Improving Out-of-Distribution Generalization of Trajectory Prediction for Autonomous Driving via Polynomial Representations

Yue Yao[1,2], Shengchao Yan[3], Daniel Goehring[2], Wolfram Burgard[4], Joerg Reichardt[1]

Abstract— **Robustness against Out-of-Distribution (OoD) samples is a key performance indicator of a trajectory prediction model. However, the development and ranking of state-of-the-art (SotA) models are driven by their In-Distribution (ID) performance on individual competition datasets. We present an OoD testing protocol that homogenizes datasets and prediction tasks across two large-scale motion datasets. We introduce a novel prediction algorithm based on polynomial representations for agent trajectory and road geometry on both the input and output sides of the model. With a much smaller model size, training effort, and inference time, we reach near SotA performance for ID testing and significantly improve robustness in OoD testing. Within our OoD testing protocol, we further study two augmentation strategies of SotA models and their effects on model generalization. Highlighting the contrast between ID and OoD performance, we suggest adding OoD testing to the evaluation criteria of trajectory prediction models.**

I. INTRODUCTION

Trajectory prediction is essential for autonomous driving, with robustness being a key factor for practical applications. The development of trajectory prediction models is catalyzed through public datasets and associated competitions, such as Argoverse 1 (A1) [1], Argoverse 2 (A2) [2], and Waymo Motion (WO) [3]. These competitions provide a set of standardized metrics and test protocol that scores prediction systems on test data withheld from all competitors. This is intended to ensure two things: objective comparability of results and generalization ability due to the held-out test set.

Among the top-performing models based on deep learning, we observe a trend to ever more parameter-rich and expressive models [4], [5], [6] - explicitly trained with the data of each competition. This begs the question of whether the stellar performance of these models is due to their ability to adapt to each dataset specifically, i.e., over-fit. Despite efforts to guard against over-fitting by withholding test samples, these test examples still share similarities with the training samples, such as the sensor setup, map representation, post-processing, geographic, and scenario selection biases employed in dataset creation. Consequently, the test scores reported in each competition are examples of *In-Distribution* (ID) testing.

For practical application, the performance of trajectory prediction should be independent of these shared biases in a single dataset and measured by *Out-of-Distribution* (OoD) testing, e.g., across different datasets. As an example, Figure 1 clearly visualizes the distribution shift caused by different

map generation processes between A2 and WO. However, the efforts for cross-dataset evaluation are hampered by the in-homogeneous data format and task details among datasets and competitions. Notable distinctions are the different lengths of scenario, observation history, and prediction horizon in each dataset.
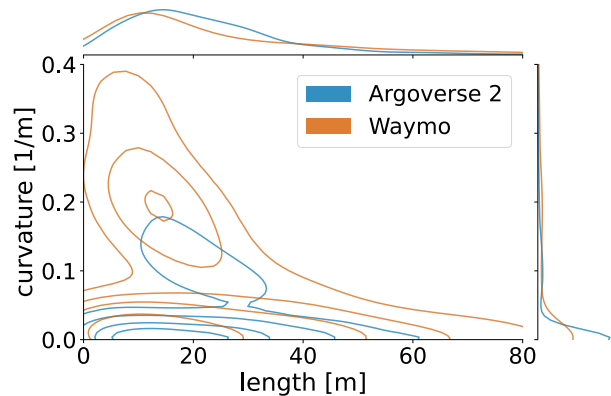

Fig. 1: Kernel density plot of the maximum absolute curvature and length for 5000 random lane segments in A2 and WO. Contours indicate the 20-th, 40-th, 60-th, and 80-th percentiles, respectively.

Therefore, in this work, we attempt to homogenize the data formats and prediction tasks to enable OoD testing across two large-scale motion datasets: A2 and WO. Based on our OoD testing, we further explore the possibilities to enhance model robustness against OoD samples.

One strategy to improve the robustness of any deep learning model is to increase the amount of training data. Bahari et al. [7] have shown that generalization performance can be improved by augmenting training data with programmatically created variations. While such an approach does increase generalization ability, it further increases training effort and cannot guarantee generalization. Further, ensuring that a data augmentation strategy covers all possible variations is not easy.

We follow a complementary strategy and instead of providing more training data, limit the expressiveness of our model by constraining its input and output representation. This comes with the added benefit of a much smaller model size, reduced training effort and faster inference times. At the same time, we retain near state-of-the-art (SotA) prediction performance for ID test cases and show significantly improved robustness in OoD testing. Our key contributions are as follows:

