NoRD: A Data-Efficient Vision-Language-Action Model that Drives without Reasoning

Supplementary Material

7. Comparison between GRPO and Dr. GRPO

We present a component-wise breakdown of Tab. 1 in Tab. 4. Except for Ego Progress, Dr. GRPO significantly outperforms GRPO. As shown in the training and validation curves in Fig. 11, both GRPO and Dr. GRPO improve over time; however, GRPO consistently lags behind Dr. GRPO. To further illustrate this, we visualize the change in mean PDM scores of the group, relative to the SFT model (step 0), across different variance groups in Fig. 10. The variance groups are defined based on intra-group tertiles. Our analysis reveals that:

1. **Low-variance samples** (Fig. 10 (a)): GRPO exhibits higher density above the $y = x$ line, particularly for initial scores in $[0.8, 1.0]$.
2. **Medium- and high-variance samples** (Fig. 10 (b,c)): Dr. GRPO outperforms GRPO, with a denser concentration above the $y = x$ line. The performance gap widens for high-variance samples, consistent with our observation that GRPO attenuates policy updates for such samples.

8. Detailed Results

8.1. Prompt Example

We show an illustrative example in Fig. 12. NoRD maintains token and inference efficiency by directly predicting the trajectory tokens.

8.2. Waymo E2E Scores

We present the detailed results of the performance of NoRD on WaymoE2E test set in Tab. 6. As is evident, NoRD is capable of performing complex multi-lane switching maneuvers, while also performing well in less-represented scenes, such as intersections and construction sites.

8.3. Effect of Vocabulary Size

We experimented with a smaller k-disc vocabulary, consisting of 512 trajectory tokens (as compared to 2048 trajectory tokens in NoRD) and found that the performance on NAVSIM degrades (Tab. 5). This is perhaps because the smaller vocabulary size cannot represent complex maneuvers like sharp turn faithfully.

9. Reward Functions

In this section, we elaborate on the reward functions used for RL post-training. The reward consists of length reward, format reward and dataset-specific reward (PDM score for NAVSIM and Normalized RFS for WaymoE2E). The output of the model is a string of action tokens like `TRAJ_0242 TRAJ_150 TRAJ_172` that are decoded to a list of waypoints of tuples `[x,y,yaw]` at 10 Hz.

Format Reward (r_f)**:** A binary reward taking values in $\{0,\ 0.25\}$. A reward of 0.25 is assigned if the prediction consists of valid space-separated trajectory tokens of the form `TRAJ_i`, where i is a zero-padded 4-digit integer in $[0, 2047]$; otherwise the reward is 0.

Length Reward (r_l)**:** A binary reward taking values in $\{0,\ 0.25\}$. The model receives a reward of 0.25 if the prediction contains the correct number of trajectory tokens (8 for NAVSIM and 10 for WaymoE2E); otherwise, the reward is 0.

Dataset Specific Reward (r_d):

1. **PDM Score for NAVSIM**: The PDM score (range: $[0, 1]$) comprehensively measures the driving quality and safety. Is it given by:

$$\text{PDM Score} = \text{NC} \times \text{DAC} \times \frac{5 \cdot \text{TTC} + 2 \cdot \text{C} + 5 \cdot \text{EP}}{12}$$

   where, No at-fault Collision (NC), Drivable Area Compliance (DAC), Ego Progress (EP), Comfort (C), and Time-to-Collision (TTC) are all within $[0, 1]$.
2. **Normalized RFS for WaymoE2E**: The RFS quantifies the alignment of the model's predicted trajectory $\hat{T}$ with a set of three pre-rated human trajectories T_r. A score $s_r \in [3, 10]$ is assigned to each rater trajectory based on whether $\hat{T}$ falls within a *trust region* defined by dynamic longitudinal $\bar{\tau}_{\text{lng}}$ and lateral $\bar{\tau}_{\text{lat}}$ thresholds (scaled by current velocity). The final score is $\max_r(s_r)$, averaged over $t \in \{3, 5\}$ seconds, and clipped to $\min(\cdot, 4)$. The Normlized RFS, with range $[0, 1]$ is then given by:

$$\text{Normalized RFS} = \frac{\max(\max_r(s_r), 4) - 4}{6}$$

The overall reward r for the predicted trajectory is therefore given as:

$$r = \frac{r_f + r_l + r_d}{1.5}$$

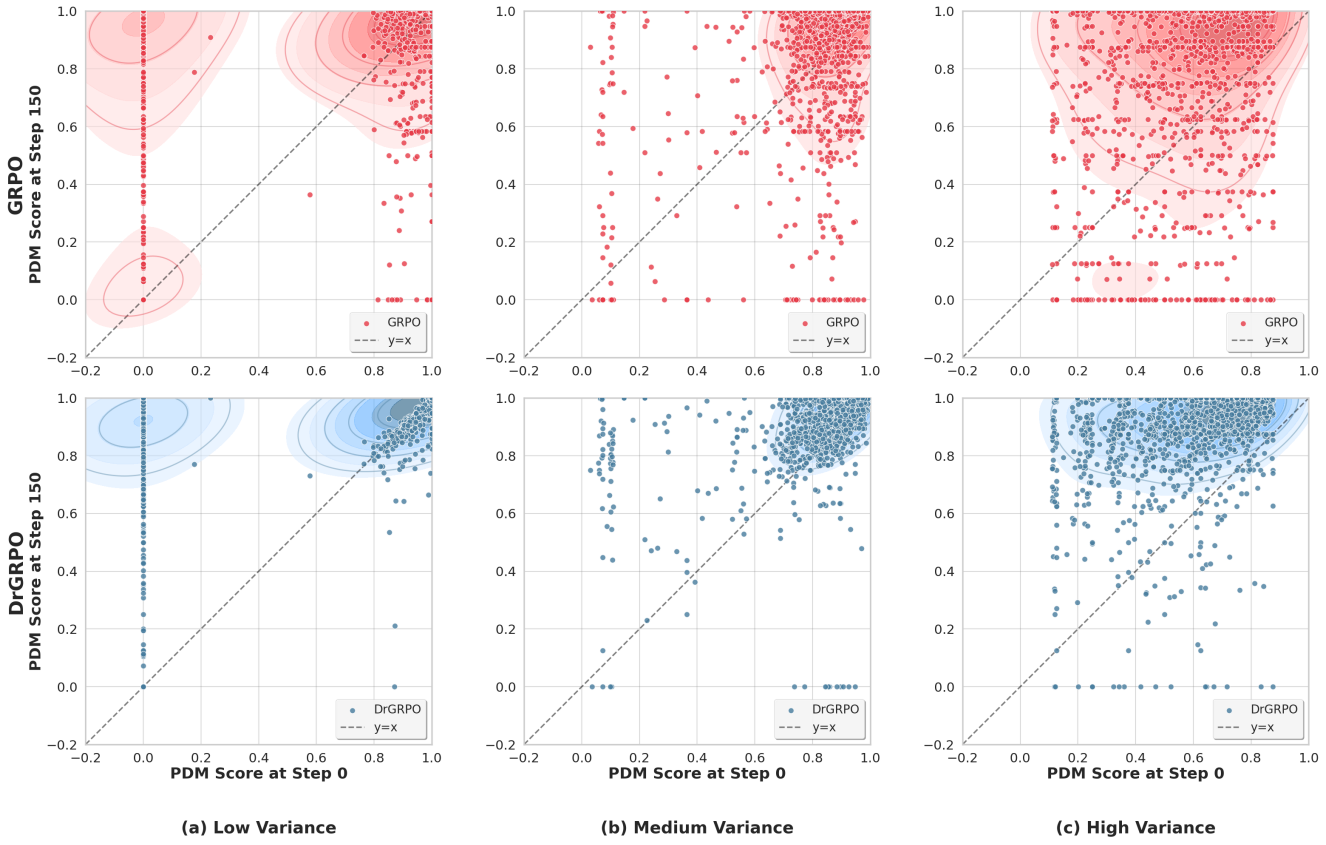

Figure 10. **Training improvement patterns for GRPO (top, red) and Dr. GRPO (bottom, blue) across intra-group variance levels.** The $y = x$ line indicates no change in PDM score. GRPO shows strong improvements for low-variance samples with initial scores in $[0.8, 1.0]$ (panel (a)), while Dr. GRPO outperforms GRPO for medium- and high-variance samples (panels (b) and (c)), with denser concentration above $y = x$.

Table 4. Detailed comparison of RL-fine-tuning of NORD-BASE with GRPO and Dr. GRPO. Dr. GRPO based RL fine-tuning is almost always better than GRPO.

Method	**PDMS↑**	**Collision↑**	**DAC↑**	**Direction↑**	**Progress↑**	**TTC↑**	**Comfort↑**
NORD-BASE	76.66	96.45	86.37	94.62	71.58	90.37	99.97
NORD-BASE+GRPO	77.18	91.89	90.12	91.84	**80.06**	80.13	99.96
NORD-BASE+Dr. GRPO	**85.62**	**97.56**	**94.92**	**95.94**	79.30	**93.53**	**100**

Table 5. Effect of k-disc vocabulary size on the performance of NORD on navtest.

