DriveDPO: Policy Learning via Safety DPO For End-to-End Autonomous Driving

Shuyao Shang[1,2] * **Yuntao Chen**[3] * **Yuqi Wang**[1,2] **Yingyan Li**[1,2] **Zhaoxiang Zhang**[1,2] †

[1]NLPR, Institute of Automation, Chinese Academy of Sciences,
[2]University of Chinese Academy of Sciences [3]MiroMind
{shangshuyao2024, wangyuqi2020, liyingyan2021, zhaoxiang.zhang}@ia.ac.cn
chenyuntao08@gmail.com

Abstract

End-to-end autonomous driving has substantially progressed by directly predicting future trajectories from raw perception inputs, which bypasses traditional modular pipelines. However, mainstream methods trained via imitation learning suffer from critical safety limitations, as they fail to distinguish between trajectories that appear human-like but are potentially unsafe. Some recent approaches attempt to address this by regressing multiple rule-driven scores but decoupling supervision from policy optimization, resulting in suboptimal performance. To tackle these challenges, we propose **DriveDPO**, a Safety Direct Preference Optimization Policy Learning framework. First, we distill a unified policy distribution from human imitation similarity and rule-based safety scores for direct policy optimization. Further, we introduce an iterative Direct Preference Optimization stage formulated as trajectory-level preference alignment. Extensive experiments on the NAVSIM benchmark demonstrate that DriveDPO achieves a new state-of-the-art PDMS of 90.0. Furthermore, qualitative results across diverse challenging scenarios highlight DriveDPO's ability to produce safer and more reliable driving behaviors.

1 Introduction

End-to-end autonomous driving has achieved remarkable progress in recent years. Unlike traditional modular pipelines, end-to-end methods directly predict future trajectories from raw sensor inputs, avoiding error accumulation across modules and simplifying the overall system design. Mainstream approaches [1–4] primarily rely on imitation learning, as shown in Fig. 1a. While effective in producing human-like behavior, imitation learning faces two critical safety issues. First, imitation learning minimizes the geometric distance between predicted and human trajectories. However, even slight deviations from human trajectories may lead to dangerous outcomes, as shown in Fig. 2a. Second, commonly used symmetric loss in imitation learning, such as mean squared error, penalizes deviations equally in both directions, while the safety impact can differ substantially across deviation directions, as shown in Fig. 2b. Consequently, policies trained solely through imitation often produce behaviors that appear reasonable but may be unsafe under actual driving conditions.

To address safety concerns, some recent methods [5–7] introduce rule-based teachers and adopt multi-target distillation to regress multiple rule-driven metrics as supervision signals, as shown in Fig. 2b. While such designs improve upon pure imitation learning regarding safety, they independently learn separate scoring functions for each anchor trajectory without directly optimizing the underlying policy distribution, ultimately leading to suboptimal driving performance.

*Co-first author.
†Correponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

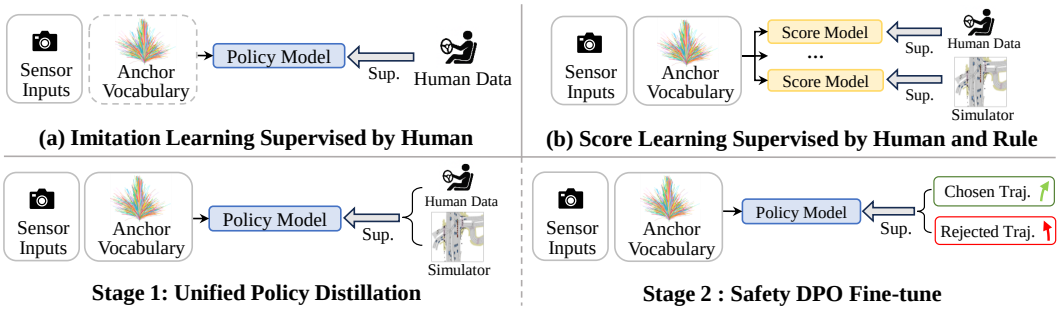

Figure 1: **Comparison of different training paradigms for end-to-end autonomous driving.** (a) The policy is trained by imitation learning. [3]. (b) The model trains multiple score heads using human and rule-based supervision signals [5, 6]. (c) Our method first pretrains the policy using a unified supervision signal that fuses human imitation similarity with rule-based safety scores, and then finetunes it via Safety DPO. Sup.: Supervision. Traj.: Trajectory.

Motivated by the safety issues of imitation learning and the indirect optimization limitation in score-based methods, we propose **DriveDPO**, a Safety Direct Preference Optimization Policy Learning framework. First, to address the challenge that score-based methods optimize per-anchor scores instead of the overall policy distribution, we propose a unified policy distillation approach that merges human imitation similarity and rule-based safety scores into a single supervisory signal for the policy model. Unlike score-based methods that construct separate score heads for each trajectory candidate, our method directly supervises the policy distribution over all anchors, enabling more coherent and end-to-end policy optimization. However, directly combining imitation and safety supervision into a single training objective formulates a multi-objective optimization problem [8–10]. To overcome this, we introduce the iterative Direct Preference Optimization framework [11] and propose Safety DPO, which reformulates the supervision as a trajectory-level preference alignment task. It enhances the policy's responsiveness to safety-oriented preferences through more stable and targeted optimization. Through preference learning on trajectory pairs, Safety DPO promotes human-like and safe trajectories over those that are human-like but unsafe, enabling more precise safety preference alignment in policy learning.

We evaluate the proposed framework on the NAVSIM benchmark [12] along with Bench2Drive benchmark [13]. Under a unified setting using a ResNet-34 perception backbone [14], our method achieves a new state-of-the-art PDMS of 90.0, surpassing the SOTA imitation-based method by 1.9 PDMS and the SOTA score-based method by 2.0 PDMS. Further qualitative results also show that our method substantially improves the safety of the learned policy in complex scenarios.

The contributions of this paper can be summarized as follows: **(1)** We identify fundamental challenges in existing imitation learning methods and score-based methods: Pure imitation learning fails to distinguish between trajectories that appear human-like but are potentially unsafe, while score-based methods decouple score prediction from direct policy optimization, resulting in suboptimal performance. **(2)** To overcome these challenges, we propose DriveDPO, a Safety DPO Policy Learning framework that first distills a unified policy distribution from human imitation and rule-based safety scores, followed by a DPO-based refinement stage for improved policy optimization. **(3)** We conduct comprehensive experiments on the NAVSIM benchmark and achieve a new state-of-the-art PDMS of 90.0, significantly advancing performance across multiple safety-critical metrics. By effectively suppressing unsafe behaviors, our method demonstrates great potential for safety-critical end-to-end autonomous driving applications.

