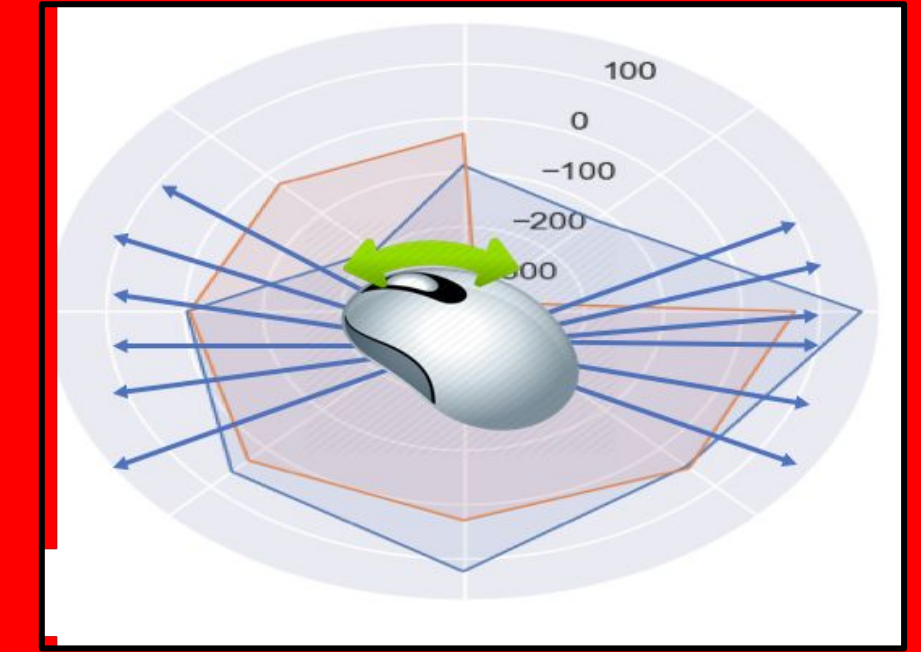# Inverse Reinforcement Learning (IRL) on Mouse Movement Data

Sankalp Chauhan (Sonny), Beiyu Lin

Computer Science Department, The University of Texas Rio Grande Valley, Edinburg, Texas, 78539

https://towardsdatascience.com/mouse-movement-modelling-to-predict-online-fraud-973d90b201e2

Motivation:

- Our goal was to run an inverse reinforcement learning (IRL) algorithm on mouse movement data to predict people's behaviors
- Be able to design webpages with their own personalized content
- Another aspect was to investigate how behavior is different in the general population than in different subpopulations
- It can be contended how characteristics such as age, health condition, and education have on daily routines.
- The results of the experimentation will help to automate diagnoses and predict behavioral features of individuals within a group
- In light of the evidence people in the areas of sociology, psychology, and anthropology will be able to align theories with human behavior.
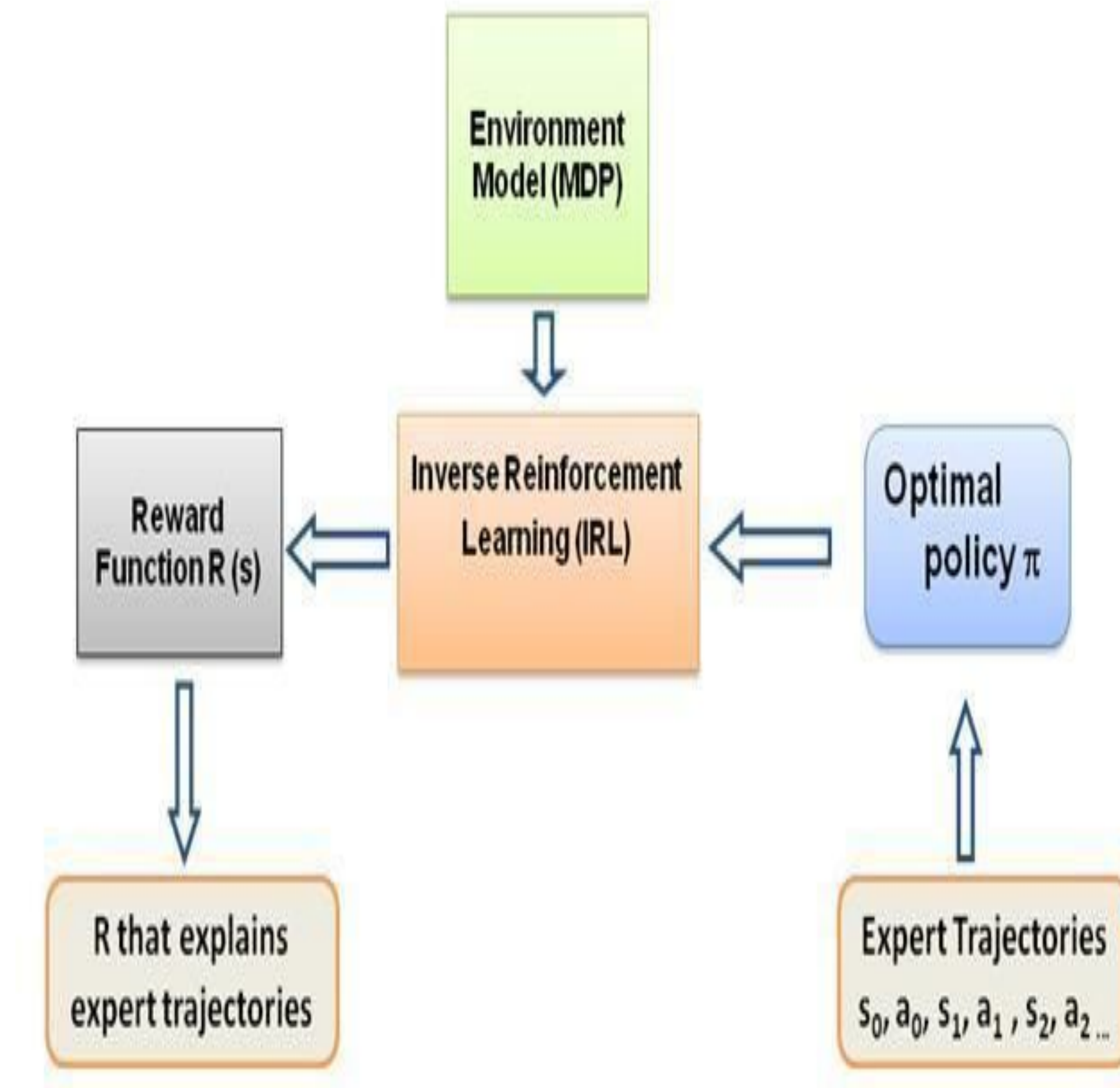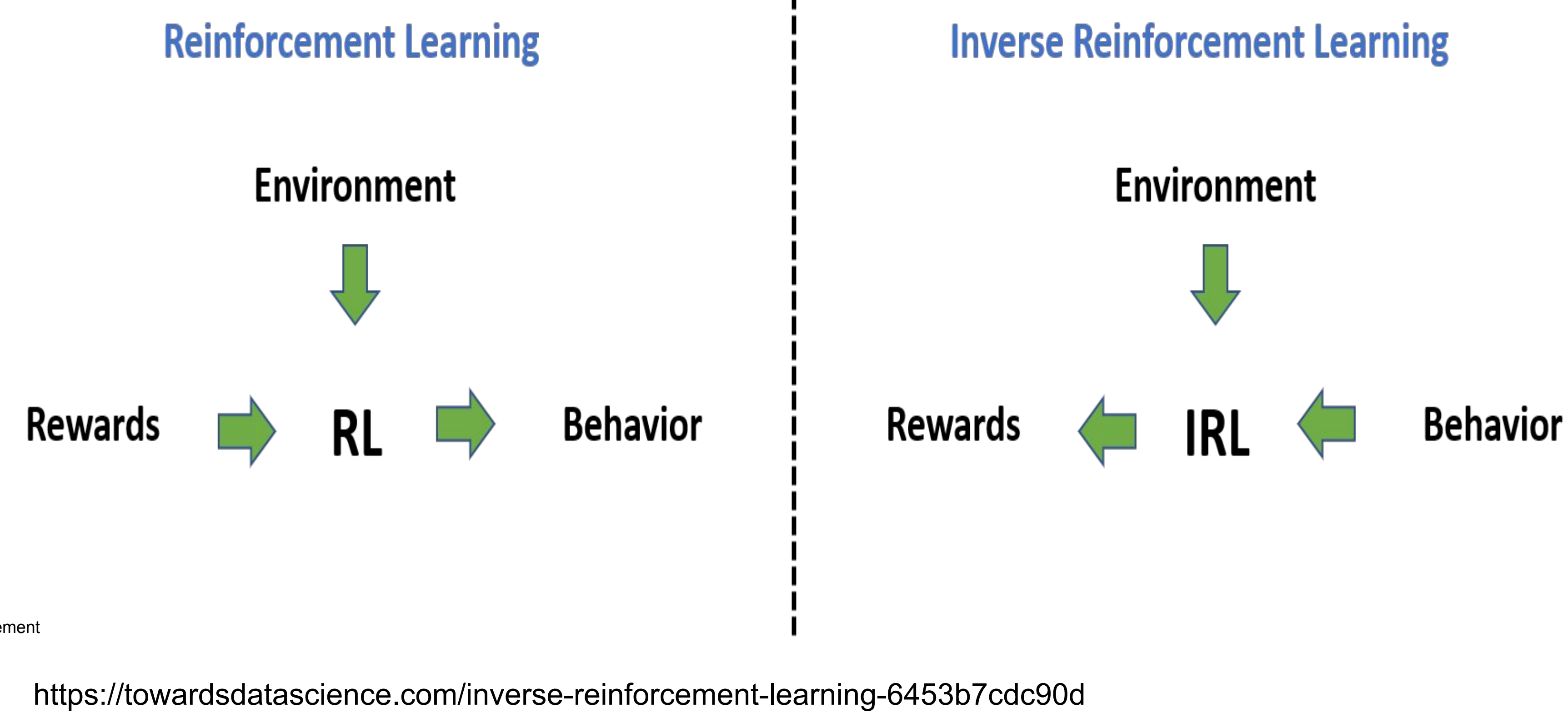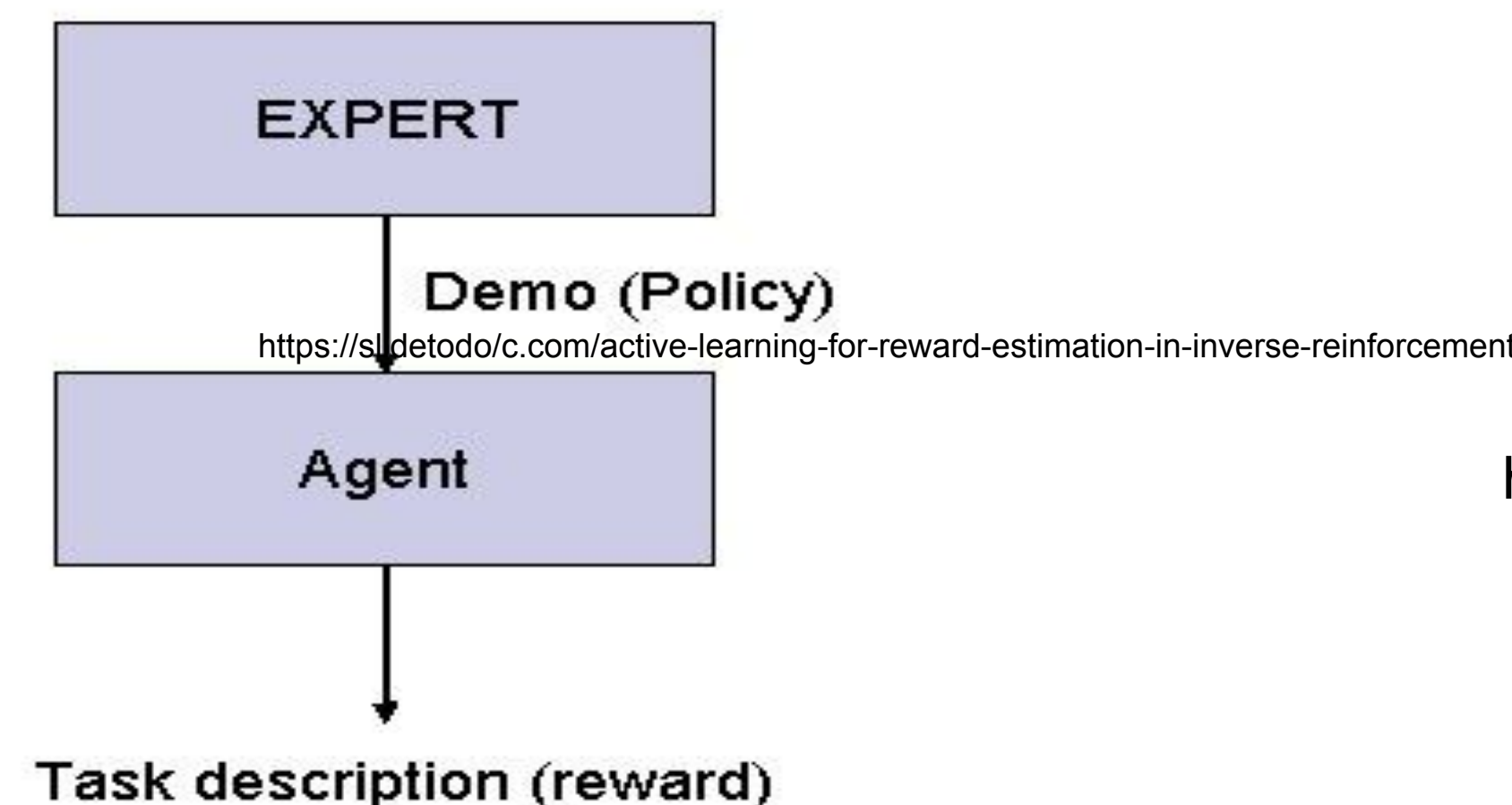
Objective

