TAIL-MDP: MULTI-DIMENSIONAL LONG-TAIL LEARNING WITH PROMPT-GUIDED DECODING FOR TRAJECTORY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Trajectory prediction datasets typically exhibit pronounced long-tail distributions, causing trained models to underperform in rare yet safety-critical scenarios. Existing long-tail learning methods mainly characterize long-tailedness from a single dimension, making them insufficient for driving scenarios with multi-dimensional tail properties. Moreover, they primarily focus on representation refinement while overlooking the decoding stage, leaving encoded tail cues insufficiently exploited for trajectory generation. To this end, we propose Tail-MDP, a multi-dimensional long-tail learning framework with tail-aware decoding for trajectory prediction. We first reformulate tail sample identification as an anomaly detection problem with respect to head-dominant patterns, and use autoencoder reconstruction errors as a unified criterion to quantify long-tailedness across three key dimensions affecting future motion—motion dynamics, social interactions, and topological constraints. Here, the autoencoders are refined through training sample selection and sparse reconstruction to improve estimation reliability. Based on the resulting multi-dimensional tailness estimates, we introduce dimension-wise contrastive learning to learn representations that capture diverse tail characteristics. Furthermore, we develop a prompt-guided decoding mechanism that generates prompts according to the sample's tailness degree and injects them into the decoder via prefix tuning, thereby guiding the model to focus on tail cues during trajectory generation. Experiments on Argoverse2 and nuScenes show that Tail-MDP consistently improves performance on diverse tail scenarios, outperforms SOTA long-tail methods, and generalizes across competitive baselines.

1 INTRODUCTION

Learning-based trajectory prediction methods have achieved substantial progress on major benchmarks Cheng et al. (2023); Zhang et al. (2024a). However, prevalent datasets exhibit pronounced long-tailed distributions, where routine scenarios such as cruising dominate, whereas safety-critical cases, such as collision avoidance, remain scarce Makansi et al. (2021). This imbalance inevitably biases the learning process toward frequent head scenarios, causing tail representations to be mixed up with the head ones and degrading performance in tail scenarios. Therefore, a principled framework for long-tail identification and adaptation in trajectory prediction is urgently needed.

For long-tail identification, most methods characterize tail scenarios from a single dimension Makansi et al. (2021); Wang et al. (2023), for example, identifying interaction-related tail samples based on Kalman filtering prediction errors. However, long-tailedness in trajectory prediction manifests across heterogeneous dimensions Thuremella & Kunze (2023), such as motion characteristics and interaction patterns, making such single-dimensional criteria prone to misclassifying tail samples as head ones or conflating semantically distinct tail scenarios. Recently, SAIL Rao et al. (2026) attempts to characterize long-tailedness across multiple dimensions using manually designed attribute-specific metrics, such as TTC and jerk. However, manually specified metrics may introduce characterization bias, while attribute-specific formulations limit extensibility across dimensions.

For long-tail adaptation, most studies primarily employ contrastive learning to enhance the discriminability of tail representations. However, they often overlook the decoding stage, leaving the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

decoder without an explicit mechanism to exploit tail-specific cues during trajectory generation and potentially biasing predictions toward head-dominant behaviors. Although MoE Coden et al. (2024); Ganeshaaraj et al. (2025) has recently been introduced for tail-adaptive decoding, the tail experts may be prone to overfitting due to the limited availability of tail samples.

To this end, we propose **Tail-MDP**, which uses a unified criterion to identify tail properties across several key heterogeneous dimensions and integrates dimension-wise contrastive learning with prompt-guided decoding for adaptive handling of tail scenarios, as shown in Fig. 1.

Specifically, tail driving scenarios are not only **scarce** but also **safety-critical**, involving atypical motion patterns or high-risk interactions that substantially deviate from head-dominant driving patterns, such as straight driving and lane following Zhang et al. (2026); Lian et al. (2026). Given that anomalous samples are characterized by their deviations from normal patterns, such marked deviations provide a natural basis for regarding tail scenarios as "anomalies" relative to head-dominant ones, a perspective also reflected in prior work Yaman et al. (2026); Müller & Hein (2024). Based on this view, *we reformulate tail sample identification in trajectory prediction as an anomaly detection problem with respect to head-dominant patterns*. Under this reformulation, for three key dimensions affecting future motion—motion dynamics, social interaction, and topological constraints Huang et al. (2022)—we uniformly employ autoencoder (AE) reconstruction errors, a widely used anomaly measure, to quantify their tailness rather than manually designed heuristic indicators, yielding an error triplet as a multi-dimensional tailness profile for reliable tail sample identification.