2 Related Works

2.1 End-to-end Autonomous Driving

End-to-End Autonomous Driving [1–4, 15–23] typically maps raw sensor inputs to driving actions, either in the form of trajectories or low-level control commands, avoiding the cumulative errors and

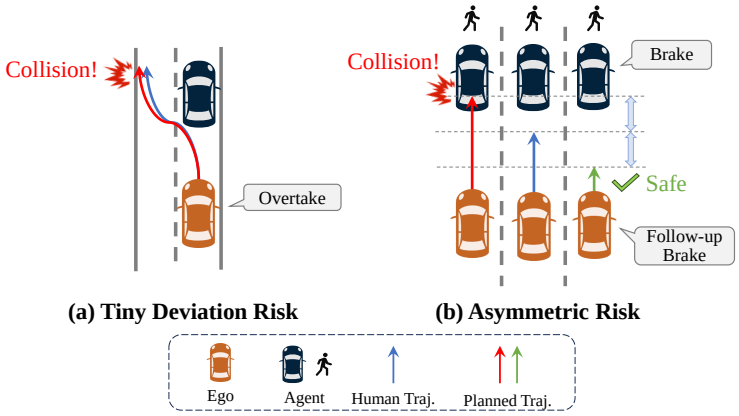

Figure 2: **Safety risks in imitation learning.** (a) In overtaking scenarios, even minor deviations from the human trajectory can lead to lane departure and collisions. However, imitation learning assigns low penalties to such trajectories due to their high similarity to human demonstrations, introducing serious safety risks. (b) In emergency braking scenarios, deviations in different directions have different safety implications: planned trajectories that overtake the human trajectory may cause rear-end collisions, while those that lag behind tend to be safe. However, imitation losses like mean squared error penalize both equally, failing to reflect the asymmetry of driving risks. Traj.: Trajectory.

interface bottlenecks inherent in traditional modular pipelines. UniAD [1] introduced a modular end-to-end architecture that enables a planning-oriented approach. VAD [2] further simplified UniAD's rasterized map representation into a vectorized format. VADv2 [3] introduced an anchor vocabulary, discretizing the continuous action space. SparseDrive [4] proposed an end-to-end method under the sparse paradigm. DiffusionDrive [24] incorporated a diffusion policy for trajectory prediction. These methods rely on imitation learning, which leads to critical safety issues. Recent works like Hydra-MDP [5] introduced a multi-teacher distillation framework that employs multiple score heads to regress imitation scores from human trajectories and rule-based metrics derived from simulator feedback. WOTE [7] employs a BEV world model to predict future BEV states of trajectories and scores them. However, these score-based methods lack direct policy distribution optimization, as they only optimize per-anchor scores independently. In contrast, we propose directly optimizing the policy distribution toward safety-aligned behavior.

2.2 Reinforcement Learning Fine-Tuning in Autonomous Driving

Reinforcement learning has gradually emerged as an important paradigm for autonomous driving research. [25] demonstrated an on-vehicle deep RL system for lane following using monocular input and distance-based reward. [26] introduced implicit affordances to enable model-free RL in urban settings with traffic light and obstacle handling. CIRL [27] combined goal-conditioned RL and human demonstration to improve success rates in CARLA. GRI [28] integrated expert data into off-policy RL for stable vision-based urban driving. Motivated by the success of reinforcement learning with human feedback (RLHF) [29–33] in large language models, an increasing number of studies have explored the paradigm of reinforcement learning finetuning in autonomous driving systems. DRIFT [34] proposed a Reward Finetuning strategy for unsupervised LiDAR object detection, where a heuristically designed reward function acts as a proxy for human feedback. BC-SAC [35] introduced using a pre-trained imitation policy as the initial policy and finetunes it in a simulated environment built from real driving data. Gen-Drive [36] proposed a behavior diffusion model that generates diverse candidate trajectories and is finetuned via reinforcement learning to favor higher-reward outputs. AlphaDrive [18] adopts a two-stage training strategy that combines supervised finetuning and reinforcement learning for planning-oriented reasoning. TrajHF [37], as a concurrent work, introduced a framework that finetunes a trajectory generator via reinforcement learning to produce trajectories more aligned with human driving style. However, TrajHF primarily focuses on driving style preference alignment without explicitly considering policy safety. In contrast, we introduce a safety RL finetuning into end-to-end autonomous driving, explicitly optimizing the policy to favor safer trajectories.

3 Preliminary

3.1 End to End Learning Task and Anchor Vocabulary

Given a raw sensor observation O, the model predicts trajectory points over the next T time steps. Each trajectory point is represented as (x_t, y_t, θ_t), where $t = 1, \ldots, T$. Traditional methods typically perform continuous trajectory regression by predicting each point in continuous space. In contrast, we adopt the Anchor Vocabulary proposed in VADv2 [3], which transforms the action space from a continuous domain $\mathbb{R}^{T\times 3}$ to a predefined discrete set of anchors $\mathcal{V} = \{a^i \in \mathbb{R}^{T\times 3}\}_{i=1}^{N}$, where the size of the Anchor Vocabulary is N, a^i denotes a trajectory consisting of T consecutive points, each with position and heading (x, y, θ). Under this formulation, the policy model π_θ assigns a probability to each anchor in the vocabulary, forming a discrete action distribution:

$$\pi_\theta(a_i) = p_i, \quad i = 1, \ldots, N, \quad \sum_{i=1}^{N} p_i = 1. \tag{1}$$

The final predicted trajectory corresponds to the anchor with the highest probability:

$$a^* = \arg\max_{a_i \in \mathcal{V}} \pi_\theta(a_i) \tag{2}$$

3.2 Iterative DPO

Direct Preference Optimization (DPO) [38] is a preference optimization framework introduced for reinforcement learning fine-tuning of large language models [39–43]. The core idea of DPO is to optimize the policy directly based on preference pairs, encouraging it to favor preferred outputs over less preferred ones. To mitigate the out-of-distribution issues, iterative DPO [11] was proposed, which introduces an intermediate Reward Model to evaluate the quality of different outputs and has achieved notable success across various domains [44–48]. Formally, given a policy π_θ and a reward model r_ϕ, for each input x the policy generates a set of N candidate trajectories $\{a_i\}_{i=1}^{N}$. The reward model assigns scalar scores $r_\phi(a_i)$ to each candidate. The sample with the highest score is selected as the chosen trajectory a_w, and the one with the lowest score is selected as the rejected trajectory a_l. The DPO loss then encourages the policy to prefer a_w over a_l relative to a fixed reference policy π_{Ref}:

$$\mathcal{L}_{\text{DPO}} = -\log \sigma \left(\beta \left(\log \frac{\pi_\theta(a_w)}{\pi_{\text{Ref}}(a_w)} - \log \frac{\pi_\theta(a_l)}{\pi_{\text{Ref}}(a_l)} \right) \right) \tag{3}$$

where $\sigma(\cdot)$ is the sigmoid function, and β is a temperature parameter.

