Causal Learning in Autonomous Driving: Evaluating the Role of Synthetic Data and Regularization in Trajectory Prediction

Student: Antoine Tissot-Favre
Supervisor: Ahmad Rahimi
CS-433, EPFL Lausanne, Switzerland

***Abstract*—This study explores causality in autonomous driving, focusing on synthetic data effectiveness, causal regularization, and out-of-distribution (OOD) generalization. Using MetaDrive, ScenarioNet, and UniTraj, we generate factual and counterfactual driving scenarios to analyze agent-specific causal effects. Results show that synthetic data enables effective training, but merging it with real-world data does not improve performance. Causal regularization enhances OOD robustness, with ranking loss outperforming contrastive loss. These findings support the development of causality-aware trajectory models and future Sim-to-Real adaptation.**

I. Introduction

Causal reasoning in autonomous driving plays a crucial role in ensuring robust trajectory prediction. Recent works have applied causality analysis to pedestrian interactions [1], providing valuable insights into social behaviors and model generalization. However, adapting these methods to autonomous vehicles introduces new challenges:

- **Higher Degrees of Freedom:** Unlike pedestrians, vehicles have complex motion dynamics governed by speed, acceleration, and steering constraints.
- **Dependence on Maps:** Vehicle trajectory predictions require structured map information, including lane markings and traffic rules.
- **Policy Variability:** Vehicles adhere to diverse driving policies (e.g., defensive, aggressive, rule-based).
- **Collision Dynamics:** Unlike pedestrians, vehicles must account for mechanical constraints and crash risks.
- **Environmental Factors:** Weather conditions, lighting, and road obstacles introduce additional complexity.

While these factors pose unique challenges, this study focuses on the impact of maps, driving policies, and traffic density in causal reasoning for autonomous vehicles. Other constraints remain interesting areas for future exploration, though they may require advancements in simulation tools to be adequately addressed.

II. Related Work

The study of causality in multi-agent systems lies at the intersection of several key areas, including simulation frameworks, causal learning, and trajectory prediction. This section highlights the tools and methods most relevant to our work.

Simulation Frameworks. Simulation environments play a critical role in studying multi-agent interactions and testing causal hypotheses. MetaDrive [2] and ScenarioNet [3] provide robust platforms for generating diverse driving scenarios, enabling detailed analysis of vehicle interactions under varying conditions. UniTraj [4] extends these capabilities by offering a unified framework for trajectory prediction.

Causal Learning. Recent advances in causal learning have highlighted its potential for improving out-of-distribution generalization and model interpretability [1]. In particular, causal annotations and counterfactual analysis have been shown to significantly enhance robustness in multi-agent contexts, as demonstrated in pedestrian interaction studies. This work builds directly on these findings by adapting causal regularization techniques to the vehicular domain.

Sim-to-Real Causal Transfer. The Sim-to-Real Causal Transfer framework [1] pioneered causal analysis in pedestrian interactions, using metric learning approaches to model causality. This study provides the foundation for our work, which seeks to extend similar methodologies to autonomous vehicles. systems.

III. Formalism

As explained in [1], formalizing the robustness challenge in multi-agent systems involves understanding the causal relationships between agents in a dynamic environment. For instance, in a motion forecasting task, the ego vehicle's trajectory is influenced by its neighboring agents. Given the joint state of all agents at a time step, a robust representation should capture how the removal or modification of certain agents alters the trajectory of the ego vehicle.

To quantify this, counterfactual simulations are employed to measure the causal effect of removing one or more agents. Formally, the causal effect is defined as:

$$E_R = |y^{\emptyset} - y^{R}|_2, \tag{1}$$

where $y^{\emptyset}$ represents the trajectory of the ego agent in the original scene, y^{R} represents its trajectory in the perturbed scene where agents R are removed, and $|\cdot|_2$ denotes the point-wise Euclidean distance. This metric enables a systematic evaluation of agent-specific influence on the ego vehicle's behavior.

In contrast to prior benchmarks, such as CausalAgents [1], which rely on human-annotated causal labels, our approach leverages simulation tools like MetaDrive and ScenarioNet to generate paired factual and counterfactual scenarios. Instead of manually labeling causal agents, we use a thresholding approach based on the observed distribution of causal effect metrics across multiple simulations. While this method still requires some manual tuning to define the threshold, it significantly reduces the need for subjective per-agent annotations, enabling a more scalable evaluation of causally-aware representations.