To further improve the reliability of error-based tailness estimation, we first construct a representative head set for AE training via multi-expert consensus: since head patterns are typically high-frequency and low-complexity, they are more readily learned across diverse models. Therefore, samples with both low mean prediction errors and low variance across models are selected. Second, to mitigate the over-generalization of AEs Gong et al. (2019), we introduce a Sparse Autoencoder that represents latent features using a finite set of head prototypes, thereby constraining the reconstruction space to the head-pattern domain and amplifying reconstruction errors for tail samples.

Based on the multi-dimensional tailness profile, we enhance the model's tail adaptability from both the encoding and decoding perspectives. On the encoding side, we perform dimension-wise contrastive learning, encouraging samples with similar tailness scores along each dimension to learn closer representations while preserving semantic consistency among head representations, thereby improving the discriminability of tail representations. On the decoding side, we introduce a lightweight prompt-guided mechanism that uses only a few parameters to steer the decoder toward encoded tail cues, enabling tail-aware generation while reducing the risk of overfitting. Specifically, samples are partitioned into multiple tailness levels based on the reconstruction errors, each associated with an independent prompt pool. During decoding, each sample retrieves and composes a continuous, scene-specific prompt from its corresponding pool, which is injected into the decoder via Prefix-Tuning Li & Liang (2021) to guide attention toward tail-relevant features, thereby reducing the bias toward head-dominant behaviors and enabling tail-aware trajectory generation.

Our main contributions are as follows: **1.** We recognize the multi-dimensional nature of long-tail phenomena in trajectory prediction and uniformly quantify tailness across dimensions using AE reconstruction errors by reformulating tail identification as anomaly detection against head patterns. The resulting tailness profile guides dimension-wise contrastive learning to improve tail representation discriminability. **2.** We recognize the critical role of decoding in long-tail learning and introduce lightweight prompt-guided decoding to adjust model attention to tail cues during generation, improving tail scene adaptation. **3.** Experiments on Argoverse2 and nuScenes demonstrate substantial gains across diverse tail cases and superior performance over SOTA long-tail learning methods.

2 Related Work

Trajectory prediction aims to infer the future behaviors of traffic agents based on environmental context and historical motion states. Learning-based methods have recently achieved substantial progress on various benchmarks, driven by advances in model architectures Chen et al. (2024); Lan et al. (2025) and training strategies Cheng et al. (2023); Lan et al. (2023). However, recent studies have recognized the prevalent long-tail phenomenon in trajectory prediction datasets. Since head scenarios dominate the data distribution, learned representations tend

