UDACITY

Logout

PROJECT

Use Deep Learning to Clone Driving Behavior

A part of the Self Driving Car Engineer Nanodegree Program

PROJECT REVIEW | CODE REVIEW 1 | NOTES

Meets Specifications

SHARE YOUR ACCOMPLISHMENT

Dear Udacian,
Well done with the precise amendments you made in this submission after the previous review. This was an exceptional submission! 😄 Please, Keep up the good work.

Quality of Code

The model provided can be used to successfully operate the simulation.

Good job here, the model you provided runs smoothly in the simulation

The code in `model.py` uses a Python generator, if needed, to generate data for training rather than storing the training data in memory. The `model.py` code is clearly organized and comments are included where needed.

Well done in producing a clearly organized code and inlcuding specific comments. I see you decided to use a generator in order to prepare some data for your model

Model Architecture and Training Strategy

The neural network uses convolution layers with appropriate filter sizes. Layers exist to introduce nonlinearity into the model. The data is normalized in the model.

Well done, you have correctly implemented the use of convolutional layers accompanied with the appropriate filter sizes.

Train/validation/test splits have been used, and the model uses dropout layers or other methods to reduce overfitting.

Good job here in using train/validation/test splits in order to get a training, and validation set and also for your implementation of dropout in order to reduce overfitting

Learning rate parameters are chosen with explanation, or an Adam optimizer is used.

Good job here in using Adam optimizer. The method is straightforward to implement, is computationally efficient, has little memory requirements

Training data has been chosen to induce the desired behavior in the simulation (i.e. keeping the car on the track).

Excellent job here in setting up your data to try to induce the best desired behavior possible

Architecture and Training Documentation

The README thoroughly discusses the approach taken for deriving and designing a model architecture fit for solving the given problem.

Excellent work here on providing a good and reasonable explanation of your approach on deriving and designing your model architecture.

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

Well done. You clearly outlined in your answer:

- The characteristics of the architecture.
- The type of model used.
- The number of layers in the model.
- The size of each layer.

The README describes how the model was trained and what the characteristics of the dataset are. Information such as how the dataset was generated and examples of images from the dataset should be included.

Good job here on providing sufficient details on how your model was trained

Simulation

No tire may leave the drivable portion of the track surface. The car may not pop up onto ledges or roll over any surfaces that would otherwise be considered unsafe (if humans were in the vehicle).

Good job here, the simulation works considerably well, no tire leaves the drivable portion of the track.

DOWNLOAD PROJECT

1 CODE REVIEW COMMENTS ›

Have a question about your review? Email us at review-support@udacity.com and include the link to this review.

RETURN TO PATH

Student FAQ