Vocabulary Size	**PDMS ↑**
512	83.07
2048	85.62

10. Dataset Details

10.1. WaymoE2E

Supervised Finetuning: We curated the SFT dataset from the official WaymoE2E training set. Frames were first strictly filtered, retaining only those that guaranteed four preceding time steps were available for consistent extraction of the ego-vehicle's historical states. The final subset was then created by uniformly sampling 20% of these valid frame sequences from all contexts. This dataset was then randomly split into training and validation sets using an $85/15$ ratio. The input images were resized to

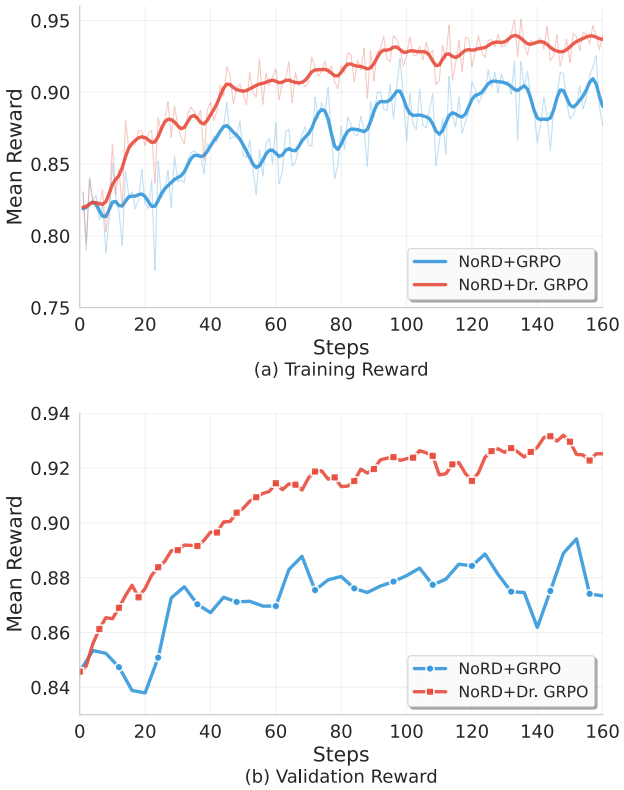

Figure 11. **Training and validation curves for RL fine-tuning with GRPO and Dr.GRPO.** Dr.GRPO (in red) consistently outperforms GRPO (in blue) on the (a) training and (b) validation sets by a significant margin.

Table 6. Detailed results on WaymoE2E Test Set.

Metric Name	**Value↑**
Construction Score	8.072616
Intersection Score	7.9252014
Pedestrian Score	7.7775736
Cyclist Score	7.8055406
Multi Lane Maneuver Score	7.8262477
Single Lane Maneuver Score	8.308635
Cut In Score	7.734755
Foreign Object Debris Score	7.6988134
Special Vehicle Score	7.7961473
Spotlight Score	6.5309787
Others Score	7.322814
ADE at 3 seconds	1.250462
ADE at 5 seconds	2.8928785
Average Score	7.709029

ensure the total number of pixels lies between 784 and 401,408, following the Qwen vision encoder's constraints.

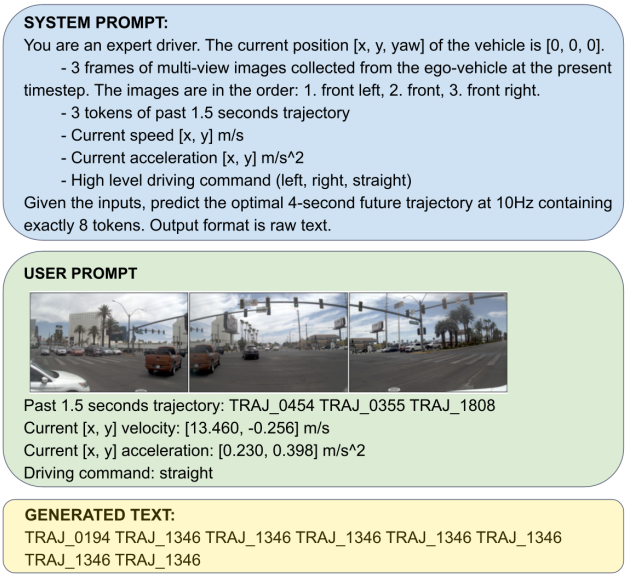

Figure 12. **Example of NoRD inference.** Given multi-view images, past trajectory, and the current velocity, acceleration, and driving command, NoRD directly predicts the trajectory tokens without explicit reasoning.

