Stable Multi-Agent Imitation Learning for Driving Simulation

Ruokun He[*]
Shanghai Jiao Tong University
hh15013075@sjtu.edu.cn

Abstract

This work is intended to reproduce the work of PSGAIL on driving simulation and try to make possible progress based on it. As a result, I succeed in getting better result using PSGAIL method then behavioral cloning and GAIL on multi-agent tasks, but not a perfect result as the original paper. I will give explanation of the reproduction result and a description of tricks used in my work. Also a comparison and discussion about model performance between behavioral cloning, GAIL and PS-GAIL will be presented.

1 Introduction

Simulation is critical for an autonomous driving method before it is employed on products. There is high risk in validating autonomous driving algorithm on real-world tests before enough simulation, as dangerous situations may occur and lead to high cost when corner case that not considerd in simulator is met. In this work, the simulator is the **Scalable Multi-Agent RL Training School** platform developped by Huawei Noah's Ark Lab[5].

Imitation learning is a common approach in this task. It considers learning a driving policy by utilizing expert demonstrations, where the assumption is that expert action is the best option in corresponding state. The learning target is teaching the agent to act just like the expert. Behavioral Cloning, a supervised method of imitation learning, is easy to implement. But due to the problem of covariate shift, its performance is not satisfactory when travelling distance increases. Inverse Reinforcement Learning(IRL) approaches solves this problem by formulating the imitation learning task as a Markov Decision Process with an agent policy and reward function learned from interaction with the environment. The learnable reward function enables the model to excavate deeper pattern of expert behaviour, improving model's learning capability.

Nowadays, imitation learning approach has been well employed on many single-agent tasks. However, for multi-agent case, there isn't as much methods. Existing single-agent approach can't be directly transplanted to multi-agent task, as transitioning from single-agent to multi-agent context will cause new covariate shift for the algorithm.

Parameter Sharing GAIL(PS-GAIL)[1] is a method based on Generative Adversarial Imitation Learning(GAIL)[2]. With some optimization for multi-agent context, PS-GAIL achieve better performance then GAIL. In this work, PS-GAIL will be reproduced. The reproduction result will be presented by comparing PS-GAIL, GAIL, Behavioral Cloning performance on different group-size multi-agent tasks.

Furthermore, considering that the training of PS-GAIL is hard and unstable, I will introduce a trick to improve the stability of PS-GAIL training. Better stability plays an important role in adjusting other factor, like super parameters, feature design, of the model.

[*] github address of the project is: https://github.com/RUOKUNH/gail-auto-driving with a brief demo on it

2 Methodology

2.1 Problem Formulation

The multi-agent imitation learning task is formulated as a Markov Decision Process problem. All agents have same observation and action spaces:

$$O_i = O_j \ and \ A_i = A_j \ \forall \ agents \ i, j$$

All agents share the same policy model and reward function:

$$\pi_i = \pi_j \ and \ R_i = R_j \ \forall \ agents \ i, j$$

2.2 Generative Adversarial Model

Backbone of PS-GAIL model is similar to GAIL, composed of a discriminator and generator. Discriminator works as a critic to score expert and agent, where the training target is to distinguish between expert and generative data. Therefore, its can take the place of reward function in MDP model. Agent uses reward generated by discriminator to do optimization. Besides, in the original paper, TRPO[3] is used as optimizing method, while in this work PPO[4] is used as a replacement.

A. Discriminator

Discriminator is intended to tell between expert and generative data, that is, for expert observation-action pairs, it gives higher score while agent observation-action pairs get lower scores. Assume discriminator D is parametrized by ϕ and I consider definition of discriminator like GAIL in this word, where D_ϕ gives the possibility of observation-action pair and the training objective is:

$$\max_{\phi} \mathbb{E}_{\pi_E}[\log D_\phi(s,a)] + \mathbb{E}_{\pi_\theta}[\log(1 - D_\phi(s,a))] \tag{1}$$

Where the sense is minimizing binary cross entropy of expert possibility with one, plus generative possibility with zero.

B. Agent

Agent uses actor-critic backbone to train on reward from discriminator, the formation is:

$$r = -\log(1 - D_\phi(s,a)) \tag{2}$$

Agent is optimized by PPO-clip steps, the total optimizing target is

$$\min_{\theta} \max_{\phi} \mathbb{E}_{\pi_E}[\log D_\phi(s,a)] + \mathbb{E}_{\pi_\theta}[\log(1 - D_\phi(s,a))] \tag{3}$$

