the dataset
5. Setup the tuning parameters, epoch and batch size.

(Keep playing with it, until you find the optimal one). In this case: EPOCH=100 and Batch Size=256.

Solution Architecture

Input

The architecture accepts a $32 \times 32 \times C$ image as input, where C is the number of color channels. The german traffic dataset has colors and therefore the number of color channels is 3.

Architecture

Layer 1: Convolutional. The output shape should be $28 \times 28 \times 6$.

Activation. RELU Activation function.

Pooling. The output shape should be $14 \times 14 \times 6$.

Layer 2: Convolutional. The output shape should be $10 \times 10 \times 16$.

Activation. RELU Activation function.

Pooling. The output shape should be $5 \times 5 \times 16$.

Flatten. Flatten the output shape of the final pooling layer.

Layer 3: Fully Connected. This should have 120 outputs.

Activation. RELU Activation function

Layer 4: Fully Connected. This should have 84 outputs.

Activation. RELU Activation function

Dropout. Dropout to avoid overfitting. Try using dropouts in different layers.

Layer 5: Fully Connected (Logits). This should have 10 outputs.

Output

Return the result of the 2nd fully connected layer.

Learning

GPU helps in getting good performance and tuning different parameters along with dropouts at different layers, helps get good performance.