Encoding the Future: Deep Representations for Traffic using Graph Neural Networks

Background

For deep learning-based approaches to be successful across the board in the field of motion planning for autonomous vehicles, there are a number of challenges that need to be addressed. One of the most crucial ones is the non-trivial problem of representing the surrounding traffic environment in such a way that it can facilitate learning for deep neural networks-based models.

In order to limit the scope of the research project, we focus solely on highway driving, where the local road network topology is both simple and homogeneous, and thus presents a more tractable research challenge than the general autonomous driving case. In this constrained setting, we choose the set of available lane changes as the agent's action space, and we can hypothesize that the set of surrounding traffic participants, together with the current ego state and possibly some top-level navigation intent (e.g. *take the next exit*), make up all the relevant features.

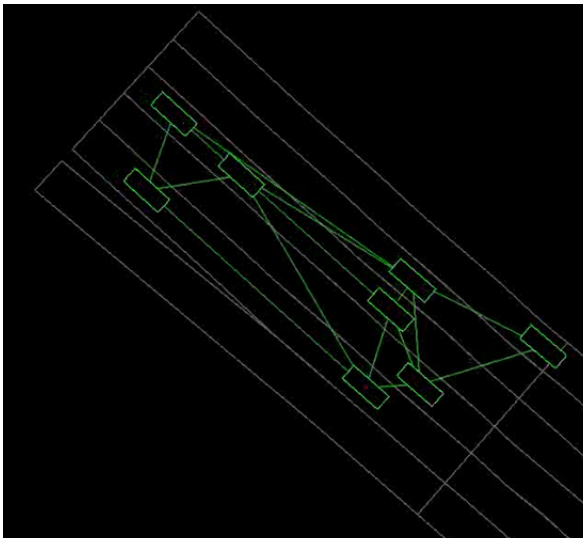

Figure 1: Example of a typical highway traffic scene.

As illustrated by Figure 1, an arbitrary number of other vehicles may participate in the current traffic environment. Accordingly, it is highly desirable that the model can process a variable number of inputs in a joint fashion – while being invariant to the order of processing them. These criteria rule out standard architectures such as MLPs or RNNs.

On the other hand, Graph Neural Networks (GNNs) present a particularly promising modelling approach for tackling this problem. Having gained significant traction in recent years, GNNs have become widely established as the mainstream framework for applying deep learning on graph-structured data [10]. By learning the optimal weights of internal message encoding modules, GNNs work by propagating information as messages along the edges in the graph, before aggregating them at the node level in a multi-layered fashion [9]. Intuitively, given how interaction effects between vehicles can be thought of as propagating across the given traffic scene, graphs are well-suited data structures for our problem domain [7].

However, despite GNNs having shown promising results in the highway driving setting by being directly adopted as the policy network for reinforcement learning-based agents [4, 5], we suggest to add an intermediate modelling layer to our planning pipeline: the *state representation layer*. Motivated by a desire to simplify and accelerate downstream learning tasks, representation learning is concerned with learning abstract state encodings that capture the essential characteristics of the environment that are relevant to the task at hand [2].

Technische Universität München

Fakultät für Informatik

Lehrstuhl für Echtzeitsysteme und Robotik

Supervisor:
Prof. Dr.-Ing. Matthias Althoff

Advisor:
M.Sc. Eivind Meyer

Research project:
KoSi

Type:
MA

Research area:
Autonomous driving
State representation learning
Graph neural networks

Programming language:
Python

Required skills:
Machine learning
Python (preferably with PyTorch experience)

Language:
English

Date of submission:
19. Oktober 2021

For more information please contact us:

E-Mail: eivind.meyer@tum.de

Technische Universität München

Fakultät für Informatik

Lehrstuhl für Echtzeitsysteme und Robotik

A common approach to training such representations is to minimize the reconstruction error via auto-encoder-based architectures [6]. We contemplate that, while attempting to reconstruct the observed traffic scene is clearly a legitimate and sensible approach, the auxiliary capacity to implicitly forecast the future environment will yield even more powerful representations.

Description

In order to formulate this representation learning objective as a tractable learning problem, we first argue that the drivable area [1] is *the* foundational environment characteristic in a planning context. By definition, the drivable area explicitly expresses where the ego vehicle can and cannot drive without colliding. Accordingly, the intermediate representations resulting from predicting this property will be maximally aligned with a motion planning objective. Furthermore, we argue that, given that we intend to focus on high-level planning over relatively long time horizons, defining drivable area within a probabilistic framework is a reasonable approach. Finally, we claim that the most straightforward (while still sufficiently comprehensive) parameterization is attained by predicting lane-centric, single-dimensional occupancy probability as a function of the longitudinal position x and the time value t.