C. Balance penalty

During experiment, I find achieving expected PS-GAIL training performance is not easy. Always the training curve will fall all of a sudden while the it is all well before. By analyzing in detail, the critic often reaches extremum when collapse happens, that is, discriminator give possibility approaching one for expert, and possibility approaching zero for agent. As such situation appears too early, it means the discriminator has met overfitting on some feature of observation and fallen into a local minimum. This problem can be alleviated by carefully adjusting learning rate of critic and policy network optimizer, but the adapting process is time-consuming. To increase efficiency, I introduce a trick called **balance penalty** to constraint action of discriminator:

$$p = \gamma \mathbb{E}[(D(s_E, a_E) - D(s_\theta, a_\theta))^2] \tag{4}$$

Intuitively, when GAN model converges, the discriminator output on expert and generative data should be similar, where balance penalty also vanishes to zero, not affecting the final performance of the model. And in practice, balance penalty is a more effective approach on making training stable then carefully adjusting learning rate for me. The new objective target of discriminator then is:

$$\max_{\phi} \mathbb{E}_{\pi_E}[\log D_\phi(s,a)] + \mathbb{E}_{\pi_\theta}[\log(1 - D_\phi(s,a))] - p \tag{5}$$

The experiment results show, a suitable range of γ is $1 \sim 3$.

2.3 Feature Design

An accurately designed feature is essential to model performance. We can observe information including lane, point cloud, vehicle position, speed and others. The observation information is in different formation and not all of them are equally importance. Therefore, we need to design a feature which can reflect high-quality information as much as possible while involving little redundant and inaccurate information. Here the feature is composed of:

Ego Vehicle Info

feature that reflect information about ego vehicle state, including: **[position, speed, vehicle size]**

Road Info

Intuitionally, we think road information like lane line edge is indispensable for driving. Here I consider **[lane relative heading, offset from lane centerline, distance to road edge at two side]**

Nearby Vehicle Info

Besides ego state, nearby vehicles also need to be considered. Furthermore, we actually pay more attention to relative information instead of absolute information when driving. So here I consider **[relative speed, relative position]** of the closest vehicle at eight directions around ego car.

Algorithm 1 PS-GAIL

```
Require: Expert demonstrations, Policy parameter $\theta$, Discriminator parameter $\phi$, PPO update paramters
for $i = 1, 2, \dots$ do
    Rollout trajectories for all agents $\tau_i \sim \pi_{\theta i}$
    Score $\tau$ with discriminator, generating reward $r_i$ through Eq.(2)
    Take a PPO step to improve agent on reward $r_i$ and get $\pi_{\theta i+1}$
    Update discriminator parameter $\phi$ by maximizing Eq.(5)
end for
```

3 Related Work

The simulator used in this work is SMARTS, a open-source platform for scalable multi-agent learning and simulation of realistic driving interaction. It can be utilized to evaluate performance of MARL algorithms in AD context and explore new algorithms.

Behavioral Cloning is a basic algorithm in imitation learning. The main idea is maximizing probability of expert action under given observation, which is achieved by supervised training approach on expert demonstration.

Generative Adversarial Imitation Learning(GAIL) is an influential generative method in imitation learning. By using IRL method to train a critic to represent the unknown reward function, with combination of GAN model, it works well in autonomous driving tasks like car following and highway-driving. By imitating driving pattern of expert with tutorials of a learned critic, it outperforms behavioral cloning on many tasks.

PS-GAIL is an extension of GAIL to multi-agent context. By setting parameter sharing policy model base on GAIL, PS-GAIL alleviates the covariate shifting problem when transitioning GAIL from single-agent tasks to multi-agent tasks. PS-GAIL works better than GAIL especially when groupsize of agents increases.

4 Experiments

Behavioral Cloning, GAIL and PS-GAIL model are used to learn policies in highway driving context and their performance will be compared. For all experimental result, I choose the best model in 500 training epochs and test it on the given benchmark evaluation set. Besides, the utility of balance penalty will also be presented by showing training curve before and after adding penalty on several super parameters. Due to experiment time limitation, each training is terminated in 250 epochs.

4.1 Evaluation Metrics

For each algorithm, the similarity of rollout trajectories between expert demonstrations is evaluated by **Frechet Distance**. Furthermore, to explicitly compare performance of each algorithm, **Collision Rate, Success Rate and Average Travelled Distance** will also be evaluated.

4.2 Results and Discussion

Fig.1 shows the evaluating results of Behavioral Cloning, GAIL, PS-GAIL on 1, 5, 10, 20, 50, 100 agents control task.