The authors are with [1]Continental AG, [2]Freie Universität Berlin, [3]Department of Computer Science, University of Freiburg, [4]Department of Engineering, University of Technology Nuremberg.
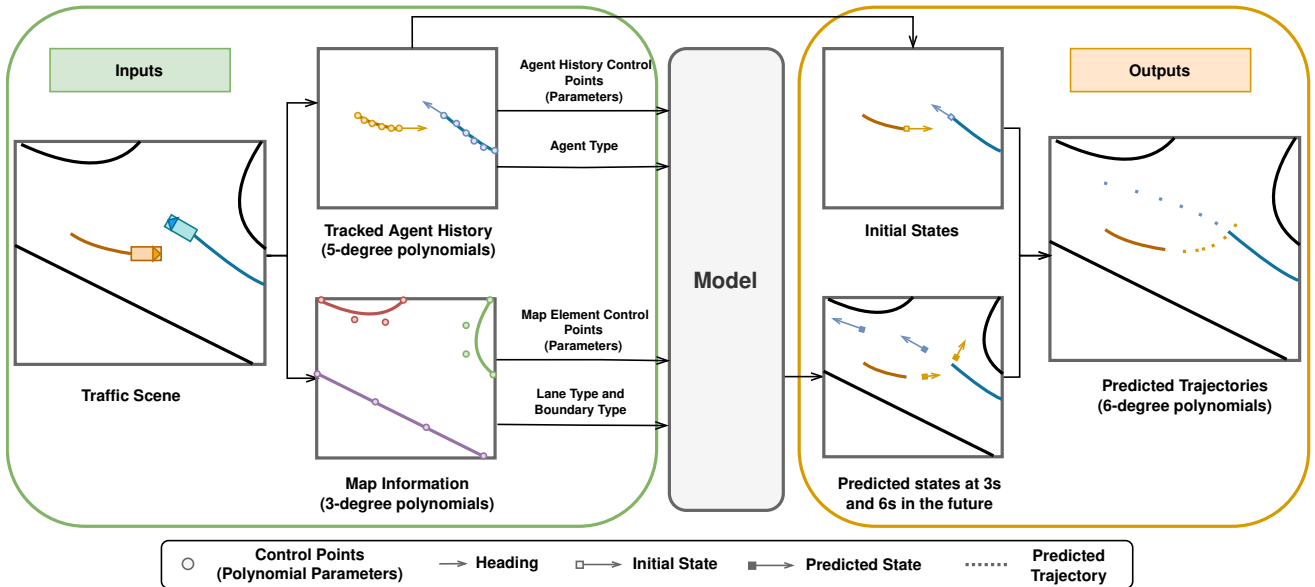
Fig. 2: Our proposed model architecture. **Inputs**: Agent histories and road geometry are both represented via polynomials. **Outputs**: The tracked initial states and predicted states are fused into one polynomial trajectory prediction, ensuring continuity of past observation and future prediction.

- We provide a dataset homogenization protocol that enables Out-of-Distribution (OoD) testing of prediction algorithms across different large-scale motion datasets.
- We study the OoD robustness of two SotA models and explore the effect of their augmentation strategies on In-Distribution (ID) and OoD test results.
- We propose an efficient multi-modal predictor baseline with competitive ID performance and superior OoD robustness by representing trajectories and map features parametrically as polynomials.

Our paper is organized as follows: We first review the recent trajectory prediction models and their design decisions with respect to data representation and augmentation strategy. Out of the body of related work, we select two benchmark models and detail how dataset and competition characteristics have shaped their model designs. Next, we propose our dataset homogenization protocol for OoD testing and detail our approach of using constrained parametric representations. We then report ID and OoD testing results for all three models plus a number of variants. Our results highlight the impact of different augmentation strategies in ID and OoD testing and the increase in OoD robustness due to our polynomial representations. Finally, we argue for adopting OoD testing as a performance measure of equal value to ID testing.

We open-source our code online [8].

II. RELATED WORK AND BENCHMARK MODELS

A. Data Representation

The formats of individual datasets and competition rules greatly influence the design decisions of the models proposed in the literature. Models based on deep learning typically opt for a direct ingestion of the data as given. Recent studies employ *sequence-based representation*, e.g., sequences of data points, as the model's input and output [9], [5], [10], [11]. This representation aligns the format of measurements in datasets and efficiently captures various information, e.g., agent trajectories and road geometries. The downside of this representation is the high redundancy and variance. The presence of measurement noise and outliers in datasets may lead to physically infeasible predictions. Additionally, the computational requirement of this approach scales with the length and temporal/spatial resolution of trajectories and road geometries.

Some previous works explored the possibility of using *polynomial representations* for predictions [12], [13]. Su et al. [13] highlight the temporal continuity of this representation, i.e., the ability to provide arbitrary temporal resolution. Reichardt [14] argues for using polynomial representations to integrate trajectory tracking and prediction into a filtering problem. Polynomial representations restrict the kind of trajectories that can be represented and introduce bias into prediction systems. This limited flexibility is generally associated with greater computational efficiency, smaller model capacity and hence better generalization. However, representing the *inputs* with polynomials has not been extensively researched and deployed in recent works. In a recent study, we showed that polynomials of moderate degree can represent real-world trajectories with a high degree of accuracy [15]. This result motivates the use of polynomial representations in our architecture.

B. Data Augmentation

Competitions typically designate one or more agents in a scenario as *focal* agent and only score predictions for focal agents. However, training the model exclusively with the focal agent's behavior fails to exploit all available data. To address this, predicting the future motion of non-focal

agents is a typical augmentation strategy for training. As there are many more non-focal agents than focal agents, another important design decision relates to how authors interpret the importance of focal vs. non-focal agent data.

We select two open-sourced and thoroughly documented SotA models: Forecast-MAE [9] (FMAE) and QCNet [5], with nearly 1900k and 7600k parameters, respectively. As summarized in Table I, both models employ sequence-based representation but exhibit different strategies in dealing with non-focal agents:

Heterogeneous Augmentation: FMAE follows the prediction competition protocol and prioritizes focal agent prediction. Thus, agent history and map information are computed within the *focal agent's* coordinate frame. Compared to the *multi-modal* prediction of the focal agent, FMAE only outputs *uni-modal* prediction for non-focal agents. The prediction error of the focal agent also weighs higher in loss function than non-focal agents.