- Our objective was to use the example of the inverse reinforcement learning (irl) algorithm from C_a_modified_main.py on the excel file.
- We want to use an example of a previously run code on an excel file
- We want to be able to run the inverse reinforcement learning algorithm on the excel file.
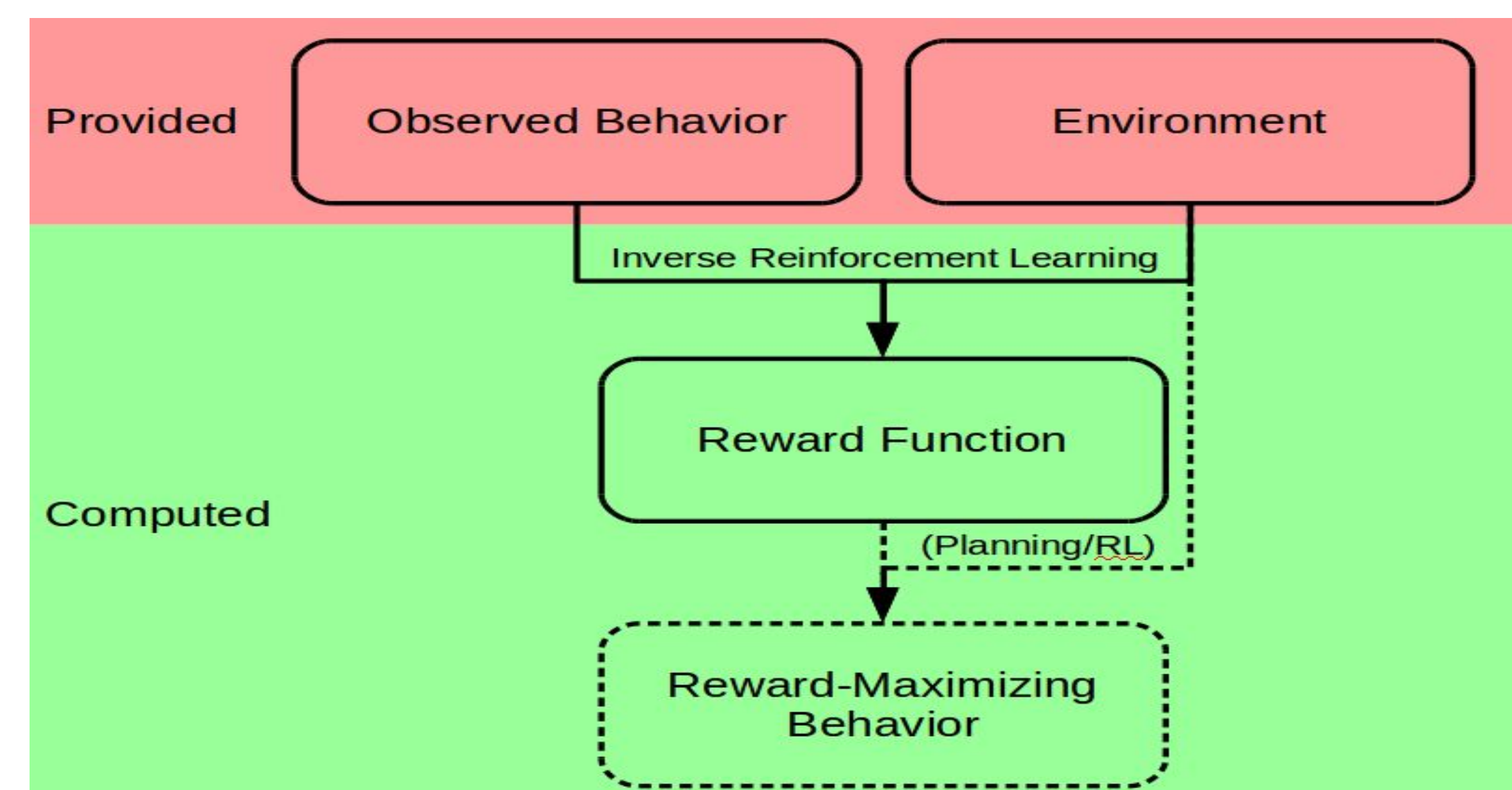
Inverse Reinforcement Learning

