Exploring Decision Transformer for Highway Automated Driving

Luca Forneris[1,2], Francesco Bellotti[1], Riccardo Berta[1], Luca Lazzaroni[1], and Changjae Oh[2]

[1] Department of Electrical, Electronic and Telecommunication Engineering (DITEN)- University of Genoa, Via Opera Pia 11a, 16145 Genova, Italy
[2] Centre for Intelligent Sensing, Queen Mary University of London, United Kingdom
luca.forneris@edu.unige.it
l.forneris@qmul.ac.uk

Abstract. The evolution of Automated Driving Functions (ADFs) is contingent upon the effective implementation of Decision-Making (DM), context perception, and predictive vehicle control. Conventional Deep Reinforcement Learning (DRL) methodologies frequently prove inadequate in dynamic settings, largely due to their inherent limitations in addressing real-time DM and assigning long-term credit. DRL via sequence modeling represents a promising avenue for addressing these challenges by combining the strengths of Attention-based architectures, such Transformer, and DRL. The integration of self-attention mechanisms with offline DRL enables long-term credit assignment, fine-tuning and prevent continuous interaction with the environment, mitigating risks related to real-world simulations and trial-and-error approaches. This paper examines the potential of Decision Transformer (DT) within the AD domain. A DT model was implemented and trained within the highway-env simulation environment. To do so, an offline RL dataset was constructed using a pre-trained Deep Q-Network (DQN) agent. The model was evaluated by comparing its performance against that of the pre-trained DQN and a random agent. Results demonstrated that the DT model exhibited superior DM capabilities, with higher average returns and longer episode durations than DQN. These findings highlight the potential of Transformer-based DRL in AD.

Keywords: Automated Driving, Deep Reinforcement Learning, Decision Transformer, Driving Simulator, Decision-Making, High-speed Maneuvers, Offline Reinforcement Learning.

1 Introduction

The application of Deep Learning (DL) is key in the advancement of Automated Driving Functions (ADFs), with the potential to enhance Decision-Making (DM) [1], context perception [2, 3], and predictive vehicle control [4, 5] tasks. Among them, DM is of paramount importance for the vehicles navigation [6] and maneuvers execution [7, 8].

Traditional DL approaches are not suitable for DM tasks because they lack the ability to adapt to dynamic environments, thus they fall short in handling the real-time DM and optimization required in such tasks. In contrast, Deep Reinforcement Learning (DRL) offers a promising solution by combining the strengths of DL with the trial-and-error mechanism typical of Reinforcement Learning (RL). This hybrid approach allows for continuous interaction with the environment, enabling the system to learn and real-time DM, thereby effectively addressing the challenges associated with DM tasks.

There are two primary types of DRL: online and offline. Online DRL involves continuous interaction with the environment, where the model learns and updates its strategy based on real-time feedback. This approach is highly effective in environments where data can be generated continuously and the system needs to adapt to new information instantly. However, it can be resource-intensive and may not always be feasible, especially in scenarios where real-time data collection is impractical or too costly. On the other hand, offline DRL uses pre-collected datasets to train the model. This method allows the model to learn from a fixed set of experiences, which can be advantageous in controlled environments or situations where gathering real-time data is challenging. Offline DRL provides a more stable and less resource-intensive training process, making it suitable for scenarios where real-time interaction with the environment is not possible or desirable. Also, fine-tuning in offline RL is made possible by continuing to train on new or existing data subsets. Both RL paradigms must tackle two main challenges: learning effective representations of observations (i.e., inputs) and determining how actions (i.e., outputs) influence future returns [9]. Attention-based models, particularly Transformers [10], are inherently capable of addressing long-term credit assignment via self-attention. They have achieved significant success in NLP [11, 12] and CV [13, 14] and have been employed in AD tasks such as object detection and scene representation.

Despite the widespread use of these architectures in perception tasks within AD, their application in DM remains a challenging area of research, despite promising outcomes. In light of the constraints of DRL in AD [9], famously high number of agent-environment interactions and risky data collection, researchers are investigating the potential of Transformer-based architectures to enhance DRL tasks. Research suggests that Transformer-based methods demonstrate superior performance in both offline [15] and online [16] DRL, likely due to their advanced sequence representation learning and enhanced long-term credit assignment capabilities.

