

Conditional Behavior Prediction for Automated Driving on Highways

Christopher Diehl¹, Timo Osterburg¹, Nils Murzyn², Georg Schneider², Frank Hoffmann¹, Torsten Bertram¹

¹Institute of Control Theory and Systems Engineering, TU Dortmund
44227 Dortmund, Germany
E-Mail: forename.surname@tu-dortmund.de

²ZF Friedrichshafen AG, Artificial Intelligence Lab,
66123 Saarbrücken, Germany
E-Mail: forename.surname@zf.com

1 Introduction

Predicting the future behavior in multi-agent systems is critical for safe motion planning and control of self-driving vehicles (SDV). Here, prediction models must account for the static geometry (e.g., static obstacles and high-definition (HD) map data) of the environment and the interactions between different moving objects. Deep Learning architectures achieve state-of-the-art results on many forecasting benchmarks. Initially, methods like [3] describe the scene as a birds-eye-view image, coming with discretization inaccuracies, high memory consumption, and computational complexity. Recent approaches [2, 7, 9] describe the traffic scene as a graph. While improving the performance, standard models predict without considering a drivers intent. In contrast, [4] and [5] conditions the prediction on a single planned trajectory or the action of the SDV, which has the downside of possible overly confident anticipation of how the SDV may influence the other agents' behavior [8]. [6] and [10] condition on fixed goal states and [7] relies on a high-level route in urban environments. This work focuses on the highway environment and wants to answer the ques-

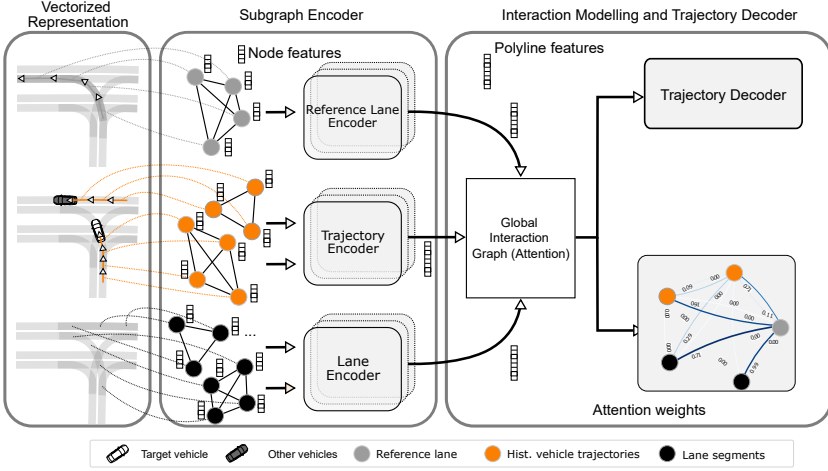


Figure 1: System architecture of conditional behavior prediction.

tion: *Do trajectory prediction methods improve by conditioning on high-level information in highway scenarios.*

2 Conditional Behavior Prediction

Problem Formulation. Assume a multi-agent system with $N + 1$ traffic participants (one target agent and $N \in \mathbb{N}^+$ other agents) described by discrete time steps t . $\mathbf{x}_t = [x_t, y_t]^\top$ describes one agent's past state and $\mathbf{y}_t = [x_t, y_t]^\top$ future state. \mathbf{x}^n and \mathbf{y}^n describe the sequence of past and future states of agent indexed by $n \in N + 1$ respectively, whereas the history has $H \in \mathbb{N}^+$ time steps and the future $F \in \mathbb{N}^+$ time steps. $n = 0$ denotes the index of the target agent. Let $\mathbf{X} = (\mathbf{x}^0, \mathbf{x}^1, \dots, \mathbf{x}^N)$ denote the joint historic state sequence of all agents, $\mathbf{M} \in \mathbb{R}^{n_M}$ an HD map and $\mathbf{R} \in \mathbb{R}^{n_R}$ the high-level route information with dimensions n_M and n_R . We could then model the probability density function of future target agent states with $p_\theta(\mathbf{y}^0 | \mathbf{X}, \mathbf{M}, \mathbf{R})$. Our goal is to learn the parameters $\theta \in \mathbb{R}^{n_\theta}$ with dimension n_θ of a neural network under a maximum likelihood estimation.