IV. Simulation Framework

The simulation framework forms the foundation for causal analysis in autonomous driving by generating diverse factual and counterfactual datasets. It integrates three core platforms: MetaDrive for procedural scenario generation, ScenarioNet for managing counterfactual variations, and UniTraj for standardized trajectory data and causal annotations [2], [3], [4].

A. Scenario Generation

Scenarios were procedurally generated using the Expert policy, which optimizes for efficient driving in complex environments. Traffic density was randomly sampled within a predefined range (between 0.17 and 0.23) to introduce variability. While the framework allows for different policies and controlled traffic densities, the dataset used in this study was generated under these specific conditions.

To ensure reproducibility, random seeds were used. Counterfactuals were derived by systematically removing agents and rerunning simulations to assess their causal influence on the ego vehicle's trajectory.

B. Data Structure

Each scenario was stored hierarchically, comprising:

- **Metadata:** Unique identifiers and configuration details.
- **Object Trajectories:** Agent states (e.g., position, velocity, acceleration).
- **Map Information:** Lane boundaries and polyline-based road features.

This structured format ensured seamless integration with UniTraj for downstream causal analysis.

C. Moment Selection via Kalman Filtering

To focus on high-impact interactions, a Kalman filter was applied to identify the most complex moments based on high prediction errors. This selection process prioritized scenarios with intense agent-to-agent and agent-to-map interactions while preventing redundancy by ensuring diverse time steps across scenarios. The filter operated using:

$$\hat{x}_{k+1} = \hat{x}_k + v_k \Delta t, \quad \hat{P}_{k+1} = P_k + Q \tag{2}$$

where $\hat{x}_k$ is the predicted state, v_k is velocity, and P_k is the covariance matrix.

D. Validation Metrics

To maintain high data quality, invalid scenarios (e.g., crashes, off-road events) were discarded. Additionally, a continuous validation metric ensured that scenario consistency was preserved over time.

By leveraging these methodologies, the simulation framework provided a scalable and rigorous approach for studying causality in autonomous driving.

V. Causal Training

The causal training phase aimed to develop models capable of identifying and quantifying causal relationships in vehicular interactions. This was achieved through a structured pipeline where the model simultaneously learned trajectory prediction and incorporated causal regularization techniques.

A. Model Architecture

The model follows an encoder-decoder architecture designed to process both factual and counterfactual trajectories:

- The **encoder** extracts representations of input trajectories.
- The **decoder** predicts future ego vehicle paths based on factual data.

To explicitly capture causal effects, the model integrates:

- **Causal Embeddings:** The encoder jointly processes factual and counterfactual data to highlight agent-specific causal influence.
- **Trajectory Prediction:** The decoder predicts future trajectories for both factual and counterfactual scenarios.

These representations serve as the foundation for both causal metric estimation and causal regularization losses.

B. Causal Regularization

To enforce causal structure within the learned representations, we apply two distinct regularization losses:

Contrastive Loss. This loss aligns non-causal counterfactual embeddings with factual representations while increasing separation from causal ones:

$$L_{\text{contr.}} = -\log \frac{\exp(d^+/\tau)}{\exp(d^+/\tau) + \sum_k \exp(d_k/\tau) 1_{E_k > \eta}}, \tag{3}$$

where d^+ is the distance to a positive (distant) counterfactual, d_k represents distances to non-causal counterfactuals, τ is a temperature hyperparameter, and η is the causal effect threshold.

Ranking Loss. This loss enforces a separation margin between causal and non-causal embeddings:

$$L_{\text{ranking}} = \max(0, d_i - d_j + m), \tag{4}$$

where d_i and d_j denote distances to agents with different causal influences, and m is the margin parameter.

These losses encourage the model to distinguish causal from non-causal agents during training.

C. Training Pipeline

The model is optimized through a multitask learning framework:

- **Embedding Learning:** The encoder generates trajectory representations for factual and counterfactual scenarios.
- **Trajectory Prediction:** The decoder predicts future trajectories based on factual data.
- **Causal Metric Computation:** Predictions for counterfactual scenarios allow estimation of agent-specific causal effects using ACE metrics (defined below).
- **Loss Computation:** The model is optimized with both a trajectory prediction loss (supervised learning) and a causal regularization loss (contrastive or ranking).
- **Hyperparameter Tuning:** Grid search was used to optimize learning rate and regularization weights.

This structured approach ensures that the model learns robust causal representations while maintaining trajectory prediction accuracy.