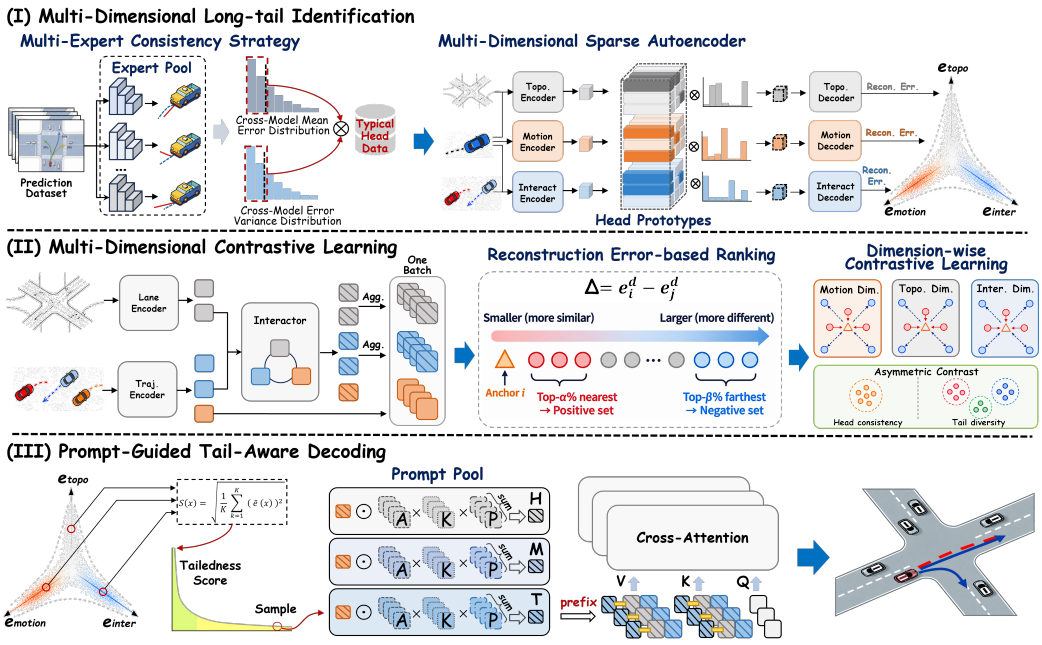

Figure 1: Overview of Tail-MDP, comprising (1) multi-dimensional long-tail identification, (2) dimension-wise contrastive learning, and (3) prompt-guided tail-aware decoding.

to be biased toward frequent patterns, leading to degraded performance on tail scenarios. To address this issue, Makansi et al. (2021) attribute long-tailedness mainly to complex interaction patterns, identify tail samples using Kalman filtering errors, and employ contrastive learning to enhance representation separability between head and tail samples. Inspired by this work, contrastive learning has been widely explored for long-tail trajectory prediction, with existing methods differing mainly in how tail scenarios are identified. Specifically, Wang et al. (2023) and Rao et al. (2025) characterize long-tailed motion patterns through feature clustering, whereas Zhang et al. (2024b) identify tail samples based on model prediction errors. Beyond contrastive learning, residual learning Lian et al. (2026) and active learning Park et al. (2025) have also been introduced for long-tail trajectory prediction, achieving promising results. However, existing methods, on the one hand, mainly characterize long-tailedness from a single dimension, making them insufficient for driving scenarios with multi-dimensional tail properties. On the other hand, they pay limited attention to the decoding stage, preventing encoded tail cues from being effectively exploited during trajectory generation and thereby limiting adaptability to tail scenarios.

3 Methodology

3.1 Multi-Dimensional Long-Tail Identification

3.1.1 Multi-Dimensional Autoencoder

In this section, we develop dimension-specific autoencoders for three key dimensions affecting future motion: motion dynamics, social interactions, and topological constraints, and subsequently use reconstruction errors as a unified criterion to quantify the long-tail properties of each sample across these heterogeneous dimensions, as detailed below.

Dimension of Motion Dynamics: It takes the historical trajectory of the target agent as input, employs a three-layer Transformer encoder to extract kinematic representations, and reconstructs the trajectory with a three-layer Transformer decoder. To achieve effective feature compression, we set the feature dimensionality ratio before and after the encoder to 4:1 and adopt the same architectural design for all subsequent modules. After training, the reconstruction error e_{dyn} could reflect the degree to which the target agent's motion pattern deviates from the head ones.

Dimension of Topological Constraints and Social Interactions: Topological constraints describe how road structures constrain agent behavior and are jointly determined by road topology and the agent state: varying topologies induce distinct constraint structures, while different agent behaviors under the same topology result in diverse constraint forms and intensities. Therefore, reconstructing lane geometry alone is insufficient to capture the behavioral constraints imposed by road topology. To address this, we jointly encode the target trajectory and lane segments, employ attention to model their interactions, and reconstruct both the road structures and the target trajectory from the resulting interactive features z. This design encourages the autoencoder to encode behavior–topology dependencies into its latent representation, such that larger reconstruction errors e_{topo} indicate rarer constraints arising from atypical topologies or complex agent behaviors. Similarly, we construct the autoencoder following the same design for the social interaction dimension. After training, e_{social} measures the deviation of social interaction patterns from those of head samples.

