Conflicting demonstrations in Inverse Reinforcement Learning

Rafaël Labbé

Supervisors: Luciano Cavalcante Siebert, Angelo Caregnato Neto

EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 25, 2023

Name of the student: Rafaël Labbé
Final project course: CSE3000 Research Project
Thesis committee: Luciano Cavalcante Siebert, Angelo Caregnato Neto, Jana Weber

Abstract

This paper aims to investigate the effect of conflicting demonstrations on Inverse Reinforcement Learning (IRL). IRL is a method to understand the intent of an expert, by only feeding it demonstrations of that expert, which may be a promising approach for areas such as self driving vehicles, where there are a lot of demonstrations from experts. This paper aims to investigate the effect of conflicting demonstrations on IRL. Demonstrations may not always come from the same expert or the expert may prioritize different goals at times. For example, a driver may not always do grocery shopping at the same store or they may take a slightly different route on different occasions. The results showcase a negative effect from severely conflicting demonstrations on the ability of Max Entropy IRL to recover rewards, but do show some slightly optimistic results on more than two goals.

1 Introduction

Reward functions are a vital component of reinforcement learning (RL) algorithms. [Sutton and Barto, 2018]They tell an agent whether an action is good or bad. Although these reward functions can often be manually specified by a human, this is not always the case. For instance, self-driving cars are a case where specifying a reward function is a very complex problem. IRL aims to tackle this problem by learning the reward function from expert demonstrations in an automated fashion.[Ng and Russell, 2000]

Over the past decades, various IRL algorithms have been suggested. [Zhifei and Meng Joo, 2012] [Arora and Doshi, 2021] One problem is that for a given set of expert demonstrations, there are many different possible reward functions that explain those demonstrations. A well known and effective IRL algorithm is Max Entropy IRL (MaxEnt IRL) [Ziebart et al., 2008], which aims to not only match feature expectations [Do and Batzoglou, 2008], but also maximize entropy to minimize the artificial certainty and bias that can otherwise be present.

One factor that can significantly complicate the process of learning the reward function from expert demonstrations is that expert demonstrations may not always involve the exact same intentions. Different experts can have different intentions or the same expert may have different intentions over multiple demonstrations. For instance, when two routes to a destination are of similar length, a human may prefer one route sometimes and seemingly randomly, the other on different occasions. Recently, multiple approaches have been taken to deal with this problem, such as the integration of a the Dirichlet process mixture model into Bayesian IRL[Choi and Kim, 2012]. Other recent noteworthy approaches include a deep adaptive approach [Bighashdel et al., 2021] and clustering based solutions such as given in [Bighashdel et al., 2022].

Although there are multiple approaches to dealing with multi-intention IRL, there is little research on the effectiveness of simpler approaches such as Max-Entropy IRL [Ziebart et al., 2008]. In this paper, the goal is to answer the following primary research question: "To what extent can IRL learn rewards from conflicting demonstrations?"

The report has the following structure: Chapter 2 dives further into the background and addresses some related work. Chapter 3 describes the methodology used in this research, where chapter 4 shows the experiments that are done. Chapter 5 showcases the results of those experiments and chapter 6 includes an analysis of these results and discusses them. Chapter 7 gives a brief overview of the ethical ramifications of the research and the reproducibility of the experiments. Chapter 8 concludes the research and introduces some ideas for future work.

2 Background

In this chapter, the main concepts that this research builds upon are explained and a brief overview of related work is given.

2.1 Inverse Reinforcement Learning

RL is a technique that aims to train agents to perform well in an environment.[Sutton and Barto, 2018] In the process, an agent is rewarded for good actions or punished for bad actions. The goal for the agent is then to maximize its rewards, which naturally leads to more good actions and less bad actions. A reward function determines which actions to reward and which to punish.[Sutton and Barto, 2018]

IRL on the other hand, takes trajectories from experts, agents that are assumed to perform in an optimal way and it then seeks to estimate the reward function that these agents optimize. Afterwards, RL agents can then use the reward function the IRL algorithm produced and learn a policy that achieves a goal that is similar to the original expert. Recovering the underlying reward function is an ill-posed problem [Ziebart et al., 2008], as there exist many different underlying intentions that showcase the same demonstrations. One commonly used IRL algorithm is Max Entropy IRL [Ziebart et al., 2008], which uses the principles of maximizing entropy to reduce such ambiguity and select a reward function that is unbiased.