4 Method

In this section, we propose a Safety DPO Policy Learning framework. First, we introduce the overall policy model architecture (Sec. 4.1), which includes perception encoding, anchor tokenization, and cross-attention fusion for decision-making. To enable direct policy optimization that integrates human demonstrations and rule-based safety signals, we propose Unified Policy Distillation (Sec. 4.2), which integrates imitation similarity and rule-based score into a single supervision distribution. Finally, to alleviate the optimization challenges of multi-objective supervision and further improve policy safety, we present a Safety DPO method (Sec. 4.3), which fine-tunes the policy via iterative DPO.

4.1 Policy Model Architecture

The input to our model includes multi-view camera images, LiDAR point clouds, the current ego state, and a navigation command. The model outputs a probability distribution over a predefined set of N discrete candidate trajectories, as shown in Fig. 3a. We begin by constructing a discrete set of anchor trajectories to define the output space of the policy network. Specifically, we apply k-means clustering on human driving trajectories from the NAVSIM Navtrain split [12] to obtain N representative anchor trajectories, denoted as: $\mathcal{V} = \{a_i\}_{i=1}^{N} \in \mathbb{R}^{N\times T\times 3},\ a_i \in \mathbb{R}^{T\times 3}$. Each anchor trajectory a_i consists of T future steps, with each step represented by $(x_t, y_t, \theta_t),\ t = 1, ...T$. These anchors are passed through an Anchor Tokenizer, which begins by applying NeRF-style Fourier positional encoding [49] to capture spatial structure:

$$\Gamma = \gamma(\mathcal{V}) = \left(\sin(2^0\pi\mathcal{V}), \cos(2^0\pi\mathcal{V}), \ldots, \sin(2^{L-1}\pi\mathcal{V}), \cos(2^{L-1}\pi\mathcal{V})\right) \in \mathbb{R}^{N\times T\times 6L} \tag{4}$$

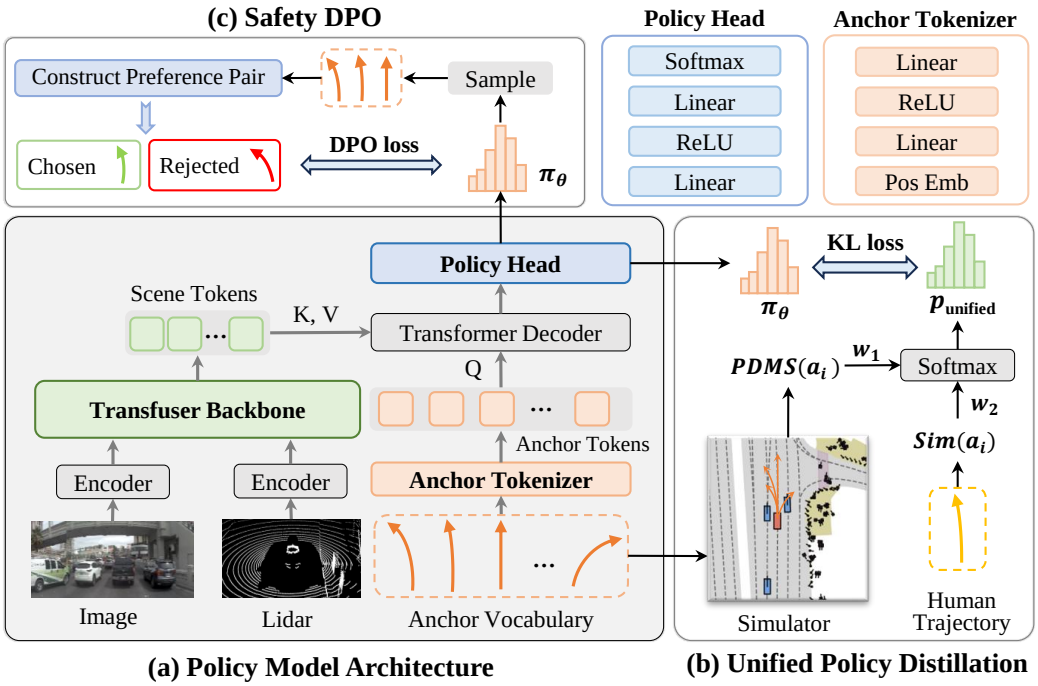

Figure 3: **Overall pipeline of the Safety DPO Policy Learning framework.** (a) Given multi-view camera images and LiDAR point clouds, a Transfuser backbone encodes the scene into a set of scene tokens. Meanwhile, a predefined Anchor Vocabulary is processed by an Anchor Tokenizer to produce anchor tokens. The anchor tokens attend to the scene tokens through a Transformer decoder to capture context-aware representations, which are then passed to the Policy Head to produce a probability distribution π_θ. (b) To provide a unified supervisory signal, we compute a distribution p_{unified} by combining human trajectory similarity and rule-based safety score. This distribution supervises the policy output via a KL divergence loss. (c) To further improve policy optimization, we introduce the iterative DPO framework. We sample K candidate trajectories from the predicted policy distribution π_θ, and construct a preference pair to fine-tune the policy via a DPO loss.

where L is the positional encoding dimension. The encoded trajectories Γ are then passed through a multi-layer perceptron (MLP) to produce the final anchor token representation:

$$E_{\text{anchor}} = \text{MLP}(\Gamma) \in \mathbb{R}^{N \times T \times d} \tag{5}$$

where d is the embedding dimension. For perception, we adopt Transfuser [50] as the backbone to fuse camera images and LiDAR point clouds. The model first encodes the image and LiDAR inputs separately and then fuses them through multiple Transformer [51] modules to obtain a unified scene representation. This results in a set of Scene Tokens: $E_{\text{scene}} \in \mathbb{R}^{M \times d}$, where M is the number of scene tokens. In the final decision-making stage, inspired by [2, 5, 6], we employ a Cross-Attention-based Transformer Decoder [51] to align anchor tokens with the scene context. Each anchor token acts as a query and attends over the scene tokens to obtain a context-enhanced representation:

$$\tilde{E}_{\text{anchor}} = \text{TransformerDecoder}(Q = E_{\text{anchor}}, K = E_{\text{scene}}, V = E_{\text{scene}}) \in \mathbb{R}^{N \times d} \tag{6}$$

where $\tilde{E}_{\text{anchor}}$ denotes the fused anchor representation enriched with scene semantics. Finally, the policy network applies an MLP to each context-enhanced anchor token $\tilde{E}_{\text{anchor}}$, followed by a softmax function to obtain the final policy distribution over anchor trajectories:

$$\pi_\theta(a_i) = \text{Softmax}(\text{MLP}(\tilde{E}_{\text{anchor}}))_i, \quad i = 1, \ldots, N \tag{7}$$