Homogeneous Augmentation: QCNet does not focus on the selected focal agent and proposes a more generalized approach. It encodes the information of agents and map elements in each agent's *individual* coordinate frame. It outputs *multi-modal* predictions for focal and non-focal agents alike, ensuring a consistent prediction task for all agents. The loss of focal agent prediction shares the same weight as non-focal agents.

Figure 3 illustrates the augmentation strategies of our benchmark models FMAE and QCNet.
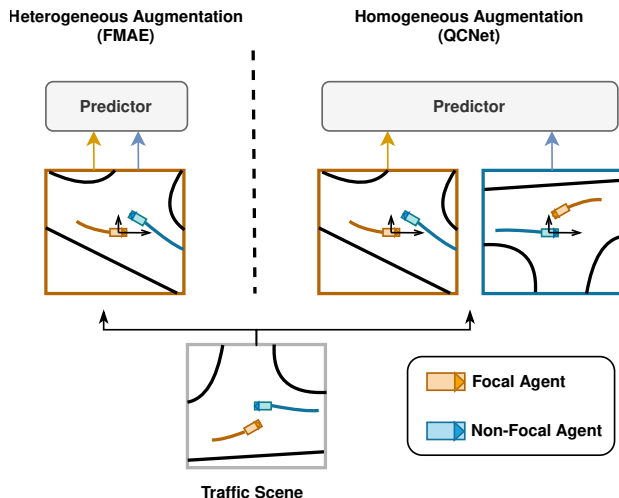

Fig. 3: The two augmentation strategies for non-focal agent data employed in benchmark models. **Left**: FMAE employs heterogeneous augmentation, representing information in the focal agent's coordinate frame and only making uni-modal predictions for non-focal agents. **Right**: QCNet employs homogeneous augmentation, encoding information in each agent's individual coordinate frame and making multi-modal predictions for both focal and non-focal agents alike.

To study the influence of these augmentation approaches independently of data representation, we add a third augmentation strategy for comparison: **no augmentation**. This can

simply be achieved by removing the loss of non-focal agents in each model, thus limiting the model to learn from focal agent behavior only. We denote the benchmark models without augmentation as "FMAE-noAug" and "QCNet-noAug".

For our own model, we will implement all three modes of augmentation for comparison.

TABLE I: Model variants under study based on augmentation strategies and data representations.

augmentation	input and output representation	
strategy	sequence-based	polynomial-based (ours)
heterogeneous	FMAE [9]	EP-F
homogeneous	QCNet [5]	EP-Q
w/o augmentation	FMAE-noAug QCNet-noAug	EP-noAug

III. OoD testing and Dataset Homogenization

The differences in the data collection processes and sensor platforms between motion datasets of different origin present us with the opportunity to perform OoD testing on truly independent data samples. However, this also comes with the challenges of working around inconsistencies in data formats and prediction tasks between datasets.

We propose our OoD testing protocol by training and testing models on two different large-scale datasets: A2 and WO. We summarise the characteristics of both datasets in Table II. Algorithmic improvements in generalization ability are more easily identified when training on the smaller of two datasets and testing on the larger ones. Consequently, we train on the training set from A2, comprising 199 908 samples. As the agents' future trajectories in the test sets of the A2 and WO competitions are not accessible, they cannot be included in our homogenization protocol. Therefore, for ID testing, we settle for the validation split of A2, and for OoD testing, we use the validation split of WO. The difference between ID and OoD results demonstrates a model's robustness against OoD test cases.

There are multiple notable distinctions between A2 and WO. One obvious distinction is the different scenario lengths: a scenario in A2 is 11 seconds long and consists of 5 seconds of observation history and 6 seconds of future to predict. WO only provides 1.1 seconds of observation history but requires an 8-second prediction horizon. Moreover, A2 designates and scores only one *focal agent* per scenario, whereas there are up to 8 such *focal agents* in WO. The focal agent in A2 is never the ego vehicle and remains fully observed throughout the prediction horizon, which is not guaranteed in WO.

To facilitate the OoD testing, we homogenize the datasets and adopt the following settings for both datasets and summarize them in Table II:

- History Length: We set the history length to 5 seconds (50 steps) as in A2.
- Prediction Horizon: We maintain the 6-second prediction horizon for training but only evaluated the first 4.1s of prediction due to the limited recording length in WO.
- Map Information: We exclude boundary information and the label of junction lanes due to the information's absence

TABLE II: Datasets comparison and homogenization protocol for OoD testing.

	Argoverse 2 (A2) Dataset [2]	Waymo Open Dataset (WO) [3]	Homogenized Dataset train	test
#samples (train / val / test)	199 908 / 24 988 / 24 984	487 002 / 44 097 / 44 920	-	
#cities	6	6	-	
scenario length	11s	9.1s	-	
sampling rate	10 Hz	10 Hz	-	
history length	5s	1.1s	5s	
prediction horizon	6s	8s	6s	4.1s
map information — lane center	yes	yes	yes	
map information — lane boundary	yes	not for junction lanes	no	
map information — junction lanes labeled	yes	no	no	
prediction target — # focal agents	1	up to 8	1	
prediction target — ego included	no	yes	no	
prediction target — fully observed	yes	not guaranteed	yes	

in WO. Only lane segments and crosswalks are considered map elements in homogenized data due to their availability in both datasets.