Approach. Similar to [11], the behavior prediction problem relies on a graph-based state representation shown in figure 1. Agent trajectories, lane information and high-level route are represented as polylines. Whereas driving lanes are represented by their left and right lane boundaries, the high-level route is captured by the lane centerline. Polyline $P_j \in \mathcal{P}$ with index $j \in \mathbb{N}^+$ are mapped onto $m - 1$ equidistant vectors $\mathbf{v}_i \in P_j$ with $\mathbf{v}_i = [(\mathbf{d}_i^s)^T, (\mathbf{d}_i^e)^T, (\mathbf{a}_i)^T, j]^T$. $\mathbf{d}_i^s, \mathbf{d}_i^e \in \mathbb{R}^2$ denote the 2-D start and end positions w.r.t. the self-driving vehicles coordinate system with $i \in \mathbb{N}^+$. Further, $\mathbf{a}_i \in \mathbb{R}^{n_a}$ is a set of polyline attributes with dimension n_a . A one hot vector classifies the lane type (route or normal lane polyline). In VectorNet (VN) [2], fully connected sub-graphs encode the corresponding information by multi-layer-perceptrons (MLP). A self-attention mechanism captures the higher-order interactions between sub-graphs and provides attention weights. Inspection of the attention weights provides inside into the relevance of polylines for the agents decision making. The trajectory is then decoded using another MLP. In the unimodal case, the network predicts a single trajectory that minimizes the L2 Loss between predicted and ground truth trajectory in the training data. This work also extends VN to predict k trajectories. In the multi-modal case, the approach minimizes the min-of-k loss [3]. C-VN denotes our conditional model.

3 Evaluation

This section evaluates the approach in highway scenarios with dense traffic in the CARLA [13] simulator (Version 0.9.11) by comparing C-VN to the unconditioned model (VN). The simulated training data is generated by the CARLA traffic manager mode that controls the behavior of all agents simultaneously. The lane change (LC) behavior of the target agent was modified using the LC model of [12].

Metrics. *Average Displacement Error (ADE):* The displacement error between the predicted trajectory and the ground truth averaged over all time steps for unimodal prediction. *Final Displacement Error (FDE):* The displacement error between the predicted trajectory and the ground truth averaged over the last

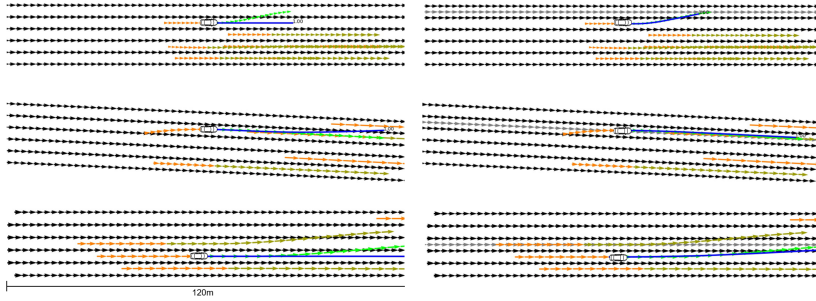


Figure 2: Qualitative comparison. Left: VN. Right: C-VN. All elements are visualized as vectors. Black: Lane Markings. Grey: Center line of the target lane. Orange: Agent position history. Blue: Predicted trajectory. Green: Ground truth future trajectory of the target agent. Yellow: Ground truth future trajectory of other agents.

time steps of each prediction, when the model only predicts one mode. *Minimum Average Displacement Error* ($\min_k \text{ADE}$): In the multimodal case, the model predicts k trajectories and the evaluation calculates the ADE for every mode and then takes the minimum.

Prediction Results. Figure 2 compares prediction results using VN or C-VN. Notice that the additional high-level information allows the model to predict lane changes earlier (rows one and three). Row two further shows that prediction better matches the ground truth and predicts a trajectory that terminates closer to the target lane center. The quantitative results in Table 1 support the initial hypothesis that conditioning benefits the trajectory prediction.

Applications to Motion Planning. In the framework of imitation learning the same approach can be employed for motion planning of an SDV rather than prediction. In that case, the vehicle is regulated along the predicted trajectory by underlying PID controllers for lateral and longitudinal motion. Figure 3 visualizes the planned trajectory during successful closed-loop control in a dense lane change scenario.