Each autoencoder exhibits clear dimension-specific sensitivity, with its reconstruction error responding primarily to variations in the corresponding dimension, as further validated in **Appendix A.3**. Consequently, we construct a multi-dimensional long-tail identification scheme based on the triplet errors $\mathbf{E} = \{\mathbf{e}_{\text{dyn}}, \mathbf{e}_{\text{topo}}, \mathbf{e}_{\text{social}}\}$, enabling comprehensive assessment of sample's long-tail attributes.

3.1.2 Reconstruction Error Calibration

To improve the reliability of reconstruction errors for tailness estimation, we enhance the autoencoder from two aspects: training data and reconstruction mechanism.

Multi-Expert Consistency Strategy: Autoencoders should be trained on head data to capture their dominant structure. However, since head annotations are unavailable, we design a multi-expert consistency strategy to identify a set of representative head samples for AE training. Given that head samples in driving scenarios are typically high-frequency and low-complexity Wang et al. (2023), they are generally easier for neural networks to learn and generalize from, and thus tend to exhibit consistently low prediction errors across diverse models Arpit et al. (2017); Hacohen et al. (2020). Accordingly, we select E open-source models with diverse architectures and training strategies to form a heterogeneous expert ensemble, whose diversity could help mitigate model-specific bias despite the limited ensemble size, as further validated in **Appendix A.4**. For each sample, we compute the cross-model mean and variance of minFDE and select those with both low mean error and low variance as representative head samples.

Sparse Autoencoder: Conventional autoencoders often over-generalize to tail samples, producing low reconstruction errors that hinder tail identification Gong et al. (2019). To counter this, we introduce a sparse autoencoder (SAE) with an external bank M of head prototypes m_i, learned through the reconstruction objective. Instead of directly using the encoded latent feature z for reconstruction, we reconstruct it as a weighted aggregation of these head prototypes:

$$s_i = \frac{\boldsymbol{z}^\top \boldsymbol{m}_i}{\|\boldsymbol{z}\|_2 \|\boldsymbol{m}_i\|_2}, \qquad w_i = \frac{\exp(s_i)}{\sum_{j=1}^{N} \exp(s_j)}, \qquad \hat{\boldsymbol{z}} = \tilde{\boldsymbol{w}}\boldsymbol{M} = \sum_{i=1}^{N} \tilde{w}_i \boldsymbol{m}_i. \tag{1}$$

By restricting feature representations to the head-pattern space, this mechanism increases reconstruction errors for tail samples, effectively mitigating the over-generalization effect. Furthermore, to prevent prototype homogenization after training and high-fidelity reconstruction of tail samples through dense prototype combinations during inference, we combine hard shrinkage with entropy minimization to enforce sparse addressing, as shown in Eq. 2. This mechanism creates a competitive bottleneck that encourages the memory to learn representative prototypes during training while activating only a small subset during inference, thereby mitigating both issues.

$$\hat{w}_i = \frac{\max(w_i - \epsilon, 0) \cdot w_i}{|w_i - \epsilon| + \kappa}, \qquad \tilde{w}_i = \frac{\hat{w}_i}{\sum_j |\hat{w}_j|}, \qquad L^e_{auto} = \sum_{i=1}^{N} -\widehat{w}_i \cdot \log(\widehat{w}_i), \tag{2}$$

where w_i and $\widehat{w}_i$ denote the cosine-similarity weights before and after shrinkage, with ϵ and κ serving as the shrinkage threshold and a very small positive scalar, respectively.

3.2 Multi-Dimensional Contrastive Learning

Based on the resulting multi-dimensional tailness profile, we propose a dimension-wise contrastive learning mechanism to enhance the discriminability of learned representations.