- Focal Agent: We take the same focal agent as labeled in A2. From WO, only the first fully observed, non-ego agent in the list of focal agents is chosen. As the list of focal agents is unordered, this corresponds to randomly sampling a single fully observed focal agent.

Fulfilling the focal agent's selection criterion above, we have 24 988 valid samples from the A2 validation set for ID testing and 42 465 valid samples from the WO validation set for OoD testing, respectively.

IV. PROPOSED MODEL AND DATA REPRESENTATION

The key innovation of our model is its choice of representation for map elements and trajectories on both input and output sides of the model. Here, we propose to use a parametric representation in terms of polynomials. This constrains the model's expressiveness compared to sequence-based data representations. Such a parametric representation is easily pre-computed for map elements and algorithms for online tracking and filtering agent observations into trajectories already exist [14].

Our prior work on trajectory representation [15] has shown that polynomials are an accurate representation of agent trajectories and should not present a limiting factor to the achievable prediction performance. Independent research suggests that sequence-based data representations may carry spurious high-frequency information, which can result in over-fitting [16].

With the polynomial inputs and outputs as our key innovation, we denote our contribution: *Everything Polynomial* (EP). We now elaborate our approach to represent data with polynomials.

A. Inputs as Polynomials

We employ Bernstein polynomials to represent the agent history and map geometry. The parameters of Bernstein polynomials have spatial semantics as *control points*.

Agent History: Based on the Akaike Information Criterion (AIC) [17] presented in [15], the 5-second history trajectories of vehicles, cyclists, and pedestrians in A2 and WO are represented optimally with 5-degree polynomials. We also use the 5-degree polynomial for the ego vehicle. We track the control points of agent history with the method proposed in [14] and incorporate the observation noise models proposed in [15].

Map: Map elements, such as lane segments and crosswalks, are represented with 3-degree polynomials - a degree consistent with OpenDRIVE [18] standards. We fit the sample points of map elements via the total-least-squares method by Borges-Pastva [19]. Lane segments in WO are longer and more curved than those in A2 as visualized in Figure 1, posing a challenge for fitting them with 3-degree polynomials. Therefore, we iteratively split the lane segments by half until the fit error is under 0.1m.

The polynomial representation only requires 40.8% and 8.7% of data space compared to the sample points provided in A2 and WO, respectively.

B. Outputs as Polynomials

The 6-second predicted trajectories are formulated as 6-degree polynomials, one degree higher than indicated by AIC in [15]. Instead of predicting the polynomial parameters of future trajectories, our model predicts several future kinematic vehicle states. By fusing the predicted future states with current tracked states, we recalculate the polynomial parameters and reconstruct the future trajectory. Details are presented in Appendix I.

C. Model

The pipeline of EP is illustrated in Figure 2. EP employs the popular encoder and decoder architecture and its implementation details are presented in Appendix I.

Following the different augmentation strategies by our benchmark models, we implemented three EP variants: (1) EP-F employs the heterogeneous augmentation of FMAE. (2) EP-Q employs the homogeneous augmentation of QCNet. (3) EP-noAug has no augmentation. Table I summarizes the design choices of the different models under study.

Due to the compact polynomial representations for inputs and outputs, EP variants only employ around 345k model parameters, which is only 4.5% the size of QCNet and 18.2% the size of FMAE as listed in Table III.

Since we focus on OoD robustness, scaling up our model for more model capacity and potentially better ID performance is not prioritized in this work.

TABLE III: Result of In-Distribution testing (6-second prediction horizon) in competition setting on Argoverse 2 test set. For all the metrics, lower the better. Inference time is tested on one Tesla T4 GPU with a scenario of 50 agents and 150 map elements.

model	$minADE_1$ [m]	$minFDE_1$ [m]	$minADE_6$ [m]	$minFDE_6$ [m]	# model params [k]	inference time [ms]
QCNet [5]	1.702 (100.0%)	4.309 (100.0%)	0.643 (100.0%)	1.244 (100.0%)	7600 (100.0%)	120.89 (100.0%)
Forecast-MAE [9] w pre-train	1.741 (102.2%)	4.355 (101.1%)	0.709 (110.3%)	1.392 (111.9%)	1900 (25.0%)	12.63 (10.4%)
EP-Q (ours)	2.134 (125.4%)	5.415 (125.7%)	0.841 (131.0%)	1.683 (135.3%)	334 (4.4%)	5.72 (4.7%)
EP-F (ours)	1.887 (110.9%)	4.567 (106.0%)	0.801 (124.6%)	1.526 (122.7%)	345 (4.5%)	4.66 (3.9%)
EP-F w flip (ours)	1.846 (108.5%)	4.456 (103.4%)	0.786 (122.2%)	1.485 (119.4%)		

V. EXPERIMENTAL RESULTS

A. Experiment Setup on Homogenized Data

On the homogenized A2 training data, all models are trained from scratch with hyperparameters reported for A2 competition performance. Due to our dataset homogenization settings, we have to adjust the data pre-processing and corresponding encoding layers in QCNet, e.g., removing the layers for encoding the label of junction lanes. FMAE's architecture is not affected by our data homogenization. Training settings of EP variants are summarized in Appendix I-C.