4.2 Unified Policy Distillation

Score learning methods [5–7] typically construct independent score heads for each trajectory anchor, predicting per-anchor scores rather than directly optimizing the policy distribution. To address this

issue, we propose a unified policy distillation approach that compresses all supervision signals into a single policy distribution target, enabling end-to-end consistency optimization of the policy output. Specifically, we combine human imitation similarity and a rule-based safety score to serve as a supervisory signal over the output probability distribution. For each candidate trajectory $a_i \in \mathbb{R}^{T\times 3}$ in the anchor vocabulary $\mathcal{V}$, we first define the imitation similarity between each candidate trajectory a_i and the human reference trajectory $\hat{a} \in \mathbb{R}^{T\times 3}$ as the negative Euclidean distance. To ensure comparability across different scenes, we apply a softmax function to obtain the normalized relative similarity Sim_i across all anchors:

$$\text{Sim}(a_i) = \text{Softmax}\left(-\|a_i - \hat{a}\|_2\right), \quad i = 1, \ldots, N \tag{8}$$

To obtain a rule-based safety score for each anchor trajectory, we use the high-fidelity NAVSIM simulator [12] which can perform forward simulation for each trajectory and returns multiple rule-based indicators, including No At-Fault Collision (NC), Drivable Area Compliance (DAC), Ego Progress (EP), Time-to-Collision (TTC), and Comfort (C). These metrics are then aggregated into a single scalar score known as the PDM Score (PDMS), computed as: $\text{PDMS} = \text{NC} \times \text{DAC} \times (5 \times \text{EP} + 5 \times \text{TTC} + 2 \times \text{C})/12$. We then can evaluate each anchor trajectory using the NAVSIM simulator to compute its corresponding PDMS through forward simulation:

$$\text{PDMS}(a_i) = \text{ForwardSimulation}(a_i), \quad i = 1, \ldots, N \tag{9}$$

In constructing the unified supervision distribution for policy learning, we introduce the log transformation to map the imitation similarity and the safety score from the range of $[0, 1]$ to $(-\infty, 0]$, which amplifies the differences when the values are small. As a result, if an anchor has a low safety score, its corresponding value after the transformation will be significantly lower than that of a safe anchor, effectively distinguishing safe trajectories from unsafe ones. In contrast, score-based methods regress scores for each anchor independently and cannot capture the sharp disparity between unsafe and safe trajectories. Moreover, the log transformation preserves the relative differences in imitation similarity, guiding the policy to favor those that are safe and more aligned with human behavior. Finally, we apply a softmax function to construct a soft-target distribution to introduce a competition mechanism among candidate anchors. The final unified supervision distribution p_{unified} is:

$$p_{\text{unified}}(a_i) = \text{Softmax}\left(w_1 \cdot \log(\text{Sim}(a_i)) + w_2 \cdot \log(\text{PDMS}(a_i))\right), \quad i = 1, \ldots, N \tag{10}$$

where w_1 and w_2 are weighting coefficients. We train the policy model by minimizing the KL divergence between the predicted distribution π_θ and the unified supervision distribution p_{unified}:

$$\mathcal{L}_{\text{unified}} = \text{KL}\left(p_{\text{unified}} \,\|\, \pi_\theta\right) \tag{11}$$

4.3 Safety DPO

With the pre-trained policy obtained via unified policy distillation, a multi-objective optimization problem [8–10] arises due to the joint supervision from both imitation and safety signals. To mitigate this challenge, we adopt the iterative DPO framework [11] and propose Safety Direct Preference Optimization (Safety DPO), which further refines the policy by explicitly favoring trajectories that are both human-like and safe while suppressing those that appear human-like but risky. We begin by sampling K candidate trajectories from the current policy distribution $\pi_\theta \in \mathbb{R}^N$. The naive approach selects the trajectory with the highest score in p_{unified} as the chosen sample and the one with the lowest score as the rejected sample. However, this approach results in overly simplistic preference pairs, limiting the effectiveness of preference-based optimization. To address this, we continue to use the highest-scoring trajectory as the chosen sample a_w but design two strategies for selecting the rejected sample a_l. The first selection method is Imitation-Based Rejected Trajectory Selection, which identifies trajectories that are spatially close to the human reference but exhibit poor safety performance. Specifically, it selects the rejected trajectory that is closest to the human trajectory while having a low PDMS:

$$a_l = \arg\min_{a_i} \|a_i - \hat{a}\|_2 \quad \text{s.t. } \text{PDMS}(a_i) < \tau, \quad i = 1, \ldots, K \tag{12}$$

where τ is a predefined safety threshold and $\hat{a}$ denotes the human trajectory. The second selection method is Distance-Based Rejected Trajectory Selection, which identifies unsafe candidates spatially close to the chosen trajectory. Specifically, it selects the rejected trajectory that has a low PDMS but is closest to the chosen trajectory a_w:

$$a_l = \arg\min_{a_i} \|a_i - a_w\|_2 \quad \text{s.t. } \text{PDMS}(a_i) < \tau, \quad i = 1, \ldots, K \tag{13}$$

Table 1: **Comparison of end-to-end driving methods on the NAVSIM.** The best results are denoted by **bold** and the second best results are denoted by underline. C: Camera. L: LiDAR.

