Machine Learning Engineer Nanodegree

Capstone Proposal

Daniel Paulus
November 3rd, 2016

Proposal

Domain Background

The project domain will be the analysis and optimization of traffic light control using machine learning. Efficient ways of organizing traffic light control in an urban environment are important for numerous reasons like lowering fuel consumption, grid lock prevention and overall faster transportation times. Additional challenges include prioritizing traffic dynamically for emergency or police cars or rerouting traffic in case of accidents. Nowadays traffic lights are controlled by static algorithms using time slots. A machine learning approach could potentially optimize traffic. Because traffic lights today are autonomous, unmanaged systems a simulation environment to test a machine learning strategy is necessary.

Problem Statement

A machine learning solution for direct traffic light control will be developed. Using structural and statistical information about traffic junctions and their traffic lights, a machine learning approach using reinforcement learning algorithms should be employed to optimize every traffic lights behavior.

The agent based simulation SUMO[1] will be used because real world data is not available. It provides all the quantifiable and statistical data needed to train and evaluate algorithms. Data can be accessed during the simulation using a Python API or after the simulation in the form of csv files. This way the simulations can be reproduced an unlimited number of times.

Datasets and Inputs

The inputs are the data that SUMO provides. [2] Because SUMO is an agent based simulation it provides very detailed statistics which can be accessed during simulations using a Python API. This is ideal for a reinforcement learning algorithm

---

[1] www.sumo.dlr.de
[2] http://www.sumo.dlr.de/userdoc/Simulation/Output.html

that adapts to traffic statistics. In addition there is static structural information in the form xml files, which describe the road layout, connections, intersections and agent spawns.

The scenario used for evaluation will be the TAPASCologne[3] dataset, which represents 24 hours of traffic information in the city of cologne and is freely available for non-commercial projects under the creative commons license.

Solution Statement

The solution model should take into account the different types of traffic lights and junctions. For example a driveway should probably have a different strategy of control than a pedestrian crossing.

For each of the different types of junctions, a reinforcement learner will be trained to behave optimally. This way the traffic lights will only optimize their behavior in a local sense. The idea is that a solution model using locally optimized behavior based on different types of junctions and traffic lights could improve the overall, global traffic quality.

The reason behind this approach is that a possible implementation of global optimization significantly more demanding in terms of computational and hardware capability. For a global solution all traffic lights would have to be in a network with 24/7 availability.

For further work a global model should be built and evaluated against this solution model. Possibly the solution model found in this project could be used as a fallback mechanism if the global traffic controller should fail for some reason.

Benchmark Model

Finally the optimized traffic control will be compared to statistics of the current scheduling of traffic lights in the TAPASCologne scenario. Using the statistics that the simulation provides, simulation runs using the conventional traffic light controls can be compared to the machine learning optimized model.

Evaluation Metrics

The following metrics will be used to evaluate the solution model:

- Duration: average trip duration
- WaitingTime: average time spent standing (involuntarily)
- TimeLoss: average time lost due to driving slower than desired

---

[3] http://sumo.dlr.de/wiki/Data/Scenarios/TAPASCologne

- DepartDelay: average time vehicle departures were delayed due to lack of road space

They are provided by the simulation and are directly related to the overall efficiency of the traffic system.

Project Design

The first step will be to run a clustering algorithm over all available junction types with traffic lights. The feature selection will be based on the structural information in the TAPASCologne scenario's XML files. For dimensionality reduction of the feature space different algorithms like PCA and ICA will be used evaluated by their results. The clustering algorithms of scikit-learn[4] will be used and evaluated. For the evaluation of the different clustering strategies, the following steps will be executed multiple times. Finally they will be evaluated using the benchmark model.

Once the clusters have been identified, a reinforcement learner for each type needs to be implemented using the PyBrain[5] library. This way different algorithms like Q-Learning or SARSA can be trained and evaluated. The learner will be connected to a running simulation using SUMOs TraCI[6] API for direct traffic control. The necessary code will be implemented in Python. The rewards for the algorithms will have to be carefully determined by the proposed evaluation metrics.

Finally, the evaluation metrics of a full run of the TAPASCologne scenario will then be compared to a simulation of TAPASCologne with direct traffic control by the solution model.

---

[4] http://scikit-learn.org/
[5] http://pybrain.org/
[6] http://www.sumo.dlr.de/wiki/TraCI