B. Metrics

For ID testing, we use the official benchmark metrics, including minimum Average Displacement Error ($minADE_K$) and minimum Final Displacement Error ($minFDE_K$), for evaluation. The metric $minADE_K$ calculates the Euclidean distance in meters between the ground-truth trajectory and the best of K predicted trajectories as an average of all future time steps. Conversely, $minFDE_K$ focuses solely on the prediction error at the final time step, emphasizing long-term performance.

For OoD testing, we also propose $\Delta minADE_K$ and $\Delta minFDE_K$ as the difference of displacement error between ID and OoD testing to measure model robustness. In accordance with standard practice, K is selected as 1 and 6.

C. ID Results with Competition Settings

Like the other studies, we first present the best ID test results of all models according to the A2 prediction competition protocol in Table III. Benchmark results are from the original authors. For EP, we also include the results when augmenting the data by symmetrizing around the x-axis, i.e. flipping all left turns into right turns and vice versa.

Though the performance in ID testing on the A2 test set is not the focus of our work, our EP still achieves near state-of-the-art performance with a significantly smaller model size. For instance, with the "flip" data augmentation, the $minFDE_1$ of EP-F is only 3.4% higher compared to QCNet.

Though EP exhibits a considerable gap compared to QCNet in terms of multi-modal prediction (K=6), it exhibits only less than 10cm additional prediction error in $minADE_6$ and $minFDE_6$ compared to FMAE with pre-train. EP has a faster inference time, requiring only 3.9% of QCNet's and 36.9% of FMAE's inference time, which is essential in real-time applications. Another notable point is the better ID performance of EP-F compared to EP-Q. This indicates that heterogeneous augmentation shows more effectiveness than homogeneous augmentation in ID testing when only a single focal agent is scored.

Next, we examine the impact of dataset homogenization and augmentation. In Figure 4, we can see that the alterations due to dataset homogenization, e.g., excluding lane boundaries, have only little effect on model performance. Removing the data augmentation, however, has a small but noticeable negative effect on ID performance for both benchmark models. These results also show that our retraining works as expected.
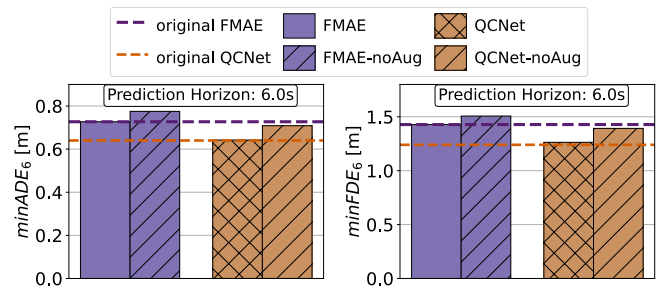

Fig. 4: ID performance of benchmarks on A2. **Dashed Lines**: Results reported by original authors with competition settings. **Bars**: Results for retrained benchmark models on homogenized datasets.

D. OoD Testing Results

The key question now is whether the results achieved in ID testing on A2 will translate to OoD testing on WO. Figure 5 summarizes the results. As discussed, the relative

and absolute increase in the prediction metric will serve as our measure of robustness and generalization ability. We will present OoD results from three perspectives: (1) augmentation strategy, (2) data representation, and (3) contrast results between ID and OoD testing.

1) Augmentation Strategy: We observe that without augmentation, i.e., excluding the non-focal agents in the loss function, all models generalize poorly on all metrics, but our model has the smallest relative and absolute increase in error in all cases.

Though heterogeneous augmentation does marginally reduce FMAE's prediction error for both ID and OoD testing, the relative and absolute increase in prediction error between ID and OoD samples is practically the same as in the non-augmented case, e.g., $+0.279$m (61.5%) vs. $+0.295$m (69.7%) in $\Delta minADE_6$.

On the other hand, the robustness of QCNet significantly benefits from the homogeneous augmentation with reduced $\Delta minFDE_1$ from $+1.802$m (71.5%) to $+1.067$m (46.4%). Compared to EP-noAug, EP-F and EP-Q replicate the behavior observed from FMAE and QCNet, e.g., with only a marginal impact from heterogeneous augmentation and a significant impact from homogeneous augmentation.

2) Data Representation: Our EP variants exhibit improved robustness compared to benchmarks with sequence-based data independent of data augmentation strategy. For instance, EP-F outperforms FMAE by demonstrating lower prediction errors and a lower increase in prediction errors in OoD testing. Similarly, compared to QCNet, EP-Q also demonstrates improved robustness, even achieving a reduction in prediction error of -0.022m (-1.9%) in $\Delta minADE_1$.

3) ID vs OoD results: We observe multiple reversals in performance between ID and OoD testing results: (1) Despite FMAE demonstrating comparable ID performance to QCNet, it exhibits significantly lower robustness than QCNet in OoD testing. (2) While FMAE and QCNet outperform EP on ID samples, EP shows lower prediction error than FMAE and QCNet across multiple metrics in OoD testing. (3) There is a reversal in performance between EP-F and EP-Q in both ID and OoD testing.

TABLE IV: ID and OoD testing results (4.1-second prediction horizon) of EP-F and EP-F (w flip).

| model | $minADE_1 | minFDE_1$ [m] | | |
	ID		OoD	
EP-F	1.08	2.56	1.32	3.45
EP-F (w flip)	1.06	2.50	1.35	3.53

Moreover, we also note the contrasting impact of "data flipping" in ID and OoD testing, as summarized in Table IV. This indicates that improper data augmentation can in fact impair model generalization.