Method	Supervision	Input	NC↑	DAC↑	EP↑	TTC↑	C↑	PDMS↑
PDM-closed [52]	–	GT Perception	94.6	99.8	89.9	86.9	99.9	89.1
Human	–	–	100.0	100.0	87.5	100.0	99.9	94.8
Ego Status MLP	Human	Ego State	93.0	77.3	62.8	83.6	100.0	65.6
VADv2 [3]	Human	C	97.9	91.7	77.6	92.9	100.0	83.0
UniAD [1]	Human	C	97.8	91.9	78.8	92.9	100.0	83.4
LTF [50]	Human	C	97.4	92.8	79.0	92.4	100.0	83.8
Transfuser [50]	Human	C&L	97.7	92.8	79.2	92.8	100.0	84.0
PARA-Drive [53]	Human	C	97.9	92.4	79.3	93.0	99.8	84.0
LAW [17]	Human	C	96.4	95.4	81.7	88.7	99.9	84.6
DRAMA [54]	Human	C&L	98.0	93.1	80.1	**94.8**	100.0	85.5
GoalFlow [55]	Human	C&L	98.3	93.8	79.8	94.3	100.0	85.7
ARTEMIS [56]	Human	C&L	98.3	95.1	81.4	94.3	100.0	87.0
DiffusionDrive [24]	Human	C&L	98.2	96.2	82.2	94.7	100.0	88.1
Hydra-MDP [5]	Human & Rule	C&L	98.3	96.0	78.7	94.6	100.0	86.5
Hydra-MDP++ [6]	Human & Rule	C	97.6	96.0	80.4	93.1	100.0	86.6
WOTE [7]	Human & Rule	C&L	98.4	96.6	81.7	94.5	99.9	88.0
Ours (w/o DPO)	Human & Rule	C&L	97.9	97.3	84.0	93.6	100.0	88.8
Ours (full)	Human & Rule	C&L	**98.5**	**98.1**	**84.3**	**94.8**	99.9	**90.0**

Table 2: **Closed-loop results on Bench2Drive.** The best results are in **bold**.

Method	Efficiency↑	Comfortness↑	Success Rate (%)↑	Driving Score↑
AD-MLP	48.45	22.63	0.00	18.05
UniAD	129.21	43.58	16.36	45.81
VAD	157.94	**46.01**	15.00	42.35
TCP	76.54	18.08	30.00	59.90
Ours	**166.80**	26.79	**30.62**	**62.02**

Experiments show that both methods significantly improve the safety performance of the policy compared to naive preference construction, with the first method slightly outperforming the second. Therefore, our experiments adopt the first method as the default preference pair construction strategy. Finally, given a constructed preference pair (a_w, a_l), we apply the standard DPO loss:

$$\mathcal{L}_{\text{DPO}} = -\log \sigma \left(\beta \left(\log \frac{\pi_\theta(a_w)}{\pi_{\text{Ref}}(a_w)} - \log \frac{\pi_\theta(a_l)}{\pi_{\text{Ref}}(a_l)} \right) \right) \tag{14}$$

where $\sigma(\cdot)$ is the sigmoid function, β is a temperature parameter, and π_{Ref} is the reference policy.

5 Experiments

5.1 Benchmark

NAVSIM NAVSIM [12] benchmark combines real-world sensor data with a non-interactive simulation mechanism, which is built upon OpenScene [57], a reprocessed version of the nuPlan dataset [58]. For each frame, the NAVSIM dataset provides eight high-resolution camera images and fused point cloud data sampled at 2 Hz. The model can take a history of 1.5 seconds as input and is tasked with predicting eight future waypoints over a 4-second horizon. The final standardized training set, Navtrain, contains approximately 103,000 samples, and the test set, Navtest, contains around 12,000 samples. NAVSIM introduces a simulation-based metric called the PDM Score (PDMS), which integrates No At-Fault Collision (NC), Drivable Area Compliance (DAC), Ego Progress (EP), Time-to-Collision (TTC), and Comfort (C) into a single scalar score: $\text{PDMS} = \text{NC} \times \text{DAC} \times (5 \times \text{EP} + 5 \times \text{TTC} + 2 \times \text{C})/12$.

Bench2Drive Bench2Drive [13] is a closed-loop evaluation benchmark for end-to-end autonomous driving. The official training set contains approximately 13,638 short clips, covering 44 categories of interactive scenarios, 23 weather conditions, 12 towns, and a full sensor suite. The evaluation set is organized into 220 short routes that assess various interaction capabilities under different towns and weather. The official closed-loop metrics are Driving Score (DS) and Success Rate (SR), with Driving Efficiency and Comfortness additionally reported; specifically, SR requires the vehicle to reach the destination within the time limit without traffic violations, while DS aggregates route completion and violation penalties into a weighted summary.

5.2 Implementation Details

For NAVSIM benchmark, our model is trained on the official NAVSIM [12] training set, and evaluated on the official test set Navtest. For Bench2Drive, our model is trained on base subset and evaluated on the official 220 evaluation routes. We follow the same perception setup and ResNet-34 backbone used in Transfuser for a fair comparison. Specifically, we use a concatenated front-view image of size 1024×256 formed by three forward-facing cameras as the visual input, fused with a 64×64 BEV LiDAR feature map. In addition, the model receives a state vector consisting of the current vehicle speed, acceleration, and navigation information. The size of the anchor vocabulary is set to $N = 8192$. We use fixed weights $w_1 = 0.1$ and $w_2 = 1.0$ for unified policy distillation. The number of frequency bands in the positional encoding is set to $L = 10$. The predefined safety threshold τ is set to 0.3. All experiments are conducted on 6 NVIDIA L20 GPUs, with a batch size of 16 per GPU. We use the AdamW optimizer [59] with a learning rate of $1e{-}4$. The model is first trained for 30 epochs using unified policy distillation, followed by 10 epochs of fine-tuning with Safety DPO. We sample $K = 1024$ trajectories from the policy distribution for each DPO iteration and set the $\beta = 0.1$. In DPO training, inspired by [11], we introduce an explicit KL regularization term to suppress distributional drift during training. Finally, similar to [35], we continue applying the KL loss from unified policy distillation during the DPO fine-tuning stage as an auxiliary loss.

5.3 Comparison with SOTA Methods

We conduct a comprehensive comparison of our method against representative end-to-end baselines on the NAVSIM Navtest split using ResNet-34 as the visual backbone, as shown in Table 1. our method using only unified policy distillation without DPO fine-tuning (Ours w/o DPO) already achieves a PDMS of 88.8, outperforming the current SOTA imitation learning method DiffusionDrive [24] (88.1) and the SOTA score learning method WOTE [7] (88.0). This demonstrates that our unified policy distillation method brings significant performance gains. After applying DPO fine-tuning, our model further improves safety-related metrics: NC increases by 0.6, DAC improves by 0.8, and TTC achieves a gain of 1.2. These results indicate a clear improvement in the safety and reliability of our policy across diverse scenarios. Our method ultimately achieves a new state-of-the-art PDMS of 90.0, outperforming the SOTA imitation learning method [24] by 1.9 and the SOTA score-based method [7] by 2.0. It also surpasses the PDM-closed method [52], which takes the privileged GT Perception. We also conduct closed-loop evaluation on Bench2Drive (Table 2). Our method outperforms representative baselines on Driving Score and Success Rate, demonstrating its effectiveness in closed-loop settings.