More precisely, this corresponds to approximating the Bernoulli occupancy probability functions $\hat{o}_l(x,t) : \mathbb{R}^+ \times \mathbb{R}^+ \to [0,1]$, where l denotes the lane index, x is the longitudinal lane position starting at $x = 0$, and t is the time offset such that $t = 0$ returns the current lane occupancies. This defines a spatiotemporal, probabilistic occupancy map as illustrated in Figure 2. By learning lane occupancy as a proxy for useful representation, our goal is that the intermediate lane encodings can be extracted and then adopted as input features for a learning-based motion planner.

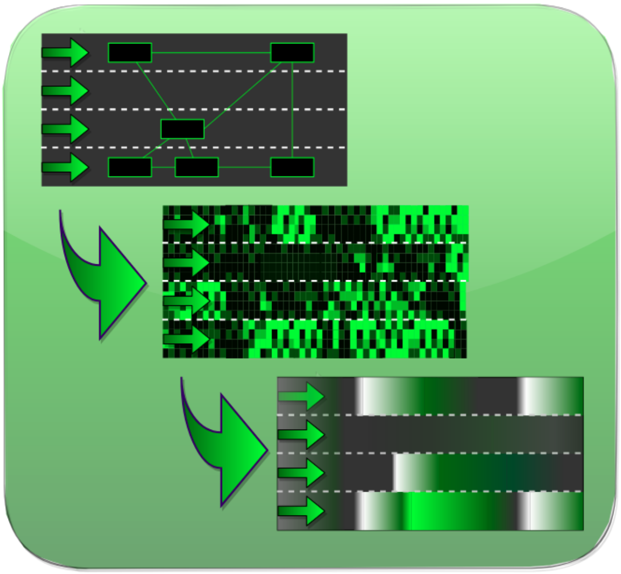

Figure 2: Overview of the representation learning pipeline. We train our architecture on a supervised occupancy prediction problem such that the intermediate fixed-length lane encoding vectors capture the essential characteristics of the driving environment.

Tasks

- An initial literature review for getting familiar with both theoretical and practical aspects of GNNs.
- A dataset containing both graph representations of simulated traffic scenes as well as corresponding multi-step future occupancy grids (discretized both spatially and temporally) has been collected and can be made available for the student at the beginning of the project.
- The primary workload and contribution of this research project will be to develop

a GNN-based predictive state representation framework implemented as a trainable PyTorch module. For effectively designing and training custom-made GNNs, it is recommended to base your solution on the PyTorch extension library PyTorch-Geometric [3].

- For evaluating the representations, a useful comparison would be to benchmark the learned representations against the observation types already available in CommonRoad-RL [8].

Technische Universität München

Fakultät für Informatik

Lehrstuhl für Echtzeitsysteme und Robotik

References

[1] Matthias Althoff and Silvia Magdici. Set-Based Prediction of Traffic Participants on Arbitrary Road Networks. *IEEE Transactions on Intelligent Vehicles*, 1(2):187–202, June 2016.

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation Learning: A Review and New Perspectives. *arXiv:1206.5538 [cs]*, April 2014. arXiv: 1206.5538.

[3] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[4] Patrick Hart and Alois Knoll. Graph Neural Networks and Reinforcement Learning for Behavior Generation in Semantic Environments. *arXiv:2006.12576 [cs]*, June 2020. arXiv: 2006.12576.

[5] Maria Huegle, Gabriel Kalweit, Moritz Werling, and Joschka Boedecker. Dynamic Interaction-Aware Scene Understanding for Reinforcement Learning in Autonomous Driving. *arXiv:1909.13582 [cs, stat]*, September 2019. arXiv: 1909.13582.

[6] Timothée Lesort, Natalia Díaz-Rodríguez, Jean-François Goudou, and David Filliat. State Representation Learning for Control: An Overview. *Neural Networks*, 108:379–392, December 2018. arXiv: 1802.04181.

[7] Sajjad Mozaffari, Omar Y. Al-Jarrah, Mehrdad Dianati, Paul Jennings, and Alexandros Mouzakitis. Deep Learning-based Vehicle Behaviour Prediction For Autonomous Driving Applications: A Review. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–15, 2020. arXiv: 1912.11676.

[8] Xiao Wang, Hanna Krasowski, and Matthias Althoff. CommonRoad-RL: A Congurable Reinforcement Learning Environment for Motion Planning of Autonomous Vehicles. page 7.

[9] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, January 2021. 01835 arXiv: 1901.00596.

[10] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph Neural Networks: A Review of Methods and Applications. *arXiv:1812.08434 [cs, stat]*, April 2021. 01090 arXiv: 1812.08434.