The differences between ID and OoD results underscore the importance of OoD testing for a thorough evaluation of trajectory prediction models.
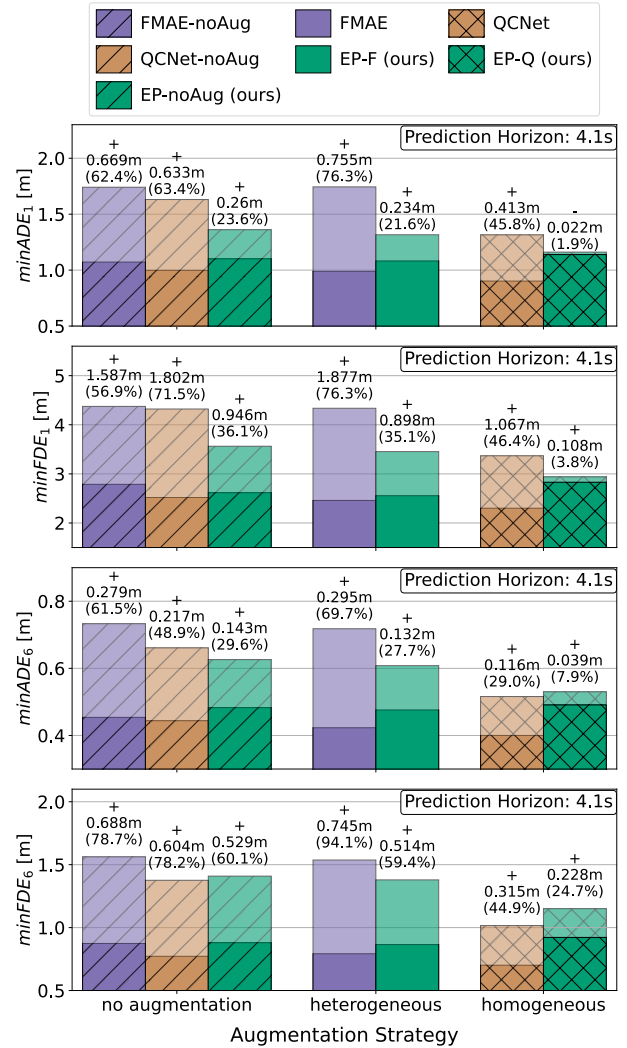

Fig. 5: The OoD testing results of FMAE, QCNet, EP and their variants. We indicate the *absolute* and *relative* difference in displacement error between ID and OoD results. **Solid**: The ID results by training on the homogenized A2 training set and testing on the homogenized A2 validation set. **Transparent**: The increased displacement error in OoD testing by training on the homogenized A2 training set and testing on the homogenized WO validation set.

VI. CONCLUSION

Recent SotA trajectory prediction models have thoroughly optimized their In-Distribution (ID) performance and present outstanding results in test sets evaluated in individual prediction competition. However, better ID performance does not automatically guarantee higher Out-of-Distribution (OoD) performance. In fact, performance rankings can and do reverse in OoD settings. Our proposed OoD testing protocol enables a fresh perspective for model evaluation in trajectory prediction that we hope the community will adopt. With our protocol, we demonstrate the robustness improvement from *homogenous augmentation* and prove the benefits of *polynomial representation* as employed in our EP model.

With a much smaller model size and lower inference time, EP achieves near SotA ID performance. Additionally, EP exhibits significantly improved out-of-distribution (OoD) robustness with minimal performance drop compared to benchmark models. This work represents an initial step toward a trajectory prediction model that is capable of generalizing to different datasets and is robust under changes in sensor setup, scenario selection strategy or post-processing of training data. More sophisticated models based on this concept will be developed to improve the performance for both In- and Out-of-Distribution evaluations.

ACKNOWLEDGMENT

REFERENCES

[1] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3d tracking and forecasting with rich maps," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[2] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, *et al.*, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[3] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. R. Qi, Y. Zhou, *et al.*, "Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9710–9719.

[4] Z. Zhou, L. Ye, J. Wang, K. Wu, and K. Lu, "Hivt: Hierarchical vector transformer for multi-agent motion prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 8823–8833.

[5] Z. Zhou, J. Wang, Y.-H. Li, and Y.-K. Huang, "Query-centric trajectory prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 863–17 873.

[6] S. Shi, L. Jiang, D. Dai, and B. Schiele, "MTR++: multi-agent motion prediction with symmetric scene modeling and guided intention querying," *CoRR*, vol. abs/2306.17770, 2023.

[7] M. Bahari, S. Saadatnejad, A. Rahimi, M. Shaverdikondori, A. H. Shahidzadeh, S.-M. Moosavi-Dezfooli, and A. Alahi, "Vehicle trajectory prediction works, but not everywhere," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 123–17 133.

[8] Continental AG, "Everything polynomial," https://github.com/continental/everything-polynomial, 2024.

[9] J. Cheng, X. Mei, and M. Liu, "Forecast-mae: Self-supervised pretraining for motion forecasting with masked autoencoders," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8679–8689.

[10] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning lane graph representations for motion forecasting," in *European Conference on Computer Vision*. Springer, 2020, pp. 541–556.