5.4 Ablation Study

Ablation on Unified Policy Distillation. Table 3 conducts the ablation study to evaluate the effectiveness of the proposed Unified Policy Distillation method. First, compared to the Score Learning method (ID-2), Unified Policy Distillation (ID-1) significantly improves overall policy performance. Furthermore, to assess the importance of combining both imitation and rule-based signals, we conduct additional experiments using single-source supervision. Results show that the Imitation-only method (ID-3) lacks the ability to distinguish high-risk trajectories and performs poorly on critical safety metrics such as DAC, resulting in poor performance. On the other hand, the Rule-only method (ID-4), which ignores human intent, tends to produce aggressive and unreasonable behaviors, leading to suboptimal results in key safety metrics such as NC and TTC.

Ablation on DPO and Rejected Trajectory Selection. Table 4 conducts the ablation study to verify the effectiveness of DPO and the proposed rejected trajectory selection method. Applying

Table 3: **Ablation on Unified Policy Distillation.** Score indicates we regress scores independently for each trajectory anchor. Policy indicates we optimize the policy distribution over all anchors. Sup.: Supervision. The best results are denoted by **bold**.

ID	Score	Policy	Human Sup.	Rule Sup.	NC↑	DAC↑	EP↑	TTC↑	PDMS↑
1	✗	✓	✓	✓	**97.9**	**97.3**	84.0	**93.6**	**88.8**
2	✓	✗	✓	✓	97.8	96.0	81.6	93.7	87.3
3	✗	✓	✓	✗	97.6	91.2	77.5	92.9	82.5
4	✗	✓	✗	✓	95.3	97.3	**85.9**	88.7	87.2

Table 4: **Ablation on DPO and Rejected Trajectory Selection.** The best results are denoted by **bold**.

Method	NC↑	DAC↑	EP↑	TTC↑	C↑	PDMS↑
w/o DPO	97.9	97.3	84.0	93.6	100.0	88.8
vanilla Selection	98.4	97.5	83.5	94.6	100.0	89.3
Distance-Based Selection	98.1	**98.2**	**84.3**	94.2	99.9	89.7
Imitation-Based Selection	**98.5**	98.1	**84.3**	**94.8**	99.9	**90.0**

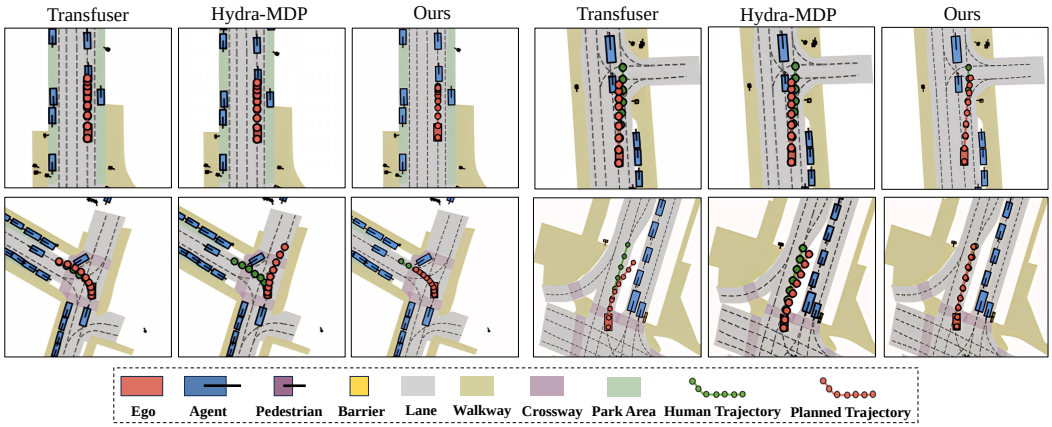

Figure 4: **Qualitative comparison with baselines.** We compare our method against Transfuser [50] and Hydra-MDP [5]. Both the Transfuser and Hydra-MDP suffer from collisions or off-road behaviors in complex road structures, fine-grained turning, and challenging emergency braking tasks. In contrast, our method consistently generates safer trajectories.

the vanilla selection method, which uses the highest and lowest score in p_{unified} as a preference pair, already improves PDMS by 0.5, indicating that preference-based optimization can effectively suppress low-quality outputs and boost overall performance. Compared to the vanilla selection method, the Distance-Based strategy yields an additional 0.9 improvement in PDMS, and the Imitation-Based strategy improves PDMS by 1.2. This indicates that these strategies further enhance the safety performance by more effectively identifying and suppressing human-like but risky trajectories.

5.5 Qualitative Comparison

Qualitative Comparison with Baselines. Figure 4 presents a qualitative comparison between our method and two baselines: Transfuser [50], representing imitation learning, and Hydra-MDP [5], representing score-based learning. While Transfuser and Hydra-MDP tend to generate unsafe behaviors such as lane departures or collisions in complex scenarios, our method consistently maintains safe and compliant decisions under diverse conditions.

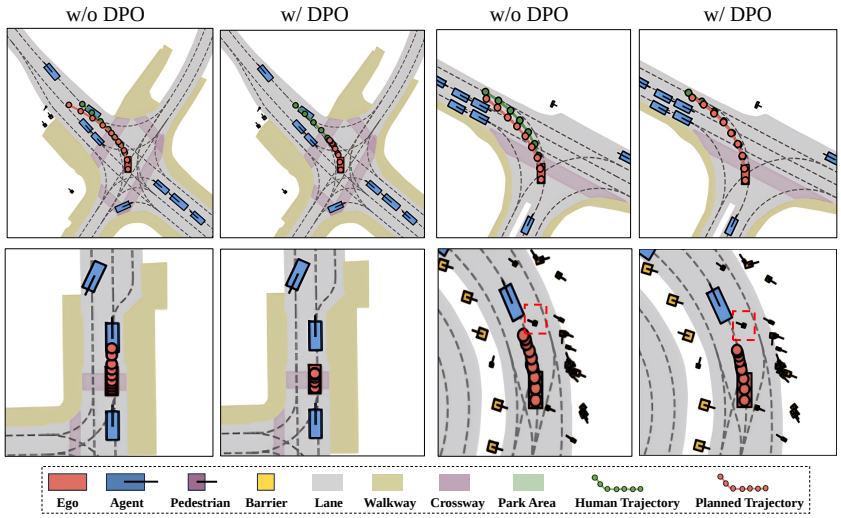

Figure 5: **Qualitative visualization of DPO-enhanced behaviors across diverse scenarios.** The DPO-enhanced policy exhibits more cautious behavior such as early deceleration and smoother cornering, and shows improved responsiveness in challenging scenarios like sudden braking of lead vehicles and pedestrian crossings.