2.2 Related work

The topic of conflicting demonstrations has been touched upon by a number of papers that discuss techniques to recover multiple intentions. A popular idea is the use of Expectation-Maximization [Do and Batzoglou, 2008] to group demonstrations, one cluster for each intention [Bighashdel et al., 2022]. Another approach includes the use of probabilistic models to estimate multiple underlying reward functions [Choi and Kim, 2012]. Another idea is to use deep learning based methods to estimate multiple underlying reward functions [Bighashdel et al., 2021].

These works all focus on finding a solution to the problem of having demonstrations from agents with different goals, but there is very little attention given to simpler algorithms that were designed for basic, single-intention IRL. This paper aims to shed more light on the performance of Max Entropy IRL on conflicting demonstrations.

3 Methodology

The aim of this chapter is to give a complete overview of the methodology of this research. The methodology can be briefly summarized in the chart below:

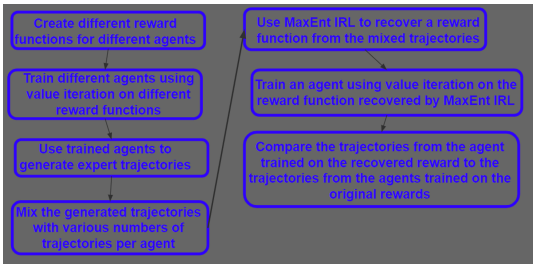

Figure 1: A graph showcasing the methodology flow

The experiments involve the creation of conflicting demonstrations. Mostly due to time constraints, this data is not taken from humans. Instead, the conflicting demonstrations are created by training separate RL agents on different reward functions, which allows each agent to prioritize different goals. Each agent is trained using value iteration [Pashenkova et al., 1997], which works well in relatively simple environments such as those used in this research.

These RL agents are then used to generate trajectories, which are fed into MaxEnt IRL to obtain an output reward function. The IRL algorithm gives an estimate of the reward function, which can be used to train another RL agent that generates its own trajectories. Similarity of the trajectories generated by the RL agent trained on the estimated reward function to the trajectories generated by the RL agents trained on the original reward function can then be used to evaluate the performance of the IRL algorithm on conflicting demonstrations.

The amount of trajectories from each RL agent can be varied to observe potential differences in the way the IRL algorithm can learn rewards from agents that it is fed less trajectories from and agents that it is fed more trajectories from. This approach allows analysis of results that can indicate to what extent demonstrations that optimize a different reward function actually affect the ability of the algorithm to learn rewards.

The primary IRL algorithm used in this experiment is maximum entropy inverse reinforcement learning[Ziebart et al., 2008], due to its simplicity, speed and proven track record in inverse reinforcement learning in simple situations. The main environment used in the research is a GridWorld environment and variations of it. It is a type of environment that can be scaled from very simple to much more complex and allows for simple and understandable reward functions that can be designed to conflict each other.

4 Experiments using conflicting demonstrations

The experiments in this paper include three main scenarios, each of which will be detailed further in this chapter.

4.1 Extremely conflicting scenario

This experiment focuses on a scenario in which each expert prioritizes a very different goal, that conflicts harshly with the other agent. This can be seen in the figure below.

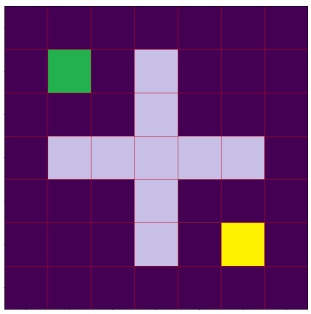

Figure 2: A grid showcasing the starting points in pink and the goals of the 2 different agents in green and yellow.

Expert 1 prioritizes the green goal with a reward 0.8 and the yellow one with a smaller reward of 0.2, whereas expert 2 prioritizes the yellow goal with a reward of 0.8 and the green one with a smaller reward of 0.2. Both agents start from the pink squares. This experiment is mainly meant to measure to what extent the MaxEnt IRL algorithm can recover rewards in extremely conflicting situations.

4.2 Conflicting demonstrations with an easier goal

This experiment focuses on a scenario where there are demonstrations with an easier goal. This can be seen in the figure below, where the yellow goal is much closer to the starting points.

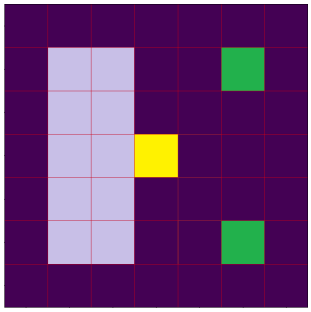

Figure 3: A grid showcasing the starting points in pink and the goals of the 2 different agents in green and yellow.