Specifically, within dimension $\mathrm{d} \in \{\mathrm{dyn}, \mathrm{topo}, \mathrm{inter}\}$, samples with similar reconstruction errors exhibit comparable degrees of long-tailedness. Therefore, for each sample i in a batch, we compute the error discrepancies $\|e_i - e_j\|$ with all other samples j and rank them in ascending order. Samples within the smallest $\alpha\%$ and largest $\beta\%$ are then selected as positive and negative samples, respectively. Subsequently, by pulling samples closer to their positives and pushing them away from their negatives, the model learns a more discriminative representation space for each dimension, thereby improving the separation between head and tail samples:

$$L_{cl}^{d} = -\frac{1}{N_b}\sum_{i=1}^{N_b}\log\frac{\sum_{k=1}^{N_{po}}\exp\left(v_i * v_k/\tau\right)}{\sum_{j=1}^{N_{np}}\exp\left(v_i * v_j/\tau\right)}, \tag{3}$$

where N_b, N_{po} and N_{np} denote the batch size, the number of positive samples and the total number of positive and negative samples, respectively; i, k, and j denote the query sample, its positive sample, and an arbitrary sample. v denotes the encoded vector, and τ =0.5 is the temperature parameter.

Besides, to avoid overemphasizing tail samples at the expense of head-pattern modeling, we introduce a head-oriented asymmetric contrastive learning scheme that aggregates representative head samples selected by Multi-Expert Consensus to enhance the semantic consistency of head data, while leaving other samples unconstrained to preserve their intrinsic multi-dimensional diversity.

$$L_{cl}^{h} = -\frac{1}{N_h}\sum_{i=1}^{N_h}\log\frac{\sum_{k=1}^{N_h-1}\exp\left(v_i * v_k/\tau\right)}{\sum_{j=1}^{N_b}\exp\left(v_i * v_j/\tau\right)}, \tag{4}$$

where N_h denotes the number of the representative head samples. The total contrastive loss is:

$$L_{cl} = L_{cl}^{h} + L_{cl}^{dyn} + L_{cl}^{topo} + L_{cl}^{inter}. \tag{5}$$

3.3 Prompt-Guided Tail-Aware Decoding

In this section, we develop a prompt-guided decoding strategy that conditions the decoding process on lightweight prompts, enabling the model to dynamically attend to tail-relevant features during trajectory generation, thereby avoiding the overfitting associated with expert-based models Coden et al. (2024) and the training instability introduced by hypernetworks Wang et al. (2023).

Sample Partition: We employ Root Mean Square to aggregate reconstruction errors across the three dimensions into a unified tailness metric $S(x)$. To account for their different numerical ranges, we apply Cumulative Distribution Function (CDF) mapping $F_k(\cdot)$ for normalization before aggregation:

$$S(x) = \sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(\tilde{e}_k(x)\right)^2}, \qquad \tilde{e}_k(x) = F_k\big(e_k(x)\big), \qquad F_k(e) = \frac{1}{D}\sum_{i=1}^{D}\mathbf{1}\left(e_k \le e\right), \tag{6}$$

where D is the number of samples and $\mathbf{1}(\cdot)$ is the indicator function. Based on the metric $S(x)$, the dataset is partitioned into Head, Middle, and Tail subsets using an empirical ratio of 7:2:1. Since subset labels are unavailable at inference, we train a classifier on the encoded scene representation to predict the corresponding subset category, with the classification loss denoted as $\mathcal{L}_{\mathrm{cls}}$.

Prompt Formulation: For each identified scenario subset, we construct a dedicated set of prompts to enable scenario-adaptive decoding. Inspired by CODA-Prompt Smith et al. (2023), we replace the rigid one-subset-one-prompt design with a set of M learnable triplets, $\{(\mathbf{P}_m, \mathbf{K}_m, \mathbf{A}_m)\}_{m=1}^{M}$, which allows knowledge sharing within each subset while capturing diverse subset-specific characteristics. Here, $\mathbf{P}_m$ denotes a prompt primitive encoding semantic patterns, $\mathbf{K}_m$ serves as a matching key for relevance estimation, and $\mathbf{A}_m$ performs dimension-wise feature reweighting so that different primitives can attend to distinct semantics. Given encoded features $\boldsymbol{x}$, the corresponding $\mathbf{A}_m$ first reweights the features, and the cosine similarities between the reweighted features and $\mathbf{K}_m$ are then normalized to obtain combination weights. The adaptive prompt $\mathbf{p}$ is finally generated by weighted aggregation of the prompt primitives $\mathbf{P}_m$:

$$\boldsymbol{\alpha} = \gamma(\boldsymbol{x} \odot \boldsymbol{A}, \boldsymbol{K}) = \left[\gamma(\boldsymbol{x} \odot \boldsymbol{A}_m, \boldsymbol{K}_m)\right]_{m=1}^{M}, \qquad \boldsymbol{p} = \sum\nolimits_{m=1}^{M}\alpha_m \boldsymbol{P}_m. \tag{7}$$