[11] B. Varadarajan, A. Hefny, A. Srivastava, K. S. Refaat, N. Nayakanti, A. Cornman, K. Chen, B. Douillard, C. P. Lam, D. Anguelov, *et al.*, "Multipath++: Efficient information fusion and trajectory aggregation for behavior prediction," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 7814–7821.

[12] T. Buhet, E. Wirbel, A. Bursuc, and X. Perrotton, "Plop: Probabilistic polynomial objects trajectory planning for autonomous driving," *CoRR*, vol. abs/2003.08744, 2020.

[13] Z. Su, C. Wang, H. Cui, N. Djuric, C. Vallespi-Gonzalez, and D. Bradley, "Temporally-continuous probabilistic prediction using polynomial trajectory parameterization," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 3837–3843.

[14] J. Reichardt, "Trajectories as markov-states for long term traffic scene prediction," in *14-th UniDAS FAS-Workshop*, 2022, p. 14.

[15] Y. Yao, D. Goehring, and J. Reichardt, "An empirical bayes analysis of object trajectory representation models," in *IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2023, pp. 902–909.

[16] D. Yin, R. Gontijo Lopes, J. Shlens, E. D. Cubuk, and J. Gilmer, "A fourier perspective on model robustness in computer vision," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[17] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Selected papers of hirotugu akaike*. Springer, 1998, pp. 199–213.

[18] Association for Standardization of Automation and Measuring Systems (ASAM), "ASAM OpenDRIVE," https://www.asam.net/standards/detail/opendrive/, 2023, accessed: 07.03.2024.

[19] C. F. Borges and T. Pastva, "Total least squares fitting of bézier and b-spline curves to ordered data," *Computer Aided Geometric Design*, vol. 19, no. 4, pp. 275–289, 2002.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[21] R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T. Liu, "On layer normalization in the transformer architecture," in *International Conference on Machine Learning*. PMLR, 2020, pp. 10 524–10 533.

APPENDIX I
IMPLEMENTATION DETAILS OF OUR MODEL

A. Model

1) Input: The control points of A agents in either *focal agent* or *individual* coordinate are denoted as $\boldsymbol{\omega}_n^a \in \mathbb{R}^{A \times 2}$ with $n \in \{0, 1, \cdots, 5\}$. Following the popular vectorization approach, we compute the vector between control points as $\boldsymbol{\delta}_n^a = \boldsymbol{\omega}_n^a - \boldsymbol{\omega}_{n-1}^a$ for capturing the kinematic of agents. To describe the spatial relationship, we employ the last control point $\boldsymbol{\omega}_5^a$ and the normalized vector $\boldsymbol{\theta}_5^a = norm(\boldsymbol{\delta}_5^a)$ as the reference position and orientation for agents. The time window of agent appearance is denoted as $\boldsymbol{TW} \in \mathbb{R}^{A \times 2}$. Therefore, the features of agents include $\boldsymbol{\Delta}^a = [\boldsymbol{\delta}_1^a, \boldsymbol{\delta}_2^a, \cdots, \boldsymbol{\delta}_5^a]$, $\boldsymbol{PI}^a = [\boldsymbol{\omega}_5^a, cos(\boldsymbol{\theta}_5^a), sin(\boldsymbol{\theta}_5^a)]$ and $\boldsymbol{TW}$. The features of M map elements are computed similarly with considering the initial point as the reference, denoting as $\boldsymbol{\Delta}^m = [\boldsymbol{\delta}_1^m, \boldsymbol{\delta}_2^m, \boldsymbol{\delta}_3^m]$ and $\boldsymbol{PI}^m = [\boldsymbol{\omega}_0^m, cos(\boldsymbol{\theta}_1^m), sin(\boldsymbol{\theta}_1^m)]$.

2) Encoder: As visualized in Figure 6, we employ simple 3-layer MLPs to encode the features of agents and map elements. Semantic attributes, such as agent type and lane type, are encoded with individual embedding layers. Embedded features are element-wise added and fused to agent tokens $\boldsymbol{T}^a \in \mathbb{R}^{A \times D}$ and map element tokens $\boldsymbol{T}^m \in \mathbb{R}^{M \times D}$, where D denotes the hidden dimension. Multiple attention blocks based on Transformer [20] and "pre-layer normalization" [21] perform the *map element-map element*, *agent-map element* and *agent-agent* attentions sequentially and update $\boldsymbol{T}^m$ and $\boldsymbol{T}^a$. All EP variants share the same encoder architecture.