In this paper, we explore the application of offline DRL using sequence modeling methods within the AD domain. Specifically, we detail the training of a Decision Transformer (DT) [15] model within the highway-env AD simulator [17] from Farama Foundation Gymnasium [18]. Our evaluation of this approach demonstrates the significant potential of attention-based DRL in advancing the capabilities and effectiveness of AD systems. The results highlight the promise of integrating advanced sequence modeling techniques to improve DM processes in AD.

2 Methodology

In this section, we describe the methodologies used in our research to explore the application of DT within the AD domain. We begin by outlining the creation of an offline RL dataset, followed by the description of the training process of the DT model. Lastly, we present the evaluation metrics and results obtained from comparing our DT model with a traditional Deep Q-Network (DQN) [19] agent.

2.1 Offline Reinforcement Learning Dataset Creation

Simulation tests offer a controlled and safe environment for the evaluation of AD systems, obviating the risks associated with real-world testing. We chose the well-established highway-env [17] (Fig. 1) as our highway DM environment. The compatibility with DRL state-of-the-art libraries makes this framework a valid starting point for prototyping architectures and methodologies.

Offline RL dataset must be composed of Markov Decision Process (MDP) trajectories, (i.e. states, actions, and rewards) to reconstruct RL transitions. State consists in a kinematic observation, a V (i.e. observed vehicles) × F (i.e. features to observe) matrix. Each observation comprises 7 features per vehicle, namely presence, horizontal and vertical coordinates, longitudinal and lateral speed, and the two trigonometric headings. We chose 10 as the number of vehicles to observe over 15 total vehicles for each episode, hence each observation consists in a 10 × 7 matrix.

The action set we used is the default discrete action space provided by the framework, already described by other works that exploit the same simulation environment [6, 20]. Such action space consists in the following actions: right, left, idle, faster, or slower.

Fig. 1. Snapshot from highway-env. The Ego Vehicle (EV), namely the learning agent, is colored in green; Non-Player Vehicles (NPVs) are light-blue.

The rewards we chose to employ are the default highway-env values, and are resumed in Table 1.

Table 1. Employed rewards

Reward	Description	Type	Weight
Collision	Reward assigned whenever the EV crashes with a vehicle	Sparse	-1
Right-most Lane	Reward used to encourage the EV to stay in the right-most lane, if possible	Dense	0.3

High-speed	Reward provided to the agent to prevent it to drive too slow. The speed range we consider when we give the EV such value is [30,36] m/s	Dense	0.3

We then trained a DQN [19] agent on a 3-lane highway environment over 250,000 training steps. Each episode includes a total of 15 vehicles and has a maximum duration of 30 simulated seconds. After training the model, we collected 150,000 episodes, for a total of 3.86M trajectories. Statistics of the dataset are listed in Table 2.

Table 2. Dataset statistics

Min return	Max return	Avg return	Trajectories	Episodes	Min duration	Avg duration
7.6	23.2	11.32	3.86M	150K	10	23.06

2.2 The Decision Transformer Architecture

In accordance with prior research [15], we employ return-conditioned upside-down RL via a cross-entropy loss to predict subsequent actions in an autoregressive manner. We feed DT with the final context length (CL) timesteps, for action, state, and return-to-go values respectively. CL refers to the number of preceding timesteps that the model considers when making a prediction. In essence, it defines the number of past observations and actions that the model incorporates to predict the subsequent action. As depicted in Fig. 2, each value is embedded into specific tokens. Then, an embedding for each timestep is obtained and incorporated into each token. This differs from the standard positional embedding employed in Transformers [10], as one timestep corresponds to three tokens. Tokens are finally processed by a GPT-2 model [21], which predicts future action tokens via autoregressive modelling.