Prefix Tuning-Based Decoder: Once the sample-specific prompts are obtained, effectively integrating them into the decoder is essential for tail-aware inference. To this end, we construct a dual-layer

cross-attention decoder, where the encoded scene features serve as keys (K) and values (V), while learnable mode queries initialized randomly act as queries (Q). An MLP then projects the decoded features into the physical coordinate space, producing (M) predicted trajectories and their confidence scores. Building on this decoder, we adopt a prefix tuning strategy to guide the inference process by prepending the prompts to the K and V sequences at each attention layer:

$$f_{Decode} = \text{CA}(\boldsymbol{h}_Q, [\boldsymbol{p}; \boldsymbol{h}_K], [\boldsymbol{p}; \boldsymbol{h}_V]). \tag{8}$$

By serving as contextual anchors, these lightweight prompts adaptively modulate the cross-attention distribution to emphasize subset-specific semantics. This reduces the dominance of head patterns over discriminative tail features, thereby improving prediction accuracy in long-tail scenarios. Finally, we employ a joint training strategy to optimize the model, where the total loss comprises the prediction loss, the proposed multi-dimensional contrastive loss and the subset classification loss:

$$\mathcal{L} = \mathcal{L}_{pred} + \mathcal{L}_{cl} + \mathcal{L}_{cls} \tag{9}$$

4 Experiments

4.1 Datasets and Metrics

We evaluate Tail-MDP on two large-scale benchmarks, Argoverse2 (AV2) Wilson et al. and the UniTraj version Feng et al. (2024) of nuScenes (nuSc) Caesar et al. (2020), using **minADE** and **minFDE** with $M = 6$ predicted trajectories to assess its generalizability across heterogeneous settings. Further details on datasets, metric definitions, and implementation are provided in **Appendix A.1** and **A.2**. For each benchmark, we construct tail subsets using three distinct and widely used schemes Rao et al. (2025); Makansi et al. (2021); Wang et al. (2023) to evaluate Tail-MDP.

• **Prediction Error:** This criterion characterizes scenarios that are challenging for the baseline model. Specifically, based on the baseline's minFDE, we construct six long-tail subsets comprising the top $p\%$ highest-error samples, where $p \in \{1, 2, 3, 4, 5, 10\}$, alongside the entire dataset.

• **Kalman Error:** This criterion partitions the dataset by the minFDE of a Kalman filter (KF) Grewal (2025), which predicts motion solely from historical states. Larger values indicate that dense interactions or complex road topology render inertial extrapolation inadequate. We evaluate the top $p\%$ highest-error samples, where $p \in \{1, 2, 3, 4, 5, 10\}$, alongside the full dataset.

• **State Complexity:** To evaluate performance in unconventional ego-motion long-tail scenarios, we categorize samples into four groups—Rapid Acceleration (RA), Rapid Deceleration (RD), Sharp Lane Changes (SLC), and Sharp Turns (ST)—based on the target agent's position and velocity.

4.2 Comparison Across Multiple Long-Tail Dimensions

To evaluate Tail-MDP under diverse long-tail conditions, we use Forecast-MAE Cheng et al. (2023) as the baseline and report its performance across the aforementioned categories of long-tail scenarios in Table 1(a)–(c). On the baseline-error-based subsets, Tail-MDP consistently improves long-tail prediction on both datasets. For the top 1% most challenging samples, it reduces minADE and minFDE by 35% and 38% on Argoverse2 and by 30% and 36% on nuScenes, respectively. Similar improvements are observed on the KF-error-based subsets. For the top 1% samples, Tail-MDP reduces minADE and minFDE by 6% and 11% on Argoverse2 and by 16% and 27% on nuScenes, respectively. Furthermore, across unconventional motion categories, Tail-MDP also outperforms the baseline in all cases on both datasets. Collectively, these results demonstrate that Tail-MDP could effectively address heterogeneous forms of long-tailedness and enable more accurate prediction in rare yet safety-critical scenarios.