3) Decoder: In Figure 7, we visualize the two decoder architectures of EP-F (EP-noAug) and EP-Q due to different augmentation strategies.
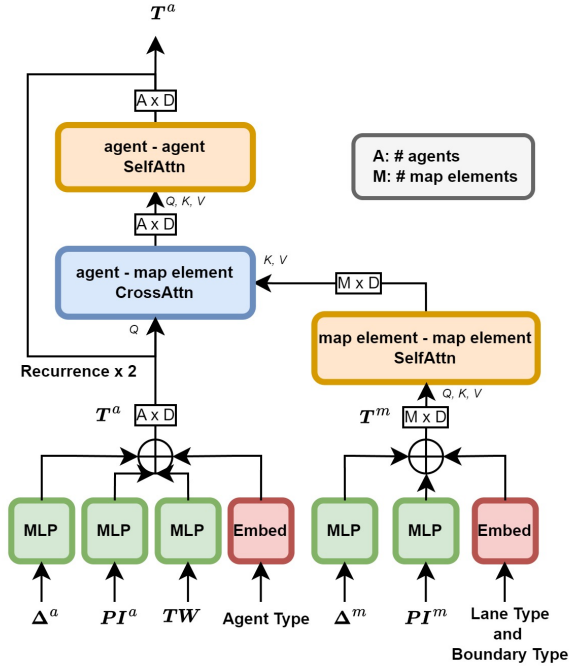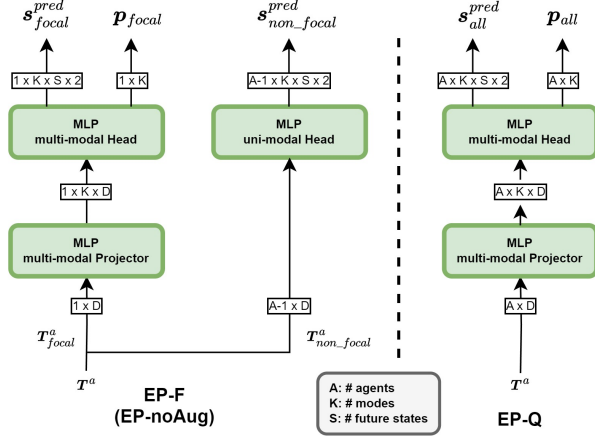
Fig. 6: Encoder architecture of EP variants.

Fig. 7: Decoder architectures of EP-F, EP-Q and EP-noAug.

The embedded agent features can be decoded in two ways: (1) For multi-modal prediction, the agent token T^a is firstly projected to multiple modes and substantially decoded to predicted states s^{pred} and probabilities p for each mode. (2) For uni-modal prediction, the agent token T^a is directly decoded to the predicted states of a single mode.

Consider the k-th mode of i-th agent, we concatenate the tracked and predicted states as the vector $s_{i,k} = [p_{i,0}^{track}, p_{i,k,30}^{pred}, v_{i,k,30}^{pred}, a_{i,k,30}^{pred}, p_{i,k,60}^{pred}, v_{i,k,60}^{pred}, a_{i,k,60}^{pred}]$, $s_{i,k} \in \mathbb{R}^{14}$, where $p_{i,0}^{track}$ denotes the tracked position at current time step, $p_{i,k,30}^{pred}, v_{i,k,30}^{pred}, a_{i,k,30}^{pred}$ are the predicted position, velocity and acceleration at 3s (30 steps) in the future, respectively. $p_{i,k,60}^{pred}, v_{i,k,60}^{pred}, a_{i,k,60}^{pred}$ are the predicted position, velocity and acceleration at 6s (60 steps) in the future, respectively.

We follow the same notation as in [15] and form

the monomial basis functions as a vector $\phi(t_m) \in \mathbb{R}^7$ by concatenation $\phi(t_m) = [\phi_0(t_m), \cdots, \phi_6(t_m)]^\top$, where t_m denotes the timestamp at m-th step in the future. The derivatives of the monomial basis function $\dot{\phi}(t_m)$ and $\ddot{\phi}(t_m)$ can be acquired by incorporating the linear derivative operator $D \in \mathbb{R}^{7 \times 7}$ mentioned in [14]. We construct the 14×14 observation matrix $H = [\phi(t_0), \phi(t_{30}), \dot{\phi}(t_{30}), \ddot{\phi}(t_{30}), \phi(t_{60}), \dot{\phi}(t_{60}), \ddot{\phi}(t_{60})]^\top \otimes \mathbf{I}_2$, where $\mathbf{I}_2$ is a 2×2 identity matrix and $\otimes$ denotes the Kronecker product. The trajectory parameters $\omega_{i,k}$ of the k-th predicted mode are expressed as $\omega_{i,k}^{pred} = (H^\top H)^{-1} H^\top s_{i,k}$.

B. Training Loss

We employ $minADE_6$ as regression loss $l^{reg,multi}$ and weighted average displacement error, with the weight of predicted probability p, as classification loss $l^{cls,multi}$ for multi-modal prediction. We calculate average displacement error as regression loss for uni-modal prediction $l^{reg,uni}$. Thus, the training losses of EP variants are expressed as:

$$
\begin{aligned}
l_{EP-F} &= l_{focal}^{reg,multi} + l_{focal}^{cls,multi} + l_{non\_focal}^{reg,uni} \\
l_{EP-Q} &= l_{all}^{reg,multi} + l_{all}^{cls,multi} \\
l_{EP-noAug} &= l_{focal}^{reg,multi} + l_{focal}^{cls,multi}
\end{aligned}
\tag{1}
$$

where sub-indices "focal," "non-focal," and "all" refer to the focal agent, non-focal agents, and all agents, respectively.

C. Experiment Settings

We report the settings of EP variants in Table V. Note that the batch size and number of epochs are designed to ensure the same training iterations for all EP variants.

TABLE V: Settings for EP Variants

model	EP-F & EP-noAug	EP-Q
hidden dimension D	64	
optimizer	Adam	
learning rate	1e-3	5e-4
learning rate schedule	cosine	
batch size	64 (128 w flip)	32
training epochs	128	64
warmup iterations	6e4	
dropout	0.2	