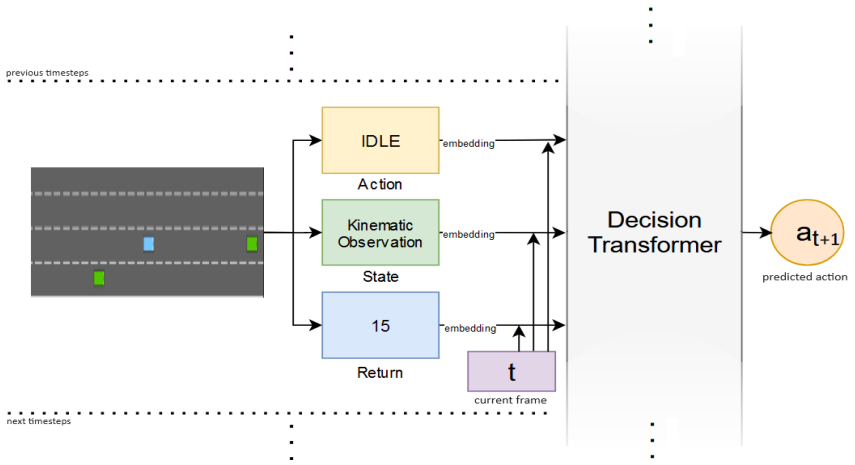

Fig. 2. Diagram describing the original DT architecture and how we used within highway-env. States (S), Returns (R), and Actions (A) are embedded taken from our offline RL dataset, and positional episodic timestep encoding is added. The obtained tokens are then fed into a GPT architecture which uses causal self-attention masking to predict actions.

2.3 Decision Transformer Training

We trained the DT model for 500 epochs. We chose the DT hyperparameters to use according to Bhargava et al. [22]. The values that differ from this set are the used attention heads (i.e. 4), steps per iteration (i.e. 10,000), and batch size (64). The training was performed on a computer equipped with an Intel i9-14900K CPU, 32 GB of RAM, and an NVIDIA GeForce RTX 4080.

2.4 Evaluation and Results

To assess our work, we compared the trained DT model with the DQN model we employed to collect data and with an agent that performs actions randomly selecting them from the action space. Table 3 illustrates the obtained results. We set the target return to the highest return value in the dataset.

Evaluation took place in the same multi-lane environment that we relied on for training DQN and DT models.

Table 4. Obtained results after 300 episodes of evaluation for each agent. Best metrics in bold.

Agent	Type	Avg return	Avg episode length	Avg speed (m/s)	Collision rate
DQN	**Online**	**11.35**	**22.25**	**21.95**	**40%**
DT	Offline	**14.01**	**22.66**	22.56	51%
Random	Heuristic	6.41	13.03	**24.02**	96%

Our DT model reaches a higher return compared to the DQN agent which, on the other hand, manages to follow a safer policy, leading to a 40% collision rate. The higher average episode length value implies that the DT agent was able to navigate the environment for a longer duration before terminating, which may be indicative of better DM capabilities. Our agent was also able to drive faster while still managing to perform better in terms of return and episode length. The random heuristic agent had the highest average speed of 24.02 m/s (86 km/h), but this came at the cost of a very high collision rate, which is understandable given the total random behavior of such agent.

3 Conclusion and Future Work

In this work, we tackled the application of DRL via sequence modeling within the AD domain. We selected highway-env as our AD environment, which, despite its high level of abstraction, serves as an effective starting point for evaluating ML models, assessing AD scenarios [23], and investigating impact of various driving factors [24]. The results

obtained are promising, demonstrating how sequence modeling can be effectively applied for DM in AD. We recognize that the DT collision rate is not yet optimal. However, the performance of the DQN and DT models is inherently interrelated. It is believed that with a more comprehensive and higher-quality dataset, along with a pretrained model exhibiting a higher success rate, the DT performances can be significantly enhanced. Improved data collection efforts will provide the necessary foundation for more effective training and results. While the DQN, a well-established DRL algorithm in highway-env, resulted in fewer collisions than our model, it yielded a lower average return, leading to more conservative driving behavior in terms of speed and collisions. We acknowledge that replacing low-level actions with more realistic, structured, higher-level actions could enhance the realism and safety of the model [6, 24, 25].

To address more realistic situations, it is crucial to employ more sophisticated driving simulators, such as the state-of-the-art CarLA [26] simulator, which takes into account vehicle physics, 3D models, and many other factors that highway-env and other driving simulators do not focus on.