RL Finetuning: We use the official WaymoE2E validation set, for which preference annotations are provided for a single frame per scenario. Consequently, we extract one sample per scenario and randomly split the resulting set into training and validation sets using an $85/15$ ratio.

10.2. NAVSIM

Supervised Finetuning: We use the official NAVSIM's training set (`navtrain`) and split it into training and validation sets for SFT using an $80/20$ ratio. The input images were resized to ensure the total number of pixels lies between 784 and 401,408, following the Qwen vision encoder's constraints.

RL Finetuning: We construct a RLFT dataset from the NAVSIM validation split originally used for supervised fine-tuning. To remove trivial driving behaviors, we filter trajectories using a constant-velocity baseline and discard samples with a final-point displacement error below 0.2 m. For turning maneuvers, we additionally enforce a minimum average heading change of 0.01 rad per timestep to eliminate mild curvature and drift. Straight trajectories are exempt from the heading filter and are filtered solely using the displacement criterion. After filtering, the remaining samples are balanced across three driving intents—straight, left, and right—by uniformly subsampling each class. The resulting dataset contains only non-trivial and dynamically diverse trajectories, providing a more rigorous training signal for reinforcement learning-based trajectory prediction

and decision-making models.

11. Implementation Details

11.1. Supervised Finetuning

We perform supervised fine-tuning of NoRD on the NAVSIM and WaymoE2E datasets using the Qwen2.5-VL-3B-Instruct backbone, adapted to predict discretized trajectory tokens from multi-view images, past trajectories, and the ego-vehicle's current kinematic states. For NAVSIM, inputs consist of three camera frames (Front-Left, Front, Front-Right), three past trajectory tokens covering the previous 1.5 seconds, current velocity and acceleration, and a high-level driving command, with the model predicting 8 future trajectory tokens over a 4-second horizon at 10Hz. For WaymoE2E, inputs include six past trajectory tokens spanning 3 seconds, and the model predicts 10 future tokens over a 5-second horizon. In both cases, trajectory tokens are incorporated into the model vocabulary. All components of the model, including the vision encoder, multimodal MLP, and language model, are fine-tuned using mixed-precision training with bf16 and gradient checkpointing to reduce memory footprint. We train the model across 16 A100 GPUs, applying DeepSpeed ZeRO Stage 3 optimization for WaymoE2E and standard distributed training for NAVSIM. We use consistent hyperparameters across datasets, including a learning rate of 5×10^{-5}, a batch size of 8 per device with 4 gradient accumulation steps, a cosine learning rate scheduler, a warmup ratio of 0.03, and gradient clipping at 1. We evaluate the model every 50 steps on the validation sets and select the best model based on minimum evaluation loss.

11.2. RL Finetuning

We perform RL fine-tuning of NoRD using Dr. GRPO to optimize task-specific rewards. We generate 8 rollouts per input to estimate group-relative advantages and update the policy accordingly. We use a batch size of 128 trajectories for NAVSIM and 256 trajectories for WaymoE2E, applying asymmetric clipping with a high clip of 0.1 and a low clip of -0.2 to stabilize policy updates. We train across 32 A100 GPUs for WaymoE2E and 30 A100 GPUs for NAVSIM, leveraging mixed-precision and gradient checkpointing for memory efficiency. We periodically evaluate the policy on validation sets and retain the checkpoint achieving the highest reward.

12. Dataset Scale Estimation

To visualize the performance-efficiency frontier in Fig. 6, we estimated the total number of training samples for all evaluated models based on their reported configurations. Across both NAVSIM and WaymoE2E, baseline methods frequently employ complex multi-dataset mixtures or utilize varying fractions of the available data. To standardize these counts, we explicitly aggregated the reported dataset percentages and official splits detailed in the respective papers' training sections. For example, on WaymoE2E, we calculate HMVLM and DiffusionLTF at approx. 500k and 730k samples based on the train and val splits in the Waymo Open Dataset for end-to-end driving and perception. Similarly, for Poutine and AutoVLA, we aggregate their reported multi-dataset percentages to approx. 700k and 210k samples, respectively. These standardizations ensure a fair relative comparison of data efficiency on the x-axis.

Figure 13. **Failure cases of NoRD.** The predicted trajectory is shown in red and the violations marked in red circle.

13. Failure Cases

While NoRD achieves strong performance, it remains susceptible to failure in certain scenarios. We present representative examples in Fig. 13. These cases can be attributed, in part, to the fact that Dr. GRPO remains susceptible to difficulty bias, which still affects the policy optimization dynamics. We therefore believe that targeted interventions to better account for task difficulty could further push the performance frontier.