CS 440 Project Proposal

Drew Kristensen

Description

The goal of this project is to create a policy learner to optimize the traffic flow in a non-virtual city environment. However, in order to accomplish this task, we will be using model based reinforcement learning approaches to tackle the end goal. By first training the architecture on an easily testable and repeatable enviornment, we can develop a model that will complete this task quicker and easier than one reliant on real world training data. One key feature in this project is keeping the inputs simple enough that they would be readily available in real life applications - that is to say the features used as inputs should be easy to observe.

This project has already been attempted in a small scale, with Wade Genders and Saiedeh Razavi[2] developing a Deep Q network for controlling one traffic intersection in the SUMO program[4], which was shown to be extremly effective. However, the hope for this project is to improve upon the techniques they used, as well as expand the project to include multiple intersections that comprise a city-like environment in the simulation.

Recently, computer vision has shown strengths in reliably detecting vehicles from not only images but real time video [6]. Using these, a stationary camera positioned atop a traffic light would be able to accuratly enumerate the number of vehicles waiting at a traffic light. Therefore, the more difficult part of the implementation of such an AI is not the recognition of vehicles in the real world, but the automated scheduling of the lights at the intersections, hence why we are applying model based reinforcement learning.

This project will be completed using reinforcemenet learning techinques to develop an efficient policy to minimze time spent waiting at lights for cars in a city environment. Since this aproach has had success in other applications such as the board game GO[7], teaching creatures mobility and actions[1][3], or playing ATARI games[5] the hope is that this project can be an example of yet another area that RL can be effective in, while creating a system that is both socially as well as environmentally beneficial.

Functionalities

The finished AI should, in order from least to most ambitious:

- Limit input features to easily observable values
- Implement some form of policy learner using reinforcement learning techniques that have been previously developed
- Minimze the amount of time that cars spend at one light
- Minimize the total amount of time cars spend stopped at all traffic lights across the city
- Be sensitive to perturbations in the traffic light system from outside forces (such as pedestrain crossing requests or emergency vehicles)

Timeline

Timeline for Capstone Project

Jan 15-Jan 21	Read the literature on RL and investigate the traffic simulation environment.
Jan 22-Jan 28	Decide on which type of reinforcement learning algorithm to use for the project, decide on the state space for the project.
Jan 29-Feb 4	Decide on which type of reinforcement learning algorithm to use for the project, decide on the state space for the project. Begin implementation of the RL agent.
Feb 5-Feb 11	Continue implementation of the RL agent.
Feb 12-Feb 18	Finish first implementation and begin training.
Feb 19-Feb 25	Training RL agent and fix any issues that arrise from the process.
Feb 26-Mar 4	Training RL agent and fix any issues that arrise from the process.

Mar 5-Mar 11	Finish training and begin testing the network.
Mar 12-Mar 18	Continue testing on various networks.
Mar 19-Mar 25	Continue testing. If time permits, test on novel environments with different layouts, test on network made from layour of Tacoma streets. Start outline of final write-up.
Mar 26-Apr 1	Continue testing on various networks. Finish outline of write-up and start the first draft.
Apr 2-Apr 8	First Draft of Paper and wrap up testing.
Apr 9-Apr 15	Finish Research and ensure proper documentation.
Apr 16-Apr 22	Start finalizing the paper and start creating the presentation.
Apr 23-Apr 29	Finish both the paper and the presentation.

Materials

Materials that I will use for this project are

- Computer
- TensorFlow framework (or another ML framework if needed)
- SUMO program for vehicular traffic simulation

Challenges

The challenges I see for the project are

- Learning the current approaches and algorithms for model based reinforcement learning
- Finding the best algorithm in model based RL for the task
- Developing a way to train the network on the model quickly and efficiently
- Choosing an algorithm such that adding intersections will not drastically impact the performance without retraining completly

Learning Outcomes

This project will fufill the learning outcomes for this course since it is an investigation and research project into one specific area within Artificial Intelligence. By using reinforcement learning techniques, I will expand upon material that I learned in the Introduction to Artificial Intelligence course that I took here, and specifically look into the area of machine learning known as reinforcement learning, which we only touched on in the class. Furthermore, it is a porject that could be easily applied to the real world for cities to beter handle traffic flow that they recieve, improving both the city and the lives of the drivers within it.

References

[1] Trapit Bansal et al. "Emergent Complexity via Multi-Agent Competition". In: *CoRR* abs/1710.03748 (2017). arXiv: `1710.03748`. URL: `http://arxiv.org/abs/1710.03748`.

[2] Wade Genders and Saiedeh Razavi. "Using a Deep Reinforcement Learning Agent for Traffic Signal Control". In: *CoRR* abs/1611.01142 (2016). arXiv: `1611.01142`. URL: `http://arxiv.org/abs/1611.01142`.

[3] Nicolas Heess et al. "Emergence of Locomotion Behaviours in Rich Environments". In: *CoRR* abs/1707.02286 (2017). arXiv: `1707.02286`. URL: `http://arxiv.org/abs/1707.02286`.

[4] Daniel Krajzewicz et al. "Recent Development and Applications of SUMO - Simulation of Urban MObility". In: *International Journal On Advances in Systems and Measurements* 5.3&4 (Dec. 2012), pp. 128–138.

[5] Volodymyr Mnih et al. "Human-level control through deep reinforcement learning". In: *Nature* 518 (Feb. 2015), 529 EP -.

[6] Joseph Redmon et al. "You Only Look Once: Unified, Real-Time Object Detection". In: *CoRR* abs/1506.02640 (2015). arXiv: 1506.02640. URL: http://arxiv.org/abs/1506.02640.

[7] D. Silver et al. "Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm". In: *ArXiv e-prints* (Dec. 2017). arXiv: 1712.01815 [cs.AI].