

DEEP-RL MODELS FOR AUTONOMOUS DRIVING

BHARTENDU THAKUR

SUPERVISOR: DR. SUMITRA S
(ASSOCIATE PROFESSOR)

*Thesis submitted in partial fulfillment of the requirements for the degree of
Master of Technology in Machine Learning & Computing*



DEPT. OF MATHEMATICS, INDIAN INSTITUTE OF SPACE SCIENCE &
TECHNOLOGY, THIRUVANANTHAPURAM

DECLARATION

I declare that this thesis titled DEEP-RL MODELS FOR AUTONOMOUS DRIVING submitted in fulfillment of the degree of MASTER OF TECHNOLOGY is a record of original work carried out by me under the supervision of DR. SUMITRA S, and has not formed the basis for award of any degree, associateship, fellowship or other titles in this or any other Institution or University of higher learning. In keeping with the ethical practice in reporting scientific information, due acknowledgements have been made wherever the findings of others have been cited.

Bhartendu Thakur
Machine Learning & Computing (SC16M051)
Dept. of Mathematics , IIST

Thiruvananthapuram
JUNE, 2018

ACKNOWLEDGEMENTS

I owe this work to my father for his blessings & support throughout my M.Tech project and throughout my life in general. I dedicate this work to my family & also to my friend Sharon Babu for constant motivation to mark advancement in my projects and having believe in my work and supporting me throughout this project.

I make my sincere gratitude to my research supervisor Dr. Sumitra S for assigning this unique project, her guidance helped me during the time of research. I thank my classmates & collaborators especially Mrs. Viji Narayan for their insightful comments, encouragements, which strengthened me to study & deliver concepts and to present it with confidence. Also, I'm grateful to Prajith KP, Dr. Aravind Vaidyanathan for endorsing this work & providing me scope for research in Scramjet Engines.

Abstract

Learning & automation contains essential vitamins of engineering, one such vitamin is Deep Reinforcement Learning (DRL) viz., conjunction of neural networks & reinforcement learning. This project is about applying Deep-RL models to train autonomous vehicles, creating a novel framework for optimal-route planning & conclusive analysis of current approaches to improve level of autonomy in self-driving cars.

The study started with supervised approach to visualize ‘learning in Deep models’ and then DRL models & its variants have been applied to specific problems such as learning to steer, route planning etc. in context of autonomous driving. In this thesis, DQN models for Enduro (Atari, in OpenAI environment), a car racing game and an implementation of actor-critic algorithm, a policy gradient approach to control self driving car (in Unity3D simulator) are presented.

Later, Reinforced Deep Learning (RDL) framework has been developed, which aims to learn weights of a neural network by alternating movements between a RL set-up and the neural network with decreasing explorations and self-rewards. The model is designed to learn with no prior/given outcomes and rewards is dependent on input features & predicted outcomes. As a proof of concept (a) Optimal Route Planning: Finding end-to-end optimal routes from source to destination node based on time, cost & other constraints, (b) Shape Predictor: Learning to predict parameters of various shapes, has been presented with different *structure* of actions & self-rewards.

Contents

Acknowledgements	i
Abstract	ii
List of Tables	v
List of Figures	vi
1 Introduction	1
1.1 Motivation	1
1.2 Thesis Outline	2
1.3 Contributions	4
2 Autonomous Driving: Supervised Approach	5
2.1 Overview	6
2.2 Dataset and Pre-processing	6
2.3 Nvidia Model	7
2.4 SqueezeNet Model	8
2.4.1 Training details & results	9
2.5 Attention Visualization	10
2.6 Conclusions	11
3 Autonomous Driving: Deep Reinforcement Approach	12
3.1 Overview	13
3.2 Q-Learning	13
3.2.1 Limitations of Q-Learning	14
3.2.2 Neural Network as function approximator	15
3.3 Deepmind's DQN	15
3.4 Improvising DQN	17
3.5 Training & Results	19
3.6 Conclusions	20
4 Autonomous Driving with Policy Gradients	22
4.1 Policy Gradients	23
4.2 Actor-Critic Algorithm	23
4.3 Simulator	25
4.4 Training & Results	26

4.5	Conclusions	26
5	Reinforced Deep Learning Framework	27
5.1	Overview	27
5.2	Method	28
5.3	RDL for multi-layer perceptron	30
5.4	Predicting Route Scores with RDL	32
5.5	Discussion	36
6	Conclusions & Future work	39
6.1	Conclusions	39
6.2	Challenges	40
6.3	Future work	40
6.3.1	Improvised DQN	41
6.3.2	Reinforced Bayesian Model	41
	Appendices	45
A	Pseudo-codes	46
A.1	RDL Algorithm	46
A.2	RDL Based: Optimal path planner	47
B	Network Architectures	48
B.1	Deepmind's Model	48
B.2	Fire-Incept Net: Improvised DQN Model	48
B.3	SqueezeNet Model	49
B.4	Skytrain Model	51
B.5	Shape Predictor	52

List of Tables

2.1	Nvidia's model Vs SqueezeNet model	9
3.1	Results: Deepmind's model Vs Improvised model	20
5.1	R^2 score for normalized outcomes	31
B.1	Deepmind's Architecture	48
B.2	Improvised Architecture	49
B.3	SqueezeNet Architecture	50
B.4	Skytrain Model	51
B.5	Shape Predictor architecture	52

List of Figures

1.1	Summary of the thesis	2
2.1	Driving Innovation (left: Nvidia, right: Waymo)	5
2.2	Overview	6
2.3	Pre-Processing	7
2.4	Nvidia CNN architecture (Bojarski et al., 2016)	8
2.5	SqueezeNet Model	8
2.6	Training (blue), Validation Losses (red)	9
2.7	Attention maps: Saliency features are (a) lanes and pedestrian, (b) wall and edges & (c) other cars	10
3.1	Subdivision in RL	13
3.2	Block diagram: Q-learning	14
3.3	Neural net to estimate Q-value	15
3.4	DQN Model	16
3.5	DQN architecture	17
3.6	Leaky-ReLU activation function	18
3.7	Incept module (with $w = 4$)	19
3.8	Reward per Interval (10000 steps)	19
3.9	Results in emulator: (a) & (c) are starting frames, (b) & (d) are ending frames, before and after training respectively	20
4.1	Block Diagram: Actor-critic algorithm (Caspi et al., 2017)	24
4.2	Simulator	25
4.3	Sample view of environment (Min, 2017)	25
4.4	Results: (a) before training, (b) after training	26
5.1	Block Diagram: RDL framework	28
5.2	Multi-layer Perceptron	30
5.3	Overview: Predicting route-scores	32
5.4	Sample data-points for journey form Node N_7 to N_{81}	33
5.5	Various losses while training	34
5.6	Optimal route will take 28 hrs & cost INR 2256	35
5.7	Sub-optimal route will take 24 hrs & cost INR 2429	35
5.8	RDL Predicted Vs Reference Scores	35
5.9	Trends of errors	35
5.10	Learning curves	38

5.11	Shape Detection outputs	38
6.1	Improvised DQN	40
6.2	Reinforced Bayesian Model	41