D. Causal Metrics

To quantify the influence of individual agents on the ego vehicle's trajectory, we define the following metrics:

Non-Causal Average Causal Effect (NC-ACE):

$$\text{NC-ACE} = \frac{1}{|NC|} \sum_{i \in NC} |y^{\emptyset} - y^{R}|_2, \tag{5}$$

where $y^{\emptyset}$ is the factual trajectory and y^{R} is the counterfactual trajectory with non-causal agents removed.

Direct Causal Average Causal Effect (DC-ACE):

$$\text{DC-ACE} = \frac{1}{|DC|} \sum_{i \in DC} |y^{\emptyset} - y^{R}|_2, \tag{6}$$

where DC represents direct causal agents.

Overall ACE:

$$\text{ACE} = \frac{1}{|NC| + |DC|} \sum_{i \in (NC+DC)} |y^{\emptyset} - y^{R}|_2. \tag{7}$$

These metrics provide a quantitative measure of how individual agents impact the ego vehicle's trajectory.

E. Evaluation Protocol

The trained models were assessed across both in-distribution and out-of-distribution (OOD) settings:

- **Baseline Comparisons:** Performance was compared against models trained without causal regularization.
- **Causal Metric Analysis:** NC-ACE and DC-ACE distributions were analyzed across different scenarios.
- **Qualitative Validation:** Factual and counterfactual trajectories were visualized to assess model interpretability.

To ensure a consistent and fair evaluation, all models were assessed using standard trajectory prediction metrics:

- **Final Displacement Error (FDE)**: Measures the Euclidean distance between the predicted final position and the ground-truth final position. Lower values indicate better long-term accuracy.
- **Minimum Average Displacement Error (minADE6)**: Computes the average displacement error over all time steps for the best trajectory among the top-6 predicted trajectories (i.e., the trajectory closest to the ground truth at all time steps).
- **Minimum Final Displacement Error (minFDE6)**: The final displacement error of the best trajectory among the top-6 predictions (i.e., the trajectory that ends closest to the ground truth).
- **Brier Final Displacement Error (Brier FDE)**: A variant of FDE that incorporates prediction confidence, penalizing both large errors and overconfident incorrect predictions.
- **Miss Rate**: The proportion of predicted trajectories that deviate beyond a predefined threshold from the ground truth.

This evaluation strategy ensures that the model effectively captures causal relationships while maintaining generalization across diverse driving scenarios.

VI. Experimental Results and Findings

A. Baseline Experiments

The primary objective of these experiments was to assess whether the AutoBot model could effectively learn from synthetic data and to evaluate the impact of combining real and synthetic datasets. To systematically investigate these questions, we conducted a series of training runs with different data configurations:

- **AutoBot trained on NuScenes:** A baseline model trained exclusively on real-world data from NuScenes, serving as a reference for comparison.
- **AutoBot trained on Synthetic Data:** The model was trained solely on procedurally generated driving scenarios to determine the learnability and effectiveness of synthetic data.
- **AutoBot trained on Merged Data (NuScenes + Synthetic):** The model was trained on a combination of real and synthetic data to analyze whether integrating both datasets improves performance or introduces inconsistencies.

1) Analysis and Key Insights: Examining the results summarized in Table I, we derive several key insights from the conducted experiments.

- **Synthetic Data is Learnable:** The model trained solely on synthetic data achieved the best performance across all metrics, with the lowest Brier FDE (1.56) and minADE6 (0.53). This indicates that the procedural dataset provides meaningful patterns for trajectory prediction.

Model	Brier FDE ↓	minADE6 ↓	minFDE6 ↓	Miss Rate ↓
AutoBot on NuScenes	2.41	0.87	1.81	0.30
AutoBot on Synthetic	**1.56**	**0.53**	**0.93**	**0.14**
AutoBot on Merged Data	2.57	0.54	1.91	0.31

Table I
PERFORMANCE OF AUTOBOT ON DIFFERENT DATASETS. LOWER VALUES INDICATE BETTER PERFORMANCE.

- **Challenges with Real Data:** The model trained on NuScenes alone exhibited the highest error rates, with a minADE6 of 0.87 and a miss rate of 30%. This suggests that real-world trajectory prediction is inherently more challenging due to increased variability, sensor noise, and diverse driving behaviors.
- **Merging Real and Synthetic Data Does Not Help:** Unsurprisingly, combining real and synthetic data resulted in worse performance compared to training on either dataset individually. The AutoBot model trained on the merged dataset had the highest Brier FDE (2.57) and minFDE6 (1.91), indicating that discrepancies between real and synthetic distributions introduced inconsistencies rather than improving generalization.

