UDACITY

Logout

< Return to Classroom

DISCUSS ON STUDENT HUB

Traffic Sign Classification

REVIEW | CODE REVIEW | HISTORY

Meets Specifications

Congrats on passing this first deep learning project!

It was good to see how you preprocessed the training data and trained a suitable CNN to come up with a successful model for this task.

For further challenge, try making the performance of your model even better (it's possible to achieve > 99% accuracy on the test set!) by making modifications to your network architecture (eg. having a deeper network with more layers, employing batch normalization, etc.) – have fun!

Files Submitted

The project submission includes all required files.

- **Ipython notebook with code**
- **HTML output of the code**
- **A writeup report (either pdf or markdown)**

Dataset Exploration

The submission includes a basic summary of the data set.

✓ The data set statistics summary was printed out using the proposed methods.

The submission includes an exploratory visualization on the dataset.

✓Well done on your data exploration!

You've provided a basic summary of the dataset as well as an exploratory visualization of the dataset using sample images and by plotting the distribution of images across the different traffic signs.

To further explore the data, you can also display a few images for each of a few traffic sign classes to get a sense of how they vary within the class.

Design and Test a Model Architecture

The submission describes the preprocessing techniques used and why these techniques were chosen.

✓ Well done using grayscaling and normalization to preprocess the data.

Rate this review

The submission provides details of the characteristics and qualities of the architecture, including the type of model used, the number of layers, and the size of each layer. Visualizations emphasizing particular qualities of the architecture are encouraged.

✓ Sufficient details about the employed architecture were provided.

The visualization of the architecture graph can be enlightening. TensorBoard is a very useful tool for that.

The submission describes how the model was trained by discussing what optimizer was used, batch size, number of epochs and values for hyperparameters.

The submission describes the approach to finding a solution. Accuracy on the validation set is 0.93 or greater.

✓ The architecture approach was discussed. Nice job!

Comments
Things to consider at the time of designing a CNN:

- Data preprocessing
- Number and type of layers
- Selection of the adequate optimizer
- Tunning of hyperparameters
- Model training
- Model's assessment metric/benchmark

The plot of the learning curve helps to see if there is a considerable gap between the training and validation accuracy. If there is one, the model may be overfitting or underfitting. More information about the learning curve here.
The following regularization techniques can be used to minimize overfitting to training data(in addition to dropout and cross-validation that you already use):

- L2 Regularization. Using lambda = 0.0001 which seemed to perform best. The important point here is that L2 loss should only include weights of the fully connected layers, and normally it doesn't include bias term. The intuition behind it is that bias term is not contributing to overfitting, as it is not adding any new degree of freedom to a model. Here is An Overview of Regularization Techniques in Deep Learning.
- Early stopping.You can apply stopping with the patience of 100 epochs to capture the last best-performing weights and roll back when the model starts overfitting training data. A validation set cross entropy loss as an early stopping metric, the intuition behind using it instead of accuracy is that if your model is confident about its predictions it should generalize better. Please check this article to know How to Stop Training Deep Neural Networks At the Right Time Using Early Stopping.
- Batch Normalization is not a regularization technique, but it helps neural network weights to convert faster and hence works well with early stopping.

Test a Model on New Images

The submission includes five new German Traffic signs found on the web, and the images are visualized. Discussion is made as to particular qualities of the images or traffic signs in the images that are of interest, such as whether they would be difficult for the model to classify.

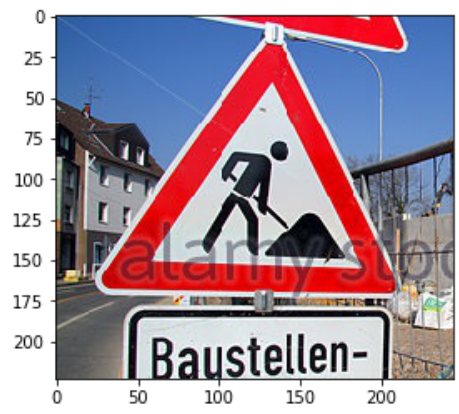
The submission documents the performance of the model when tested on the captured images. The performance on the new images is compared to the accuracy results of the test set.

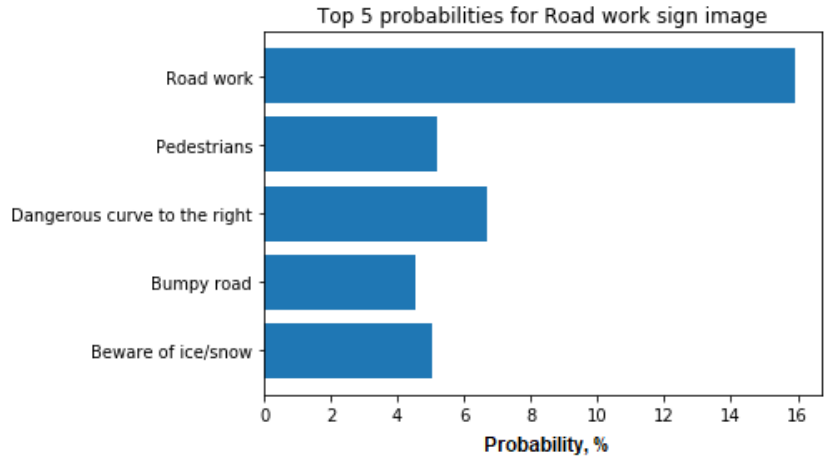
✓ Nice work testing your model on new images and discussing its performance here.

The top five softmax probabilities of the predictions on the captured images are outputted. The submission discusses how certain or uncertain the model is of its predictions.

✓ Good job outputting the top-5 softmax probabilities for new images. Consider visualizing them as a bar chart for better readability:

Rate this review





DOWNLOAD PROJECT

RETURN TO PATH

Rate this review