Adversarial Backdoor Attack by Naturalistic Data Poisoning on Trajectory Prediction in Autonomous Driving

Supplementary Material

1. Gray-box Attack Scenario

To further analyse the vulnerability of the prediction models, we examine the impact of the proposed attack in a gray-box setting. More specifically, for the surrogate, we use the same model as the victim and conduct the experiments with different seeds and parameters. Following the experiment in the previous section, we launch the proposed attack on HiVT and MM-Transformer.

According to the results in Table 1, as expected, in the gray-box attack scenario, the values of both tCA and tASR are improved by $5.10(\%)$ and $7.34(\%)$ on HiVT and MMTransformer, respectively, indicating that the attacks were more successful. However, the small gap between black and gray box scenarios shows that the proposed attack is still very effective even without direct knowledge of the victim model, therefore can be deployed under different conditions and achieve comparable results.

Table 1. Ablation study on the effectiveness of the proposed attack in gray-box vs back-box scenarios. (↑) indicates the attack is more successful.

	Black-box		Gray-box	
Backdoor-inj.	**tCA(↑)**	**tASR(↑)**	**tCA(↑)**	**tASR(↑)**
HiVT [9]	95.30	91.11	97.88 (+2.58)	96.21 (+5.10)
MMTrans. [5]	84.21	78.55	89.43 (+5.22)	85.89 (+7.34)

2. Additional Experiments

2.1. Attack's Stealthiness

In backdoor attack literature, CA and ASR are two commonly used metrics to measure attacks' unnoticeably and effectiveness. *Stealthiness is more of a subjective property and depends on the application*, e.g. for images SSIM is used to detect anomalies resulted from triggers created by alterations in the color space [3] or by adding artificial reflections [4]. Alternatively, an attack's resilience or evasiveness against defenses can be measured, e.g. by input preprocessing or activation analysis (we use gradient reshaping). As mentioned in the papar, our disguising method conceals the trigger (AtV's observation) within the road layout. Since the attack is not evident in the training data, existing preprocessing defenses, e.g. trajectory smoothing [1, 8], are ineffective. To prove this, we evaluated our most effective attack (with the highest tASR) using MMTransformer and HiVT as surrogate and backdoor-injected models (see Table 2). We report the results with no attack as "original". We used the same test set for both "original" (for the clean model) and tCA (for backdoor-injected model), termed *clean test set*. The first row is the baseline as in the paper. From the table, *without any attack*, trajectory smoothing on training data hurts the performance by 18.18% (0.66→0.78) in the *clean test set* since the data distributions are altered. Therefore, the preprocessing step limits the model's capacity to mitigate potential backdoors in the training data. In the attack scenario, when we apply trajectory smoothing and the triggers undergo the proposed disguising step (second row), our attack remains largely effective, with a minimal degradation of 2.72% (91.11→88.39) on tASR. This is because, in our trigger-generation, we have a length control and a clipping step. Hence, triggers are more similar to normal trajectories and consequently less affected by preprocessing. On the other hand, tCA drops by 21.06% (95.30→74.24) which is only 2.88% (21.06-18.18) below the clean version of the model's maximum potential on the *clean test set*. In the attack scenario, when we apply trajectory smoothing without disguising step (third row), the performance drops by approx. 24% on both metrics. This confirms the effectiveness of our disguising method and the fact that *trajectory smoothing is not an effective defense, even in training time attacks, agreeing with the findings in [8].*

2.2. Proposed Metrics

In defining our metrics, tASR and tCA, we established thresholds th_2 and th_1 using Argoverse's distribution, where mean widths of lanes and vehicles are 3.7m and 1.7m. Based on these statistics, 1*m deviation is an upper bound for a car not shifting to another lane if driving in the lane center.* Here, we also report the results for different thresholds in Table 3, showing that even under restrictive conditions (0.25m of tCA and 2m for tASR), the attack remains unnoticeable and effective.

2.3. Attack's Robustness to Inference Noise

For real-world applicability, we model potential inference noises based on the data distribution in Argoverse. Inspired by the anchor-based prediction approaches [2], we use k-means clustering to identify a set of fixed anchors, corresponding to trajectory distribution modes in data, with each distribution characterized by the cluster's mean and variance. The statistics describe the central tendency and spread of data points within each cluster. We randomly sample from a Gaussian distribution that matches each cluster's mean and variance and add the generated noises to test data, both clean and poisoned sets. We evaluated our attack's effectiveness on the noisy samples using $k = 64$ clusters [2], as shown in Table 4. These results suggest that the proposed attack is robust against potential inference time noises by achieving a success rate as high as 90.46.