Qualitative Comparison between Policies with and without DPO. To intuitively demonstrate the impact of DPO fine-tuning on policy behavior, we conduct a qualitative analysis of several challenging scenarios as illustrated in Figure 5. The DPO-enhanced policy favors safer and more conservative decisions in complex interaction scenarios, verifying that the DPO fine-tuning improves safety awareness across diverse conditions.

6 Limitations and Future Works

Despite the significant progress our method has made in enhancing the safety of end-to-end driving policies, several limitations warrant further study. First, our approach relies on the PDMS metric as the core criterion for safety evaluation. Although PDMS integrates multiple dimensions, including collision avoidance, drivable area compliance, ego progress, time-to-collision, and comfort, it remains a predefined weighted composite metric. As such, it cannot fully capture all potential risk factors in complex driving scenarios. Future work may explore more expressive and flexible trajectory evaluation metrics to build a more comprehensive safety assessment mechanism. Second, our rule-based supervision depends on high-fidelity simulators to provide rule-driven evaluation scores. While effective, the preferences derived from such simulation are inherently limited by the rules' design and the simulator's precision. Moreover, access to such high-fidelity simulators is scarce, constraining the scale and diversity of available data. Therefore, future research should explore preference optimization methods that do not rely on ground-truth perception labels by automatically mining latent preference structures from historical trajectory data. Such efforts could facilitate the development of weakly-supervised or even fully self-supervised safety alignment strategies.

7 Conclusion

In this paper, we identify key safety challenges in imitation learning for end-to-end autonomous driving, where models often produce trajectories that appear human-like but are potentially unsafe. In addition, we highlight the limitations of existing score-based methods, which decouple supervision from policy learning and fail to provide direct optimization of the policy distribution. To tackle these challenges, we propose DriveDPO, a Safety DPO Policy Learning framework. By combining unified policy distillation with Safety DPO fine-tuning, DriveDPO enables direct and effective optimization of the policy distribution. Comprehensive experiments and qualitative comparisons on the NAVSIM dataset demonstrate that DriveDPO significantly improves safety and compliance, showing its potential for deployment in safety-critical driving systems.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No. 62320106010, No. U21B2042).

References

[1] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023.

[2] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023.

[3] Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024.

[4] Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. Sparsedrive: End-to-end autonomous driving via sparse scene representation. *arXiv preprint arXiv:2405.19620*, 2024.

[5] Zhenxin Li, Kailin Li, Shihao Wang, Shiyi Lan, Zhiding Yu, Yishen Ji, Zhiqi Li, Ziyue Zhu, Jan Kautz, Zuxuan Wu, et al. Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation. *arXiv preprint arXiv:2406.06978*, 2024.

[6] Kailin Li, Zhenxin Li, Shiyi Lan, Yuan Xie, Zhizhong Zhang, Jiayi Liu, Zuxuan Wu, Zhiding Yu, and Jose M Alvarez. Hydra-mdp++: Advancing end-to-end driving via expert-guided hydra-distillation. *arXiv preprint arXiv:2503.12820*, 2025.

[7] Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via bev world model. *arXiv preprint arXiv:2504.01941*, 2025.

[8] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in neural information processing systems*, volume 31, 2018.

[9] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in neural information processing systems*, volume 33, pages 5824–5836, 2020.

[10] Weiyu Chen, Xiaoyuan Zhang, Baijiong Lin, Xi Lin, Han Zhao, Qingfu Zhang, and James T Kwok. Gradient-based multi-objective deep learning: Algorithms, theories, applications, and beyond. *arXiv preprint arXiv:2501.10945*, 2025.

[11] Xiong Wei, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. Iterative preference learning from human feedback: bridging theory and practice for rlhf under kl-constraint. In *Proceedings of the 41st International Conference on Machine Learning*, pages 54715–54754, 2024.

[12] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, et al. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. In *Advances in Neural Information Processing Systems*, volume 37, pages 28706–28719, 2024.

[13] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. *Advances in Neural Information Processing Systems*, 37:819–844, 2024.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Abbas Sadat, Sergio Casas, Mengye Ren, Xinyu Wu, Pranaab Dhawan, and Raquel Urtasun. Perceive, predict, and plan: Safe motion planning through interpretable semantic representations. In *European Conference on Computer Vision*, pages 414–430. Springer, 2020.

[16] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022.

[17] Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. *arXiv preprint arXiv:2406.08481*, 2024.

[18] Bo Jiang, Shaoyu Chen, Qian Zhang, Wenyu Liu, and Xinggang Wang. Alphadrive: Unleashing the power of vlms in autonomous driving via reinforcement learning and reasoning. *arXiv preprint arXiv:2503.07608*, 2025.

[19] Chonghao Sima, Kashyap Chitta, Zhiding Yu, Shiyi Lan, Ping Luo, Andreas Geiger, Hongyang Li, and Jose M Alvarez. Centaur: Robust end-to-end autonomous driving with test-time training. *arXiv preprint arXiv:2503.11650*, 2025.

[20] Yuntao Chen, Yuqi Wang, and Zhaoxiang Zhang. Drivinggpt: Unifying driving world modeling and planning with multi-modal autoregressive transformers. *arXiv preprint arXiv:2412.18607*, 2024.

[21] Xiaosong Jia, Junqi You, Zhiyuan Zhang, and Junchi Yan. Drivetransformer: Unified transformer for scalable end-to-end autonomous driving. *arXiv preprint arXiv:2503.07656*, 2025.

[22] Ziying Song, Caiyan Jia, Lin Liu, Hongyu Pan, Yongchang Zhang, Junming Wang, Xingyu Zhang, Shaoqing Xu, Lei Yang, and Yadan Luo. Don't shake the wheel: Momentum-aware planning in end-to-end autonomous driving. *arXiv preprint arXiv:2503.03125*, 2025.

[23] Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, et al. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. *arXiv preprint arXiv:2506.08052*, 2025.

[24] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *arXiv preprint arXiv:2411.15139*, 2024.

[25] Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *2019 international conference on robotics and automation (ICRA)*, pages 8248–8254. IEEE, 2019.

[26] Marin Toromanoff, Emilie Wirbel, and Fabien Moutarde. End-to-end model-free reinforcement learning for urban driving using implicit affordances. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7153–7162, 2020.

[27] Xiaodan Liang, Tairui Wang, Luona Yang, and Eric Xing. Cirl: Controllable imitative reinforcement learning for vision-based self-driving. In *Proceedings of the European conference on computer vision (ECCV)*, pages 584–599, 2018.