Furthermore, we stress the importance of the data quality, as it significantly impacts the performance and reliability of ML and DRL models. The quality and diversity of the data ensure optimal learning and generalization capabilities for AD scenarios. In light of the fact that fine-tuning to learn new tasks is applicable in offline DRL, opposite to online RL, it is imperative to consider the prevention of catastrophic forgetting when dealing with previously unencountered tasks. Otherwise, the model will forget the previously learned task. Recent advancements in the literature suggest that Multi-Domain DT may be employed to learn more effective representations of trajectories, while avoiding catastrophic forgetting [27].

References

1. Wang, S., Jia, D., Weng, X.: Deep Reinforcement Learning for Autonomous Driving. ArXiv. (2018).
2. Devi, S., Malarvezhi, P., Dayana, R., Vadivukkarasi, K.: A Comprehensive Survey on Autonomous Driving Cars: A Perspective View. Wireless Pers Commun. 114, 2121–2133 (2020). https://doi.org/10.1007/s11277-020-07468-y.
3. Gianoglio, C., Rizik, A., Tavanti, E., Caviglia, D.D., Randazzo, A.: On the Edge Recurrent Neural Network Approach for Ground Moving FMCW Radar Target Classification. IEEE Trans. Consumer Electron. 70, 522–534 (2024). https://doi.org/10.1109/TCE.2023.3343460.
4. Pighetti, A., Bellotti, F., Oh, C., Lazzaroni, L., Forneris, L., Fresta, M., Berta, R.: Investigating Adversarial Policy Learning for Robust Agents in Automated Driving Highway Simulations. In: Bellotti, F., Grammatikakis, M.D., Mansour, A., Ruo Roch, M., Seepold, R., Solanas, A., and Berta, R. (eds.) Applications in Electronics Pervading Industry, Environment and Society. pp. 124–129. Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-48121-5_18.
5. Lazzaroni, L., Bellotti, F., Capello, A., Cossu, M., De Gloria, A., Berta, R.: Deep Reinforcement Learning for Automated Car Parking. In: Berta, R. and De Gloria,

A. (eds.) Applications in Electronics Pervading Industry, Environment and Society. pp. 125–130. Springer Nature Switzerland, Cham (2023). https://doi.org/10.1007/978-3-031-30333-3_16.
6. Capello, A., Forneris, L., Pighetti, A., Bellotti, F., Lazzaroni, L., Cossu, M., De Gloria, A., Berta, R.: Investigating High-Level Decision Making for Automated Driving. In: Berta, R. and De Gloria, A. (eds.) Applications in Electronics Pervading Industry, Environment and Society. pp. 307–311. Springer Nature Switzerland, Cham (2023). https://doi.org/10.1007/978-3-031-30333-3_41.
7. Berta, R., Lazzaroni, L., Capello, A., Cossu, M., Forneris, L., Pighetti, A., Bellotti, F.: Development of deep-learning-based autonomous agents for low-speed maneuvering in Unity. Journal of Intelligent and Connected Vehicles. (in press). https://doi.org/10.26599/JICV.2023.9210039.
8. Lazzaroni, L., Pighetti, A., Bellotti, F., Capello, A., Cossu, M., Berta, R.: Automated Parking in CARLA: A Deep Reinforcement Learning-Based Approach. In: Bellotti, F., Grammatikakis, M.D., Mansour, A., Ruo Roch, M., Seepold, R., Solanas, A., and Berta, R. (eds.) Applications in Electronics Pervading Industry, Environment and Society. pp. 352–357. Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-48121-5_50.
9. Meulemans, A., Schug, S., Kobayashi, S., daw, nathaniel, Wayne, G.: Would I have gotten that reward? Long-term credit assignment by counterfactual contribution analysis. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.) Advances in Neural Information Processing Systems. pp. 68685–68735. Curran Associates, Inc. (2023).
10. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. ukasz, Polosukhin, I.: Attention is All you Need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.) Advances in Neural Information Processing Systems. Curran Associates, Inc. (2017).
11. Lazzaroni, L., Bellotti, F., Berta, R.: An embedded end-to-end voice assistant. Engineering Applications of Artificial Intelligence. 136, 108998 (2024). https://doi.org/10.1016/j.engappai.2024.108998.
12. Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., Wang, X., Zhai, X., Kipf, T., Houlsby, N.: Simple Open-Vocabulary Object Detection. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., and Hassner, T. (eds.) Computer Vision – ECCV 2022. pp. 728–755. Springer Nature Switzerland, Cham (2022). https://doi.org/10.1007/978-3-031-20080-9_42.
13. Mineo, R., Sorrenti, A., Proietto Salanitri, F.: FeDETR: A Federated Approach for Stenosis Detection in Coronary Angiography. In: Foresti, G.L., Fusiello, A., and Hancock, E. (eds.) Image Analysis and Processing - ICIAP 2023 Workshops. pp. 189–200. Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-51026-7_17.
14. Tian, L., Oh, C., Cavallaro, A.: Test-time adaptation for 6D pose tracking. Pattern Recognition. 152, 110390 (2024). https://doi.org/10.1016/j.patcog.2024.110390.
15. Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., Mordatch, I.: Decision transformer: Reinforcement learning via sequence