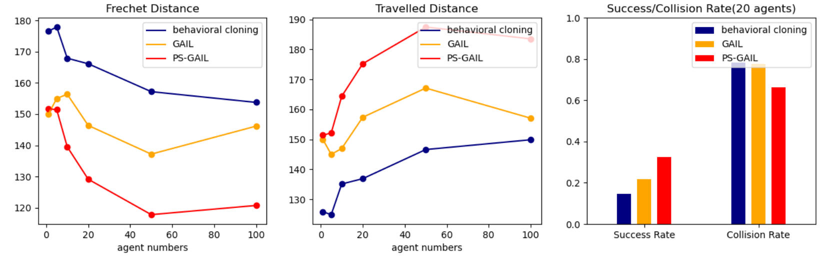


Figure 1: Evaluate Results

As the fig shows, both GAIL and PS-GAIL method outperforms Behavioral Cloning on single and multi agent tasks. When agent number is small(1, 5), GAIL has similar or higher performance compared with PS-GAIL. But as agent number increases, GAIL grow less competitive in multi-agent context while PS-GAIL still works well.

Fig.2 shows comparison of training curve before and after adding penalty into discriminator loss. The first two column shows the comparison of travelled distance and critic scores alongside timestep with and without balance penalty, under different learning rate. And the last column shows training process with different penalty coefficient.

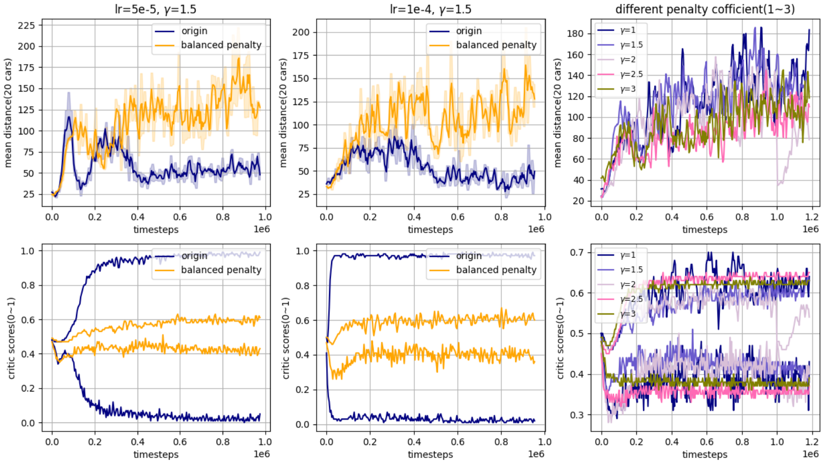


Figure 2: Balance penalty performance

We see that under two different learning rate, model with balance penalty all performs more stable then ones without penalty. And under different penalty coefficient, the training process is similar overall while coefficient $\gamma = 1.5$ performs the best therein, which shows that balance penalty do plays a role in stabilizing training process.

5 Conclusion and Discussion

Reproduction of PS-GAIL is not a simple work. At beginning, searching for a proper super parameter is a big problem for me, as GAN model collapse easily under incorrect parameters and it takes long time to finish one training. Another accompanying problem is that the validating process on effectiveness of feature design is slow. To alleviate the problem, the balance penalty is used to stabilize training. Although it hasn't been evaluated whether balance penalty affects the final performance of PS-GAIL as the model is not trained to convergence, but it works in accelerating parameter tuning and feature designing procedure.

As a reproduction result, the PS-GAIL outperforms GAIL and behavioral cloning model on multi-agent context. But the performance of my PS-GAIL is not as good as expected in original paper. This may be caused by the following reasons.

Inappropriate training approach

Carefully selected learning rate and batch size choices may lead to a better model. Furthermore, as Fig.1 shows, the model works worse on small-size(smaller than 10) and large-size(larger than 100) multi-agent task compared with middle-size. The cause may be that only 20-size multi-agent sampling is used during training. Using gradually increasing agent numbers in training may enable model to learn difference between pattern in multi-agent task and single-agent task better.

Time series information not taken into account

In driving activities, we not only consider current state, but also state several timesteps before. By replacing the last MLP layer in net structure into GRU to utilize time series information may increase model performance.

Acknowledgments

Teaching assistant Zhengbang Zhu gave much guidance on code and implementing details of model in last two months. My classmates Zhe Cao and Mingyang Jiang also provides some inspiration and help on code.

References

[1] Raunak P Bhattacharyya, Derek J Phillips, Blake Wulfe, Jeremy Morton, Alex Kuefler, and Mykel J Kochenderfer. Multi-agent imitation learning for driving simulation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1534–1539. IEEE, 2018.

[2] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29:4565–4573, 2016.