- The RL paradigm:

WORLD → Reward → Agent → Policy

- The IRL paradigm:

EXPERT → Demo (Policy) → Agent → Task description (reward)

https://slidetodo.c.com/active-learning-for-reward-estimation-in-inverse-reinforcement

3

What is Inverse Reinforcement Learning?

- Inverse Reinforcement Learning(IRL) is when given the policy, or observed behavior of agent or chatbot, one would want to infer their rewards
- Inverse Reinforcement Learning is framed as an optimization problem where we are trying to optimize:
1. the difference between the optimal action and the next best action
2. minimizing the size of the rewards in the reward function/vector

Dataset implement the algorithm

- **Primary data:** Excel file with several columns that contains the x and y coordinates of the trajectories we need as well as several other important features present.

- **Secondary Data:** C_a_modified_main.py an example of an Inverse Reinforcement Learning (IRL) algorithm that was run on an hh109 dataset. We then were able to run C_a_modified_main.py on the windows terminal. Also there was a readme file present that showed how to read the file on windows terminal

Reinforcement Learning

Environment

Rewards ⇒ RL ⇒ Behavior

Inverse Reinforcement Learning

Environment

Rewards ⇐ IRL ⇐ Behavior

https://towardsdatascience.com/inverse-reinforcement-learning-6453b7cdc90d

Provided: Observed Behavior — Environment

Inverse Reinforcement Learning

Computed: Reward Function → (Planning/RL) → Reward-Maximizing Behavior

https://dkasenberg.github.io/inverse-reinforcement-learning-rescue/

Future work

1. We want to be able to extend the inverse reinforcement learning framework to deep inverse reinforcement learning
2. Being able to utilize deep learning in relative entropy irl

https://www.researchgate.net/figure/Inverse-Reinforcement-learning-Cornell-University-2011_fig3_316786383

Environment Model (MDP) → Inverse Reinforcement Learning (IRL) → Reward Function R (s) → R that explains expert trajectories

Optimal policy π → Expert Trajectories $s_0, a_0, s_1, a_1, s_2, a_2 ...$