modeling. Advances in neural information processing systems. 34, 15084–15097 (2021).
16. Parisotto, E., Song, F., Rae, J., Pascanu, R., Gulcehre, C., Jayakumar, S., Jaderberg, M., Kaufman, R.L., Clark, A., Noury, S., Botvinick, M., Heess, N., Hadsell, R.: Stabilizing Transformers for Reinforcement Learning. In: Proceedings of the 37th International Conference on Machine Learning. pp. 7487–7498. PMLR (2020).
17. Leurent, E.: An Environment for Autonomous Driving Decision-Making, https://github.com/eleurent/highway-env, (2018).
18. Gymnasium/CITATION.cff at main · Farama-Foundation/Gymnasium, https://github.com/Farama-Foundation/Gymnasium/blob/main/CITATION.cff, last accessed 2024/05/31.
19. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing Atari with Deep Reinforcement Learning, http://arxiv.org/abs/1312.5602, (2013). https://doi.org/10.48550/arXiv.1312.5602.
20. Bellotti, F., Lazzaroni, L., Capello, A., Cossu, M., De Gloria, A., Berta, R.: Explaining a Deep Reinforcement Learning (DRL)-Based Automated Driving Agent in Highway Simulations. IEEE Access. 11, 28522–28550 (2023). https://doi.org/10.1109/ACCESS.2023.3259544.
21. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language Models are Unsupervised Multitask Learners.
22. Bhargava, P., Chitnis, R., Geramifard, A., Sodhani, S., Zhang, A.: When should we prefer Decision Transformers for Offline Reinforcement Learning?, http://arxiv.org/abs/2305.14550, (2024). https://doi.org/10.48550/arXiv.2305.14550.
23. Mahajan, N., Zhang, Q.: Intent-Aware Autonomous Driving: A Case Study on Highway Merging Scenarios, http://arxiv.org/abs/2309.13206, (2023).
24. Forneris, L., Pighetti, A., Lazzaroni, L., Bellotti, F., Capello, A., Cossu, M., Berta, R.: Implementing Deep Reinforcement Learning (DRL)-based Driving Styles for Non-Player Vehicles. IJSG. 10, 153–170 (2023). https://doi.org/10.17083/ijsg.v10i4.638.
25. Pighetti, A., Forneris, L., Lazzaroni, L., Bellotti, F., Capello, A., Cossu, M., De Gloria, A., Berta, R.: High-Level Decision-Making Non-player Vehicles. In: Kiili, K., Antti, K., de Rosa, F., Dindar, M., Kickmeier-Rust, M., and Bellotti, F. (eds.) Games and Learning Alliance. pp. 223–233. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-22124-8_22.
26. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An open urban driving simulator. In: Conference on robot learning. pp. 1–16. PMLR (2017).
27. Schmied, T., Hofmarcher, M., Paischer, F., Pascanu, R., Hochreiter, S.: Learning to Modulate pre-trained Models in RL, https://arxiv.org/abs/2306.14884, (2023). https://doi.org/10.48550/ARXIV.2306.14884.