[3] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

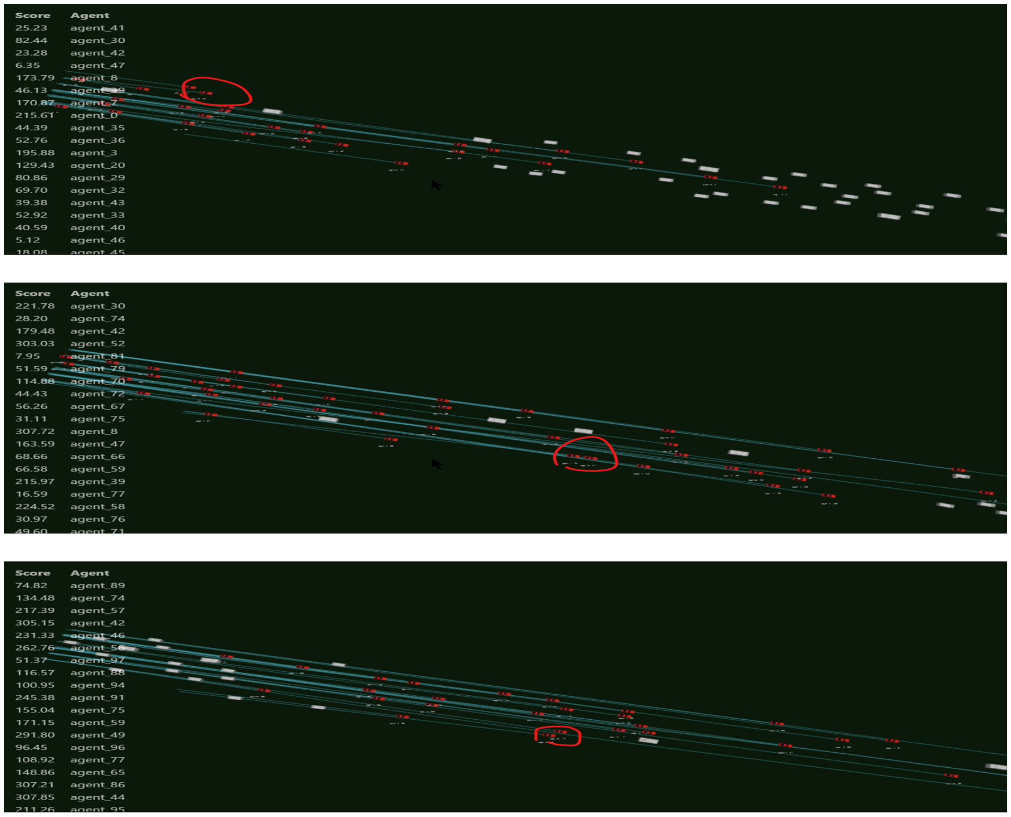
[4] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[5] Ming Zhou, Jun Luo, Julian Villella, Yaodong Yang, David Rusu, Jiayu Miao, Weinan Zhang, Montgomery Alban, Iman Fadakar, Zheng Chen, et al. Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving. *arXiv preprint arXiv:2010.09776*, 2020.

A Appendix

Due to time limit, several improvement schemes haven't been evaluated in practice. Here I will give a detailed analysis of a 100-agent simulation procedure about model problems and give possible improvement schemes.

Fig.3 shows some typical scene where collision happens.



Score	Agent
25.23	agent_41
82.44	agent_30
23.28	agent_42
6.35	agent_47
173.79	agent_8
46.13	agen[illegible]9
170.87	agent_7
215.61	agent_0
44.39	agent_35
52.76	agent_36
195.88	agent_3
129.43	agent_20
80.86	agent_29
69.70	agent_32
39.38	agent_43
52.92	agent_33
40.59	agent_40
5.12	agent_46
18.08	agent_45

Score	Agent
221.78	agent_30
28.20	agent_74
179.48	agent_42
303.03	agent_52
7.95	agent_81
51.59	agent_79
114.88	agent_70
44.43	agent_72
56.26	agent_67
31.11	agent_75
307.72	agent_8
163.59	agent_47
68.66	agent_66
66.58	agent_59
215.97	agent_39
16.59	agent_77
224.52	agent_58
30.97	agent_76
49.60	agent_71

Score	Agent
74.82	agent_89
134.48	agent_74
217.39	agent_57
305.15	agent_42
231.33	agent_46
262.76	agent_56
51.37	agent_97
116.57	agent_88
100.95	agent_94
245.38	agent_91
155.04	agent_75
171.15	agent_59
291.80	agent_49
96.45	agent_96
108.92	agent_77
148.86	agent_65
307.21	agent_86
307.85	agent_44
211.26	agent_95

Figure 3: typical collisions in simulation

Most of the time, agent drives in similar speed in simulation, it means model have learned to control speed according to distance. But sometimes rear-end collision happens when speed control is delayed. I think a possible problem is that, in current feature, ego and nearby vehicle size and vehicle center distance have been considered, but model by haven't learn to calculate the nearest distance of vehicle body, or when there is rotation, the nearest distance can't be figured out. Thus a possible improvement is replacing vehicle size and center distance as nearest distance of vehicle body.

Another problem is that for Vehicles entering the main road from turnouts, the handling maneuver is not always in time. And when the turning delays, collision will happen. I think the main cause is that samples of this type is the minority in dataset, so increasing sampling ratio of such vehicles may work.