Investigation of Near-accident Car-driving Scenario using Deep Imitation Learning and Reinforcement Learning

Team Members: Wentao Zhong, Jiaqiao Zhang Advisor: Erdem Bıyık

Motivation

The autonomous driving technology grows rapidly in recent years. However, the high-risk scenario, where a potential accident is likely to happen, could not be tackled well, since the action need to be changed correspondingly with other drivers. Hence, the agent may need to change actions significantly to stay safe.

In this project, we first take the approach of implementing deep Imitation Learning (**IL**) [3] to obtain the driver agent model using data generated by predefined dominant control law. Then, Reinforcement Learning (**RL**) [4] is applied to find the policy in high risk scenarios via switching control model considering both efficiency and safety.

The result shows that the reinforcement learning achieves a good balance in choosing modes, and the more driving modes, the better RL result will be in terms of efficiency.

Data

The dataset we used for the low-level IL training is obtained from the CARLO simulation for two different driving scenarios [1]. For the first cross traffic scenario as shown in Figure 1, we collect the expert demonstrations from three different driving modes (timid, aggressive, normal) in simulation. The control policies for the ego car in different driving modes are pre-defined based on time to collision (**TTC**), time to entering thee intersection of both ego and ado cars according to [2]. The dataset contains observations (including location of the ego vehicle, velocity of the ego vehicle, location of other vehicle with noise and velocity of other vehicle with noise), action (throttle) from the control policies and one indicator which indicates the specific driving mode. The control policies for scenario 2 are defined similarly as in scenario 1. Except for TTC, we calculate a predicted collision point based on ado car's heading, location and velocity.

The vehicle is assumed to be a point mass in the simulation. Roughly 85% data is used for training examples and 10% data is used for validation examples and 5% data is used for test examples for training procedure.

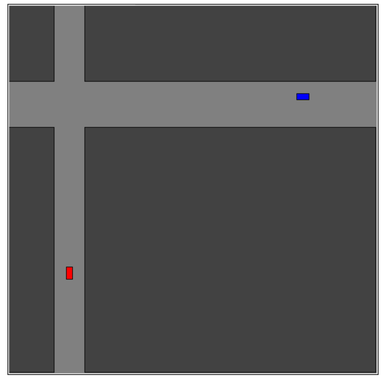

Figure 1. Scenario1: Cross Traffic

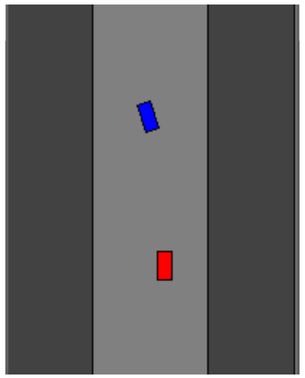

Figure 2. Scenario 2: Wrong Direction

Method

The main method is firstly trying to learn different driving mode using Conditional Imitation Learning (**CoIL**) from the dataset described before. CoIL takes the approach that all observations are fed into the first layer, while the outputs of the first layer are fed into different branches according to user-defined goals, which is shown in figure 5.[6] We defined three main driving modes according to [2], timid mode, normal mode and aggressive mode. These three modes are defined by evaluating the driving risk which is based on time for the following car to reach the preceding car and predicted time to collision. The timid driving mode have low driving risk and it tends to keep low velocity and large gap with other car. The aggressive driving mode has high driving risk and tends to keep high velocity and small gap with other cars. The normal driving mode behaves neutrally between these two modes.

Then the high-level agent which gives the commands to switch the different low-level agent obtained from CoIL before is trained using Proximal Policy Optimization (PPO) RL algorithm [5]. PPO alternate between sampling data through interaction with the environment and optimizing a novel objective function. The novel objective achieves a way to do a Trust Region update which is compatible with Stochastic Gradient Descent and simplifies the algorithm by removing the KL penalty and need to make adaptive updates. The environment used to train RL is CARLO for two different scenarios.

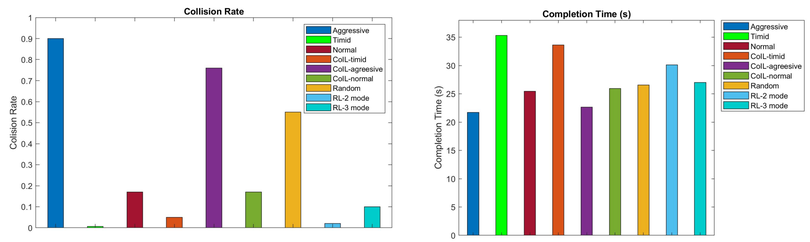

Figure 3. Cross Traffic Scenario experiments results.