4.3 Comparison with SOTA Long-tail Learning Methods

To demonstrate the effectiveness of Tail-MDP, we compare it with existing long-tail trajectory prediction methods on the baseline-error-based tail subsets of Argoverse2. For a fair comparison, all methods are implemented using the same baseline, Forecast-MAE. As shown in Table 2, Tail-MDP consistently achieves the best performance across different tail ratios. Meanwhile, it maintains comparable overall performance, demonstrating effective improvements on tail samples without compromising general prediction accuracy.

Table 1: Prediction errors on the Argoverse2 and nuScenes validation sets across multiple dimensions of long-tailedness.

(a) Samples categorized by the prediction error (minFDE) of the baseline.

Dataset	Model	Top 1%	Top 2%	Top 3%	Top 4%	Top 5%	Top 10%	All
AV2	Base	5.04/14.80	3.90/11.21	3.34/9.26	2.94/8.06	2.67/7.21	2.01/5.03	**0.71**/1.40
	Ours	**3.26/9.18**	**2.71/7.36**	**2.42/6.37**	**2.19/5.62**	**2.03/5.13**	**1.65/3.84**	0.72/**1.40**
nuSc	Base	6.26/18.20	5.05/14.98	4.40/13.15	3.98/11.95	3.67/11.07	2.80/8.63	1.14/2.22
	Ours	**4.37/11.72**	**3.56/9.73**	**3.20/8.65**	**2.95/7.94**	**2.75/7.42**	**2.21/5.94**	**1.10/2.08**

(b) Samples categorized by the prediction error (minFDE) of the Kalman Filter.

Dataset	Model	Top 1%	Top 2%	Top 3%	Top 4%	Top 5%	Top 10%	All
AV2	Base	1.40/3.09	1.24/2.66	1.16/2.48	1.12/2.34	1.06/2.18	0.96/1.95	**0.71**/1.40
	Ours	**1.32/2.76**	**1.19/2.45**	**1.10/2.30**	**1.04/2.22**	**0.99/2.12**	**0.95/1.86**	0.72/**1.40**
nuSc	Base	2.95/7.67	2.47/6.03	2.25/5.48	2.14/5.19	2.12/5.18	1.86/4.45	1.14/2.22
	Ours	**2.49/5.61**	**2.12/4.57**	**1.95/4.19**	**1.91/4.21**	**1.92/4.28**	**1.74/3.89**	**1.10/2.08**

(c) Samples categorized by state complexity.

Dataset	Model	RA	RD	SLC	ST	All
AV2	Base	0.84 /1.42	1.00/1.68	1.26/2.13	1.59/3.10	**0.71**/1.40
	Ours	**0.82/1.37**	**0.95/1.56**	**1.24/2.06**	**1.52/2.74**	0.72/**1.40**
nuSc	Base	0.96/1.54	1.28/2.39	2.04/4.92	2.70/6.76	1.14/2.22
	Ours	**0.91/1.45**	**1.23/2.29**	**1.92/4.61**	**2.48/6.29**	**1.10/2.08**

Table 2: Comparison with SOTA long-tail learning methods on the Argoverse2 validation set.