These findings highlight a key limitation: while synthetic data is a viable standalone training resource, naive merging with real-world data does not necessarily improve performance and may even degrade it. This emphasizes the need for more sophisticated domain adaptation techniques to bridge the gap between synthetic and real-world driving scenarios.

B. Impact of Causal Regularization and Out-of-Distribution Robustness

To evaluate the effects of causal regularization on model generalization, we conducted a two-step experiment:

1) **In-Distribution Analysis:** We first trained models with different losses (contrastive, ranking, and none) and tested them on standard (in-distribution) data to establish a baseline comparison.
2) **Out-of-Distribution (OOD) Evaluation:** The trained models were then assessed on unseen traffic densities and policy shifts to measure their robustness under new conditions.

1) Step 1: Training with Regularization: To determine the optimal regularization strength, a grid search was performed over $\lambda \in \{0.1, 1, 10, 100, 1000\}$. The best performances were observed with $\lambda \in \{100, 1000\}$, and the results are summarized in Table II.

Key Insights from In-Distribution Training::

- **Regularization Improves Causal Awareness:** Both ranking and contrastive loss models slightly improved ACE metrics, suggesting that causal embeddings help the model better differentiate between influential and non-influential agents.
- **Trade-off in Accuracy:** Regularized models showed a small increase in minADE6 and minFDE6, indicating that causal constraints may slightly reduce short-term accuracy in favor of better generalization.
- **Baseline Model is Overconfident:** The unregularized AutoBot model had the lowest Brier FDE, indicating high confidence in in-distribution predictions. However, this does not necessarily translate to better robustness in unseen environments.

2) Step 2: Out-of-Distribution (OOD) Generalization: After training, we tested all models in two out-of-distribution settings and the result are summarized in Table III:

- **Higher Traffic Density:** Evaluating performance under traffic densities significantly higher than those seen during training (increased from 0.17–0.23 to 0.3–0.6).
- **Policy Shift:** Testing the models in environments controlled by the Intelligent Driver Model (IDM) instead of the expert policy used during training.

Key Observations from OOD Testing:

- **Regularization Improves OOD Performance:** Models trained with causal loss exhibited better ACE scores in both high-density and policy-shift scenarios, confirming that causal embeddings improve generalization.
- **Ranking Loss Provides Strongest Causal Awareness:** Ranking loss models had the lowest ACE ALL and ACE NC values, indicating that they better differentiate between causal and non-causal agents.
- **Models Handle Traffic Density Better than Policy Shift:** ACE ALL and ACE DC values are lower in the high-density setting compared to policy shift, suggesting that models generalize better when dealing with increased traffic than when adapting to new driving behaviors.
- **Regularized Models Reduce Prediction Errors:** Both minADE6 and minFDE6 were lower for regularized models in the traffic density setting, indicating that causal constraints improve trajectory accuracy more effectively in dense traffic conditions than under policy shifts.
- **Brier FDE Shows Increased Robustness:** Regularized models demonstrated lower Brier FDE values in OOD settings, with traffic density changes yielding more confident predictions than policy shifts.
- **Miss Rate Confirms Improved Stability:** The miss rate was consistently lower for regularized models, reinforcing that causal constraints help prevent major

Model	ACE ALL ↓	ACE DC ↓	ACE NC ↓	Brier FDE ↓	minADE6 ↓	minFDE6 ↓	Miss Rate ↓
Contrastive $\lambda = 1000$	202.1	665.33	18.5	1.39	0.578	0.716	0.047
Ranking $\lambda = 1000$	201.13	661.6	17.76	1.41	0.580	0.726	0.049
Contrastive $\lambda = 100$	203.6	664.61	19.89	1.40	0.570	0.713	0.046
Ranking $\lambda = 100$	202.68	662.58	18.85	1.39	0.569	0.715	0.046
No Reg. AutoBot	203.73	664.9	19.06	**1.35**	**0.55**	**0.672**	**0.041**

Table II
BASELINE PERFORMANCE COMPARISON OF AUTOBOT MODELS TRAINED WITH DIFFERENT REGULARIZATION TECHNIQUES. LOWER VALUES INDICATE BETTER ACCURACY.

Table III
OUT-OF-DISTRIBUTION (OOD) EVALUATION: PERFORMANCE OF AUTOBOT MODELS UNDER UNSEEN TRAFFIC DENSITIES AND POLICY SHIFTS. LOWER VALUES INDICATE BETTER GENERALIZATION.