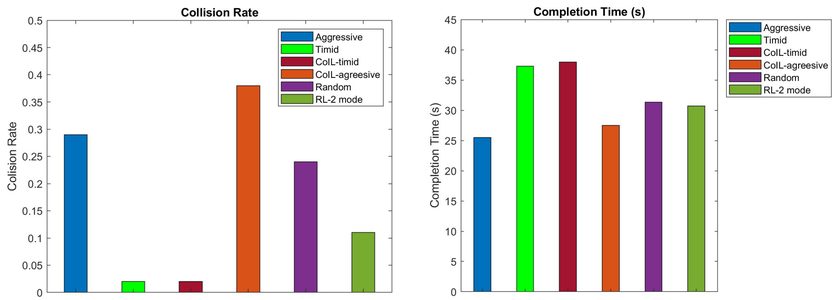

Figure 4. Wrong Scenario experiments results.

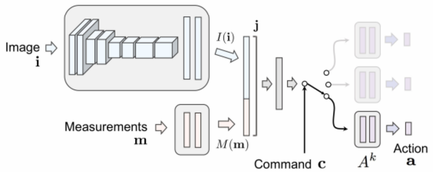

Figure 5. Network architecture for CoIL.

Results and Discussion

The results of scenario 1 is shown in Figure 3. A random policy is included for later comparison with results from Reinforcement Learning. The collision rate and completion time of random policy are in the middle of three hard code policies. Results of RL-2 mode and RL-3 mode shows that the reinforcement learning achieves a good balance in choosing mode. Both RL results has a lower completion time and lower collision rate than the random one. And the collision rate of RL policy is almost as low as timid, while the completion time of RL policy is much lower than the timid one. Also, compared to RL-2 mode, RL-3 mode has a lower completion time but a higher collision rate.

For wrong direction scenario, we only train RL policy in two driving modes. The completion time of RL-2 mode is about the same as the random policy, but the collision rate of RL-2 mode is much lower than random one (from 0.11 to 0.24).

From Figure 3, the overall trend of the collision rate is on the opposite direction of completion time. The results of CoIL is close enough to the original policy, which shows that the low-level driving mode is well trained. For RL, both RL policy (2 modes and 3 modes) achieve good result compared to random policy (lower collision rate and faster completion time). This shows that RL policies can choose different driving modes depending on different observations balancing both efficiently and safety. In addition, from the results of training RL from 2 driving mode and 3 driving mode, it is interesting to find that RL-3 mode has a faster completion time and a bit higher collision rate.

Conclusions

In summary, we can draw a conclusion that the approach - first using conditional imitation learning to learn driving model from expert, then training a high-level policy using reinforcement learning does a great job in two high-risk scenarios. Although the scenarios shown in this project are rather simple and there are some assumptions in the simulation, this does shed light on the application of CoIL-RL in more complicated scenario where the motion planning of the vehicle is challenging due to the environment.

Future Work

There are some work remain to be done. First, more high-risk scenarios including Halting car, Merge, Unprotected Turn can be used to evaluate the performance of different driving model and switching techniques. Second the simulation can be done in CARLA[7] where the physical model is more realistic. Finally, the expert data can be obtained from real drivers instead of hard coded policy.

Contact

<Wentao Zhong>
<Stanford University>
Email: zhong133@Stanford.edu

<Jiaqiao Zhang>
<Stanford University>
Email:qiao1997@stanford.edu

<Erdem Bıyık>
<Stanford University>
Email: ebiyik@stanford.edu

References

1. Zhangjie Cao, Erdem Biyik, Woodrow Z. Wang, Allan Raventos, Adrien Gaidon, Guy Rosman, and Dorsa Sadigh. Reinforcement learning based control of imitative policies for near-accident driving. Science and Systems (RSS), July 2020.
2. Qingwen Xue, Ke Wang, Jian Lu, and Yujie Liu. Rapid driving style recognition in car-following using machine learning and vehicle trajectory data. Journal of Advanced Transportation, 2019:1–11, 01 2019.
3. Markus Wulfmeier, Dushyant Rao, Dominic Zeng Wang, Peter Ondruska, and Ingmar Posner. Large-scale cost function learning for path planning using deep inverse reinforcement learning. The International Journal of Robotics Research, 36(10):1073–1087, 2017.
4. Dean A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In Proceedings of the 1st International Conference on Neural Information Processing Systems, NIPS'88, page 305–313,Cambridge, MA, USA, 1988. MIT Press.
5. John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. CoRR, abs/1707.06347, 2017.
6. F. Codevilla, M. Müller, A. López, V. Koltun and A. Dosovitskiy, "End-to-End Driving Via Conditional Imitation Learning," 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, 2018, pp. 4693-4700, doi: 10.1109/ICRA.2018.8460487.
7. Dosovitskiy, Alexey & Ros, German & Codevilla, Felipe & Lopez, Antonio & Koltun, Vladlen. (2017). CARLA: An Open Urban Driving Simulator.