Expert 1 prioritizes the green goal with a reward 0.4 each and the yellow one with a smaller reward of 0.2, whereas expert 2 prioritizes the yellow goal with a reward of 0.8 and the green ones with a smaller reward of 0.1 each. Both agents start from the pink squares. The purpose of this experiment is to test whether easier goals pose a significant problem in conflicting demonstrations.

4.3 Conflicting demonstrations with 3 agents

This experiment addresses a scenario where there are three agents with conflicting goals. This can be seen in the figure

below, where the yellow, green and orange colors represent three goals for three different agents.

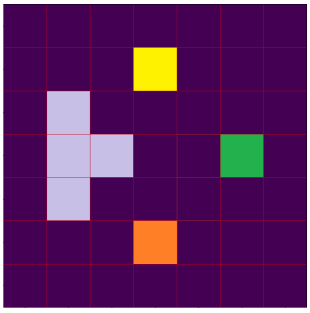

Figure 4: A grid showcasing the starting points in pink and the goals of the 3 different agents in green, pink and yellow.

Expert 1 prioritizes the green goal with a reward 0.8. Expert 2 prioritizes the orange goal with a reward of 0.8 and expert 3 prioritizes the yellow goal with a reward of 0.8. Each agent gets a reward of 0.1 for goals it does not prioritize. They start from the pink squares.

5 Results

This chapter presents the results of the conducted experiments.

5.1 Severely conflicting Experiment

To measure the effect of conflicting demonstrations on the Max Entropy IRL algorithm's ability to learn the underlying reward function, trajectories of an RL agent trained by using value iteration on the reward recovered by the IRL algorithm are compared to the trajectories of the original 2 RL agents. Since trajectories often differ considerably in length, the difference between trajectories is calculated using the Dynamic Time Warping distance measure. The results are shown in 5.

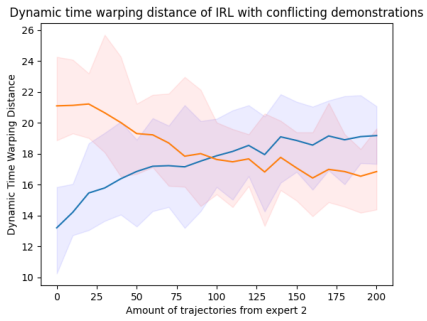

Figure 5: A graph showcasing the effect of severely conflicting demonstrations on the DTW distance

The IRL algorithm is given 100 demonstrations from agent 1 and 0 up to 200 demonstrations from agent 2. The graph showcases in blue the DTW distance from agent 1 trajectories and in orange the DTW distance from agent 2 trajectories.

In the figure above, the blue line shows the probability of the IRL algorithm trajectories going to the goal that expert 1 prioritizes, whereas the orange line shows the probabilities of going to the goal that expert 2 prioritizes.

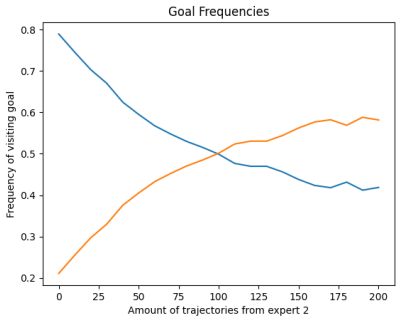

Figure 6: A graph showcasing the effect of severely conflicting demonstrations on the goal the IRL algorithm prioritizes

5.2 Scenario with easier conflicting reward

In this scenario, expert 2 prioritizes a much easier to reach goal. In 7 the blue line showcases the probability of going to the harder goals that expert 1 prioritizes and the orange line shows the probability of going to the easy goal that expert 2 prioritizes.

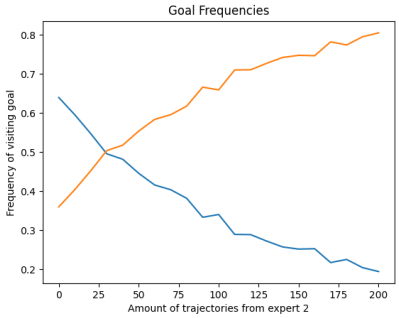

Figure 7: A graph showcasing the effect of an easy conflicting goal on the goal that the IRL algorithm prioritizes

As can be seen above, the orange line overtakes the blue one much earlier in the graph than in 6, since the second goal is easier, so the IRL algorithm focuses on it more.

5.3 Scenario with easier conflicting reward

In this scenario, there are 3 experts. In 7, the blue line showcases the DTW distance of the IRL trajectories from the original trajectories when all conflicting demonstrations come from only expert 2. The orange line shows the DTW distance when an equal number of conflicting demonstrations come from both experts 2 and 3, each one contributing half of the trajectories.

