Bosch Global Software Technologies

MS/ EXV-XC

Applied CV Coding Assignment

Version: 1.1.2

Timing: 1 week from when you receive this assignment

This interview is aimed at testing your programming and analytical skills required to work effectively in an end to end data science project.

From understanding and analyzing a dataset building a model pipeline and deploying in a container evaluating and explaining said pipeline visualizing the performance both quantitatively and qualitatively.

A GitHub repository where you add all these code elements is expected along with documentation. Details on what must go in the repository is covered in each individual tasks.

Ensure you are able to deploy the **data** task alone in a container. Please document on how to run and use the container for the data task and for the model and evaluation task, please give clear description on how to use them (Preferably in the GitHub's markdown README)

1. **Data Analysis (10 points):** The first part of the assignment is to analyze the BDD dataset for the task of Object detection. This means you can only focus on the 10 detection classes with bounding box such as light, signs, person, car. Download the dataset from here: BDD Dataset . For Object detection, you would need to download 100k Images (5.3GB) and Labels (107 MB). Drivable areas and lane marking or any other semantic segmentation data are not required. The data analysis can include (but not limited to):

- checking the distribution of the training samples for object detection under various classes. (write the parser and data structure to handle the images and json for analysis)
- analyzing the train and val split. (test set is not required for analysis)
- Based on the analysis, see if you can identify any anomalies or patterns in each of the object detection classes.
- visualizing the stats of the dataset in a dashboard.
- Identifying and visualizing interesting/unique samples in different classes.

Ensure whatever analysis on data that you have done, please add the details in a document in the GitHub repository. Also, please add all the code that you had used to analyze the data in the repository. Following coding standards such as PEP8 is crucial. For functions, classes having appropriate docstring is recommended. Feel free to use libraries such as black to format your code, pylint to check for PEP8. There are no restrictions on what library you use. The data analysis code must be part of a container which we can take it and test it out in interviewer's system without any additional installations (Docker must be self contained).

2. **Model (5 [+ 5] points):** In this stage, please choose a model of your own choice (using pre-trained model zoo trained on BDD dataset is allowed or training the data on your own is also accepted). While the freedom to choose the pre-trained model is yours, the reasoning for it must be sound. This includes why the chosen model and you should be able to explain model architecture. Please document these in the repository. Code snippets/working notebooks must be in the repository. **Additional points:** While it is understandable that training the model from scratch might be too time consuming, we would like to see if you could build the loader to load the dataset into a model and even train for 1 epoch for a subset of the data by building the training pipeline. Having a code snippet for this would help you gain additional points.

3. **Evaluation and Visualization (10 points):** Evaluate your model on the validation dataset and document the quantitative performance. Please document your personal analysis on what works and what does not work for the model and why. Connect the evaluation with visualization. For visualization, both quantitative and qualitative analysis are required on the validation dataset. Metrics for quantitative visualization is to be decided by the candidate and explained in the documentation as to why those were chosen. In terms of qualitative performance, identify tools that can show ground truth and predictions visualized on the data. With that see if you can cluster performance on where the model fails. Use the above steps of evaluation and visualization to help suggest improvements to the model or through data. Connect the data analysis inputs here to help identify patterns in the model's performance.

Best wishes