Table 1: Quantitative Prediction results. Bold numbers mark the best result.

	Metric	ADE [m]	↓FDE [m]	↓min ₂ ADE [m]	↓
VN		0.60	1.77	0.52	
C-VN		0.58	1.60	0.49	

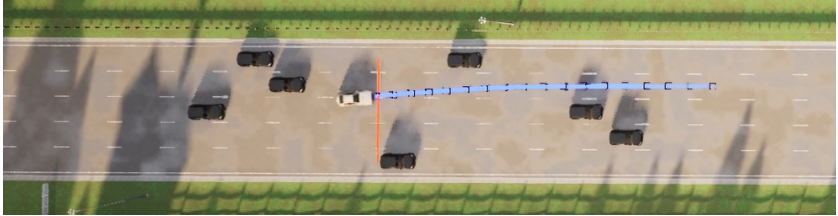


Figure 3: Results in CARLA by controlling a SDV by C-VN as a motion planning module. The blue visualizes the predicted trajectory of the SDV (white). The SDV performs a lane change in dense traffic triggered shortly before the orange line.

4 Conclusion

This work presents a graph-based conditional behavior prediction approach for automated driving. The comparative analysis reveals that the conditioning improves the prediction. We further demonstrate the suitability of conditional behavior prediction for motion planning. Future work should investigate the conditioning on network predictions of agents joint behaviors.

Acknowledgments

This research was funded by the Federal Ministry for Economic Affairs and Energy on the basis of a decision by the German Bundestag in the project ”KISSaF - AI-based Situation Interpretation for Automated Driving”.

References

- [1] A. Vaswani et al. “Attention is All you Need”. In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*. 2017.
- [2] J. Gao et al. “VectorNet: Encoding hd maps and agent dynamics from vectorized representation”. In: *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.
- [3] H. Cui et al. “Multimodal Trajectory Predictions for Autonomous Driving using Deep Convolutional Networks”. In: *International Conference on Robotics and Automation (ICRA)*. 2019.
- [4] T. Salzmann et al. “Trajectron++: Dynamically-Feasible Trajectory Forecasting With Heterogeneous Data”. In: *European Conference on Computer Vision (ECCV)*. 2020.
- [5] C. Diehl, T. Sievernich, M. Krüger, F. Hoffman, T. Bertram “UMBRELLA: Uncertainty-Aware Model-Based Offline Reinforcement Learning Leveraging Planning”. In: *Advances in Neural Information Processing Systems, Machine Learning for Autonomous Driving Workshop*. 2021.
- [6] N. Rhinehardt, R. McAllister, K. Kitani, S. Levine “PRECOG: PRediction Conditioned On Goals in Visual Multi-Agent Settings”. In: *International Conference on Computer Vision (ICCV)*. 2019.
- [7] S. Khandelwal et al. “What-if motion prediction for autonomous driving”. In: *arXiv preprint arXiv:2008.10587*. 2020.
- [8] S. Tang, W. Zhan, M. Tomizuka “Interventional Behavior Prediction: Avoiding Overly Confident Anticipation in Interactive Prediction”. In: *arXiv preprint arXiv:2204.08665*. 2022.
- [9] M. Liang et al. “Learning Lane Graph Representations for Motion Forecasting”. In: *European Conference on Computer Vision (ECCV)*. 2020.

- [10] J. Ngiam et al. “Scene Transformer: A unified architecture for predicting multiple agent trajectories”. In: *International Conference on Learning Representations (ICLR)*. 2022.
- [11] C. Diehl, T. Waldeyer, F. Hoffman T. Bertram “VectorRL: Interpretable Graph-based Reinforcement Learning for Automated Driving”. In: *31. Workshop Computational Intelligence..* 2021.
- [12] C. Wissing, K.-H. Glander, C. Haß, T. Nattermann, T. Bertram “Development and test of a lane change prediction algorithm for automated driving”. In: *Fahrerassistenzsysteme*. 2017.
- [13] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. “CARLA: An open urban driving simulator”. In *Proc. of the Conference on Robot Learning (CORL)*. 2017.