3. Additional Defense Mechanisms

In addition to the defense mechanism evaluated in Section 4.4 of the paper, we show that whether another category of de-

Table 2. Ablation on the trigger's stealthiness. "*" indicates clean model results and "Traj. sm." stands for trajectory smoothing.

Original (↓*)	**Traj. sm.**	**Disguising**	**Benign/Poison** (↑)	**tCA**(↑)	**tASR**(↑)
0.66	×	✓	0.75/3.67	95.30	91.11
0.78 (-18.18%)	✓	✓	2.18/2.94	74.24 (-21.06%)	88.39 (-2.72%)
		×	2.41/2.61	70.67 (-24.63%)	64.12 (-24.73%)

Table 3. Ablation on metrics' thresholds.

Threshold	**tCA**(↑)	**Threshold**	**tASR**(↑)
th_1= 0.75m	96.04	th_2= 1m	95.30
th_1= 0.50m	91.11	th_2= 1.5m	93.66
th_1= 0.25m	84.51	th_2= 2m	89.18

Table 4. Ablation on the attack's robustness.

Number of clusters $(k) = 64$		
Trigger	**tCA**(↑)	**tASR**(↑)
Original	91.11	95.30
With noise	83.37	90.46

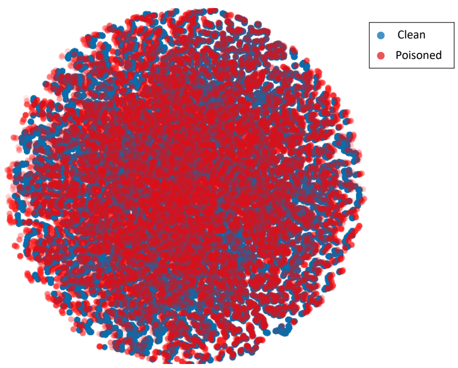

Figure 1. The latent space visualization of both clean and poisoned samples.

fences, namely latent space inspection, can be used against the proposed attack. Recent studies on defense mechanisms show that backdoor attacks tend to leave a tangible trace in the latent space of the backdoor-injected (victim) model. Therefore, the latent representations of the clean and poisoned samples form separate clusters. By inspecting the distributions of the representations, for instance by using methods, such as K-means, one can determine whether the training data is poisoned.

From this perspective, we visualize the latent representations of poisoned vs clean samples. We obtain the representations from AgentFormer [7], a trajectory prediction model that uses a conditional variational autoencoder (CVAE), and utilize t-SNE [6] to visualize the clusters formed by the samples. As illustrated in Fig. 1, the latent representations of the clean and poisoned samples are distributed similarly and are not forming well-separated clusters. This means that, the latent space inspection defense mechanisms can not be effective against the proposed backdoor attack.

References

[1] Yulong Cao, Chaowei Xiao, Anima Anandkumar, Danfei Xu, and Marco Pavone. AdvDO: Realistic adversarial attacks for trajectory prediction. In *ECCV*, 2022. 1

[2] Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. MultiPath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. In *CoRL*, 2019. 1

[3] Wenbo Jiang, Hongwei Li, Guowen Xu, and Tianwei Zhang. Color backdoor: A robust poisoning attack in color space. In *CVPR*, 2023. 1

[4] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *ECCV*, 2020. 1

[5] Yicheng Liu, Jinghuai Zhang, Liangji Fang, Qinhong Jiang, and Bolei Zhou. Multimodal motion prediction with stacked transformers. In *CVPR*, 2021. 1

[6] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 2008. 2

[7] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *ICCV*, 2021. 2

[8] Qingzhao Zhang, Shengtuo Hu, Jiachen Sun, Qi Alfred Chen, and Z Morley Mao. On adversarial robustness of trajectory prediction for autonomous vehicles. In *CVPR*, 2022. 1

[9] Zikang Zhou, Luyao Ye, Jianping Wang, Kui Wu, and Kejie Lu. HiVT: Hierarchical vector transformer for multi-agent motion prediction. In *CVPR*, 2022. 1