Model	ACE ALL ↓	ACE DC ↓	ACE NC ↓	Brier FDE ↓	minADE6 ↓	minFDE6 ↓	Miss Rate ↓
OOD - Increased Traffic Density							
Ranking $\lambda = 1000$	103.43	653.02	13.17	4.62	2.31	3.86	0.41
Ranking $\lambda = 100$	101.7	654.59	11.86	4.87	2.42	4.13	0.42
Contrastive $\lambda = 100$	101.19	654.4	11.46	4.88	2.5	4.13	0.43
No Reg. AutoBot	101.25	654.8	11.3	5.1	2.8	4.35	0.48
OOD - Policy Shift (IDM instead of Expert Policy)							
Ranking $\lambda = 1000$	111.55	667.76	11.69	5.17	2.61	4.43	0.466
Contrastive $\lambda = 1000$	110.4	667.22	12.66	5.36	2.7	4.61	0.468
No Reg. AutoBot	110.33	670.07	11.18	5.29	2.71	4.44	0.49

deviations from the ground truth trajectory. However, traffic density scenarios resulted in lower miss rates than policy shifts, indicating more stable predictions.

- **Baseline Model Struggles More in Policy Shift:** The unregularized AutoBot had the highest miss rate and FDE scores, particularly in the policy shift setting, highlighting its reduced reliability when encountering unfamiliar agent behaviors.

To conclude this part, we have seen that causal regularization enhances model robustness, particularly in unseen traffic conditions. The models generalize better under traffic density increases than under policy shifts, suggesting that adapting to new agent behaviors is more challenging than handling increased complexity. Ranking loss provides the strongest generalization, while contrastive loss enhances stability. Future work could explore combining both techniques and introducing policy-aware learning strategies to improve adaptability.

VII. CONCLUSION

This study investigated the role of synthetic data, causal regularization, and out-of-distribution (OOD) robustness in trajectory prediction for autonomous driving. We evaluated whether synthetic data could effectively train models, whether combining real and synthetic data improves performance, and how causal regularization affects generalization.

Key Findings:

- **Synthetic Data is a Viable Alternative:** The model trained exclusively on synthetic data achieved strong predictive accuracy, demonstrating that procedurally generated scenarios provide meaningful learning signals.
- **Merging Real and Synthetic Data Does Not Improve Performance:** The combined dataset led to degraded performance, suggesting domain adaptation challenges between procedural and real-world driving distributions.
- **Causal Regularization Enhances OOD Robustness:** Regularized models exhibited improved generalization, with ranking loss outperforming contrastive loss in distinguishing causal interactions.
- **Trade-Off Between Accuracy and Causality:** While causal constraints improve robustness, they introduce slight performance trade-offs in in-distribution settings, indicating a balance between causal learning and trajectory accuracy.

Implications for Sim-to-Real Transfer. While this study focused on evaluating causal representations in simulation, the next logical step is leveraging the existing framework to implement Sim-to-Real (Sim2Real) adaptation. Future work should focus on:

- Using existing tools to implement Sim2Real transfer within the current pipeline, without introducing additional external components.
- Validating generalization on real-world datasets such as NuScenes, ensuring that learned causal relationships remain interpretable and effective beyond simulation.

Final Thoughts. Our results underscore the importance of causal reasoning in autonomous driving and highlight the potential of synthetic data for training robust trajectory models. However, the challenges of domain adaptation and Sim2Real transfer remain critical areas for future research. By further refining causal learning techniques and integrating Sim2Real mechanisms into our framework, we can move

closer to developing trajectory prediction models that are both interpretable and reliable in real-world autonomous driving scenarios.

REFERENCES

[1] Y. Liu, A. Rahimi, P.-C. Luan, F. Rajič, and A. Alahi, "Sim-to-real causal transfer: A metric learning approach to causally-aware interaction representations," 2023. [Online]. Available: https://arxiv.org/abs/2312.04540

[2] Q. e. a. Li, "Metadrive: Composing diverse driving scenarios for rl," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[3] Q. Li, Z. Peng, L. Feng, Z. Liu, C. Duan, W. Mo, and B. Zhou, "Scenarionet: Open-source platform for large-scale traffic scenario simulation and modeling," *Advances in Neural Information Processing Systems*, 2023.

[4] L. Feng, M. Bahari, K. M. B. Amor, É. Zablocki, M. Cord, and A. Alahi, "Unitraj: A unified framework for scalable vehicle trajectory prediction," *arXiv preprint arXiv:2403.15098*, 2024.