

Fig. 5. Trained with 2k samples of each functional scenario (6k samples overall), VectorNet can predict the Ego's trajectory for all three functional scenarios.

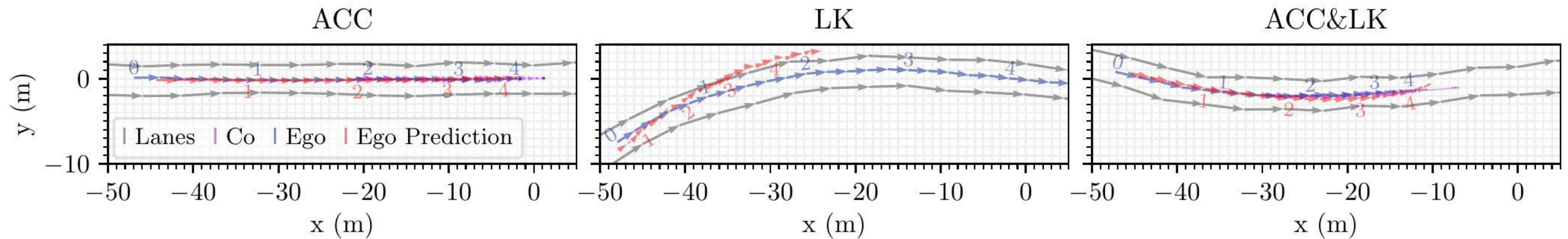


Fig. 6. Trained on 2k samples of the scenario ACC&LK, VectorNet learns that the Ego follows the Co, which enables predictions for the scenario ACC. However, the prediction for the scenario LK indicates that VectorNet primarily considers the Co's trajectory to predict the lateral behavior of the Ego, not the lanes. VectorNet does not know that the Ego's lateral behavior is determined by the lanes, and only the longitudinal behavior is determined by the Co.

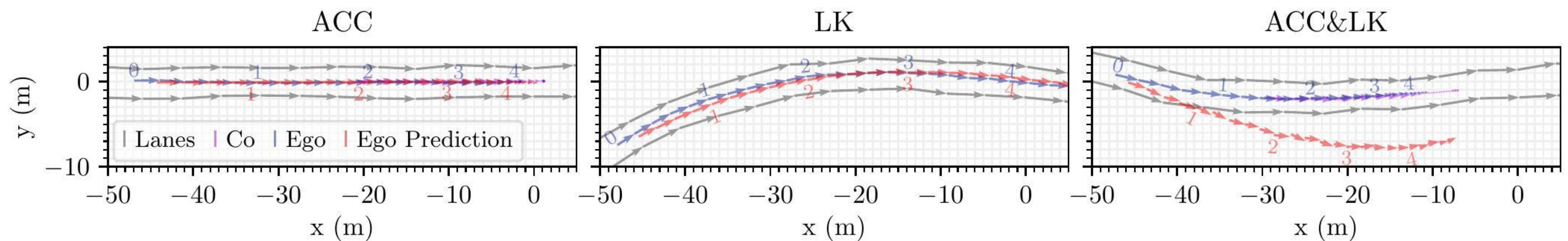


Fig. 7. Trained on the scenarios ACC and LK (2k samples each), generalization to ACC&LK is not achieved. Here, within the training data, the presence of the Co indicates a straight trajectory of the Ego; curved lanes indicate a constant velocity of the Ego. Accordingly, there is insufficient variance for generalization.

since ACC is a special case of ACC&LK (a straight road). The generalization to LK, however, is poor. Row 4 shows that no generalization from ACC and LK to ACC&LK is achieved. Fig. 6 and Fig. 7 indicate that these results are due to insufficient variance in the training data.

Row 9 shows that generalization from ACC and LK to ACC&LK is achieved if some data from the scenario ACC&LK are added. The resulting ADE is better than what is achieved with the same number of samples of ACC&LK on their own (see row 10). Hence, existing data can improve predictions for new functional scenarios or reduce the amount of data necessary to achieve a certain predictive performance.

Originally, VectorNet uses 211k training and 41k validation samples [23 p. 5]. Our results show that with $\leq 2k$ samples, reasonable ADEs are achievable. We expect that adjusting the architecture or hyperparameters would allow working with even less data or increasing predictive performance.

B. Assessment Based on Predicted Evaluation Metrics

Next, we assess the prediction of evaluation metrics derived from VectorNet's predictions (see Fig. 1). The baselines are an extra-tree (ET) and a Bayesian neural network (BNN) with [17]'s hyperparameters. All models are trained with 2k samples. Table III shows that for different evaluation metrics, different models achieve the best mean average errors (MAEs).

The regression models' high performance can be attributed to them learning directly on the relevant features (5 to 10 scenario inputs). However, they cannot account for the complex spatial and temporal nuances described in the scenario embeddings (although it would be possible to take these nuances into account, this would result in up to 150 features, which regression models can hardly process). VectorNet, on the other hand, can account for the nuances but must first identify the relevant features influencing the Ego's trajectory (for the ACC scenarios, the feature (FT) matrix has 581 entries¹). Hence, learning based on vectorized scenario embeddings is more flexible but also more challenging.

$$^1(2 \text{ lanes} \cdot 24 \frac{\text{vec.}}{\text{lane}} + \text{Co} \cdot 25 \frac{\text{vec.}}{\text{Co}} + \text{Ego} \cdot 1 \frac{\text{vec.}}{\text{Ego}}) \cdot 7 \frac{\text{FT}}{\text{vec.}} = 518 \text{ FT}$$

TABLE III
MEAN AVERAGE ERRORS (MAE) OF PREDICTED EVALUATION METRICS

Scenario	Evaluation Metric	MAE ET	MAE BNN	MAE VectorNet
ACC	a_{\min}	$0.13 \frac{\text{m}}{\text{s}^2}$	$0.07 \frac{\text{m}}{\text{s}^2}$	$0.58 \frac{\text{m}}{\text{s}^2}$
	d_{\min}	0.87 m	0.44 m	0.26 m
LK	$p_{\text{lat max}}$	0.09 m	0.08 m	0.38 m
ACC&LK	a_{\min}	$0.19 \frac{\text{m}}{\text{s}^2}$	$0.09 \frac{\text{m}}{\text{s}^2}$	$0.74 \frac{\text{m}}{\text{s}^2}$
	d_{\min}	1.14 m	0.55 m	0.43 m
	$p_{\text{lat max}}$	0.08 m	0.08 m	0.21 m