In the graph below, the blue line shows the probability of the IRL trajectories going to the goal of expert 1 when all trajectories come from expert 2 and the orange line shows the probability of the IRL trajectories going to the goal of expert 1 when trajectories come in equal amounts from both expert 2 and expert 3.

Figure 8: A graph showcasing the effect of conflicting demonstrations with different goals in a three agent scenario on DTW distance.

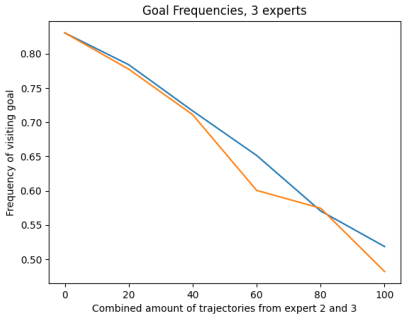

Figure 9: A graph showcasing the effect of conflicting demonstrations with different goals in a three agent scenario on goal prioritization.

As can be seen above in both 8 and 9, whether the conflicting demonstrations come from one expert that prioritizes one goal or from two different experts that prioritize different goals does not seem to make a significant difference.

6 Discussion

Previous researchers on this topic figured it was necessary to use different algorithms, since they did not even provide much analysis of MaxEnt IRL on conflicting demonstrations. The results in this paper appear to confirm that thought, since conflicting demonstrations had a strong negative effect, as can be seen in scenarios one and two.

Scenario three does show something different, namely that having more than two different goals may not necessarily make it harder for MaxEnt IRL to learn rewards. This may be a result of the fact that the experiments here were relatively simple and the goals were not sufficiently different for MaxEnt IRL to have additional problems with recovering the rewards there.

MaxEnt IRL does not cluster goals and this may be a big part of the reason why it performs so poorly on conflicting demonstrations. It does not have a good way of figuring out which goals belong to which agents and that makes it struggle on conflicting demonstrations.

7 Responsible Research

No human data is used in this research, primarily due to time related constraints. The data used is produced by ourselves through the use of RL agents on virtual environments. Therefore, there are no ethical consequences of this that should be considered. Efforts to make the research reproducible include a clear description of the methodology that's used and the experiments that are undertaken, each of which occupies its own section

The code is made available on Github.[1]

8 Conclusions and Future Work

The main goal of this paper was to investigate the effect of conflicting demonstrations on MaxEnt IRL. The results show that there is a strong negative effect of conflicting demonstrations in IRL. An interesting phenomenon found in the results is that it appears to be the case that MaxEnt IRL does not struggle with additional goals beyond two different experts, although this requires more research to confirm. This research also sticks to quite simple scenarios, so future research may look into making them fit to more practical problems and using more modern IRL algorithms.

References

[Arora and Doshi, 2021] Arora, S. and Doshi, P. (2021). A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500.

[Bighashdel et al., 2022] Bighashdel, A., Jancura, P., and Dubbelman, G. (2022). Model-free inverse reinforcement

[1] Github Repository of conflicting demonstrations in IRL.

learning with multi-intention, unlabeled, and overlapping demonstrations. *Machine Learning*.

[Bighashdel et al., 2021] Bighashdel, A., Meletis, P., Jancura, P., and Dubbelman, G. (2021). Deep adaptive multi-intention inverse reinforcement learning. *CoRR*, abs/2107.06692.

[Choi and Kim, 2012] Choi, J. and Kim, K.-E. (2012). Nonparametric bayesian inverse reinforcement learning for multiple reward functions. *Advances in Neural Information Processing Systems*, 1:305–313.

[Do and Batzoglou, 2008] Do, C. B. and Batzoglou, S. (2008). What is the expectation maximization algorithm? *Nature Biotechnology*, 26(8):897–899.

[Ng and Russell, 2000] Ng, A. and Russell, S. (2000). Algorithms for inverse reinforcement learning. *ICML '00 Proceedings of the Seventeenth International Conference on Machine Learning*.

[Pashenkova et al., 1997] Pashenkova, E., Rish, I., and Dechter, R. (1997). Value iteration and policy iteration algorithms for markov decision problem.

[Sutton and Barto, 2018] Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA.

[Zhifei and Meng Joo, 2012] Zhifei, S. and Meng Joo, E. (2012). A survey of inverse reinforcement learning techniques. *International Journal of Intelligent Computing and Cybernetics*, 5(3):293–311.

[Ziebart et al., 2008] Ziebart, B. D., Maas, A., Bagnell, J. A., and Dey, A. K. (2008). Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 3*, AAAI'08, page 1433–1438. AAAI Press.