[28] Raphael Chekroun, Marin Toromanoff, Sascha Hornauer, and Fabien Moutarde. Gri: General reinforced imitation and its application to vision-based autonomous driving. *Robotics*, 12(5):127, 2023.

[29] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in neural information processing systems*, pages 27730–27744, 2022.

[30] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in neural information processing systems*, volume 30, 2017.

[31] Timm Teubner, Christoph M Flath, Christof Weinhardt, Wil Van Der Aalst, and Oliver Hinz. Welcome to the era of chatgpt et al. the prospects of large language models. *Business & Information Systems Engineering*, 65(2):95–101, 2023.

[32] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Ren Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, et al. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. In *Forty-first International Conference on Machine Learning*, 2024.

[33] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[34] Katie Luo, Zhenzhen Liu, Xiangyu Chen, Yurong You, Sagie Benaim, Cheng Perng Phoo, Mark Campbell, Wen Sun, Bharath Hariharan, and Kilian Q Weinberger. Reward finetuning for faster and more accurate unsupervised object discovery. In *Advances in Neural Information Processing Systems*, volume 36, pages 13250–13266, 2023.

[35] Yiren Lu, Justin Fu, George Tucker, Xinlei Pan, Eli Bronstein, Rebecca Roelofs, Benjamin Sapp, Brandyn White, Aleksandra Faust, Shimon Whiteson, et al. Imitation is not enough: Robustifying imitation with reinforcement learning for challenging driving scenarios. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7553–7560. IEEE, 2023.

[36] Zhiyu Huang, Xinshuo Weng, Maximilian Igl, Yuxiao Chen, Yulong Cao, Boris Ivanovic, Marco Pavone, and Chen Lv. Gen-drive: Enhancing diffusion generative driving policies with reward modeling and reinforcement learning fine-tuning. *arXiv preprint arXiv:2410.05582*, 2024.

[37] Derun Li, Jianwei Ren, Yue Wang, Xin Wen, Pengxiang Li, Leimeng Xu, Kun Zhan, Zhongpu Xia, Peng Jia, Xianpeng Lang, et al. Finetuning generative trajectory model with reinforcement learning from human feedback. *arXiv preprint arXiv:2503.10434*, 2025.

[38] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741, 2023.

[39] Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR, 2024.

[40] Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is dpo superior to ppo for llm alignment? a comprehensive study. *arXiv preprint arXiv:2404.10719*, 2024.

[41] Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems*, volume 37, pages 124198–124235, 2024.

[42] Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *arXiv preprint arXiv:2401.08417*, 2024.

[43] Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

[44] Jie Liu, Zhanhui Zhou, Jiaheng Liu, Xingyuan Bu, Chao Yang, Han-Sen Zhong, and Wanli Ouyang. Iterative length-regularized direct preference optimization: A case study on improving 7b language models to gpt-4 level. *arXiv preprint arXiv:2406.11817*, 2024.

[45] Songjun Tu, Jiahao Lin, Xiangyu Tian, Qichao Zhang, Linjing Li, Yuqian Fu, Nan Xu, Wei He, Xiangyuan Lan, Dongmei Jiang, et al. Enhancing llm reasoning with iterative dpo: A comprehensive empirical investigation. *arXiv preprint arXiv:2503.12854*, 2025.

[46] Gong Xudong, Feng Dawei, Kele Xu, Yuanzhao Zhai, Chengkang Yao, Weijia Wang, Bo Ding, and Huaimin Wang. Iterative regularized policy optimization with imperfect demonstrations. In *Forty-first International Conference on Machine Learning*, 2024.

[47] Daechul Ahn, Yura Choi, San Kim, Youngjae Yu, Dongyeop Kang, and Jonghyun Choi. Isr-dpo: Aligning large multimodal models for videos by iterative self-retrospective dpo. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1728–1736, 2025.

[48] Geon-Hyeong Kim, Youngsoo Jang, Yu Jin Kim, Byoungjip Kim, Honglak Lee, Kyunghoon Bae, and Moontae Lee. Safedpo: A simple approach to direct preference optimization with enhanced safety. *OpenReview*, 2024. Preprint.

[49] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[50] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12878–12895, 2022.

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, volume 30, 2017.

[52] Daniel Dauner, Marcel Hallgarten, Andreas Geiger, and Kashyap Chitta. Parting with misconceptions about learning-based vehicle motion planning. In *Conference on Robot Learning*, pages 1268–1281. PMLR, 2023.

[53] Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. Para-drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15449–15458, 2024.

[54] Chengran Yuan, Zhanqi Zhang, Jiawei Sun, Shuo Sun, Zefan Huang, Christina Dao Wen Lee, Dongen Li, Yuhang Han, Anthony Wong, Keng Peng Tee, et al. Drama: An efficient end-to-end motion planner for autonomous driving with mamba. *arXiv preprint arXiv:2408.03601*, 2024.

[55] Zebin Xing, Xingyu Zhang, Yang Hu, Bo Jiang, Tong He, Qian Zhang, Xiaoxiao Long, and Wei Yin. Goalflow: Goal-driven flow matching for multimodal trajectories generation in end-to-end autonomous driving. *arXiv preprint arXiv:2503.05689*, 2025.

[56] Renju Feng, Ning Xi, Duanfeng Chu, Rukang Wang, Zejian Deng, Anzheng Wang, Liping Lu, Jinxiang Wang, and Yanjun Huang. Artemis: Autoregressive end-to-end trajectory planning with mixture of experts for autonomous driving. *arXiv preprint arXiv:2504.19580*, 2025.

[57] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 815–824, 2023.

[58] Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, et al. Towards learning-based planning: The nuplan benchmark for real-world autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 629–636. IEEE, 2024.

[59] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the abstract and introduction, we accurately claim the contributions made in policy learning for end-to-end autonomous driving.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In the supplementary material, we provide a detailed analysis of the limitations of our proposed method.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In Sec. 4, we provide a detailed description of the proposed method, and additional Implementation Details are included in the supplementary material to support reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code will be open-sourced upon paper acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide detailed descriptions of the experimental settings in Sec. 5 and in supplementary material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to computational expense, error bars are not formally reported; however, we fix the random seed during all experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide details of the computational resources used in the supplementary material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research presented in this paper fully conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: The proposed method has no direct societal impact, but we hope it can inspire further research toward safer end-to-end autonomous driving.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The proposed method poses no such risks, so no specific safeguards are necessary.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets used in this paper have been properly cited and their usage complies with the respective licenses and terms of use.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method in this paper does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.