Model	Venue	Top 1%	Top 2%	Top 3%	Top 4%	Top 5%	All
Baseline Cheng et al. (2023)	ICCV	5.04/14.80	3.90/11.21	3.34/9.26	2.94/8.06	2.67/7.21	**0.71/1.40**
+Resample	/	4.28/12.40	3.39/9.60	2.93/8.02	2.61/6.97	2.43/6.38	0.73/1.46
+Reweight	/	4.31/12.62	3.35/9.50	2.88/7.87	2.56/6.82	2.37/6.17	0.73/1.43
+Makansi et al. (2021)	ICCV	4.25/12.45	3.33/9.50	2.91/8.03	2.59/7.00	2.39/6.33	0.74/1.41
+Tract Zhang et al. (2024b)	IV	4.35/12.65	3.49/9.89	3.04/8.31	2.68/7.18	2.45/6.44	0.72/1.42
+Amend Coden et al. (2024)	arxiv	4.60/13.48	3.58/10.24	3.10/8.62	2.76/7.52	2.55/6.80	0.73/1.44
+ProtoAug Lu et al. (2026)	IoT	3.96/12.19	/	/	/	2.41/6.99	0.73/1.40
Ours	/	**3.26/9.18**	**2.71/7.36**	**2.42/6.37**	**2.19/5.62**	**2.03/5.13**	0.72/**1.40**

4.4 Generalization Across Different SOTA Models

Table 3: Generalization performance of Tail-MDP across different SOTA models.

Baseline	Setting	Top 1%	Top 2%	Top 3%	Top 4%	Top 5%	All
Forecast-MAE Cheng et al. (2023)	Vanilla	5.04/14.80	3.90/11.21	3.34/9.26	2.94/8.06	2.67/7.21	**0.71/1.40**
	Tail-MDP	**3.26/9.18**	**2.71/7.36**	**2.42/6.37**	**2.19/5.62**	**2.03/5.13**	0.72/**1.40**
EMP Prutsch et al. (2024)	Vanilla	4.39/12.74	3.42/9.96	2.93/8.48	2.62/7.54	2.40/6.87	0.71/1.39
	Tail-MDP	**3.47/9.75**	**2.76/7.73**	**2.39/6.56**	**2.19/5.92**	**2.03/5.42**	**0.70/1.39**
SHARP Prutsch et al. (2026a)	Vanilla	3.94/11.78	3.03/9.02	2.63/7.63	2.35/6.74	2.15/6.10	0.64/1.20
	Tail-MDP	**3.33/9.40**	**2.67/7.42**	**2.336/6.35**	**2.15/5.69**	**1.99/5.20**	**0.64/1.19**

As shown in Table 3, we report the performance of Tail-MDP integrated into multiple SOTA predictors beyond Forecast-MAE. The results show consistent improvements on tail scenarios, demonstrating its strong cross-model applicability. Moreover, Tail-MDP still delivers substantial gains on top-performing models such as SHARP, highlighting that tail-oriented learning remains essential even for highly competitive predictors and further validating the effectiveness of Tail-MDP.

4.5 Validation of Reconstruction-Based Tail Identification

To validate the reformulation of tail identification as anomaly detection, we construct head and tail sets based on two established long-tail criteria, State Complexity and KF error, and train the AE on the head sets. As shown in Table 4, tail samples consistently exhibit higher reconstruction errors,

Table 4: AE reconstruction errors.

Metric	Head	Tail
KF	5.2 ($\times 10^{-3}$)	17.6 ($\times 10^{-3}$)
State	8.8 ($\times 10^{-3}$)	26.5 ($\times 10^{-3}$)

Table 5: ROC-AUC values.

Model	KF	State
AE	0.89	0.93
SAE	**0.92**	**0.99**

Table 6: Ablation study on multi-dimensional contrast and prompt-guided decoding.

Contrast				Prompt						
Dyn.	Inter.	Topo.	Asym.	Decode	Top 1%	Top 2%	Top 3%	Top 4%	Top 5%	All
					5.04/14.80	3.90/11.21	3.34/9.26	2.94/8.06	2.67/7.21	**0.71**/1.40
√					3.93/11.34	3.12/8.82	2.75/7.47	2.46/6.52	2.26/5.85	0.73/1.43
	√				4.02/11.63	3.22/9.05	2.82/7.68	2.52/6.68	2.31/6.00	0.74/1.47
		√			4.06/11.84	3.26/9.23	2.87/7.85	2.55/6.82	2.35/6.15	0.75/1.49
	√	√			3.98/11.63	3.20/9.02	2.83/7.67	2.52/6.69	2.33/6.05	0.75/1.49
√	√	√			3.54/10.00	2.91/7.93	2.56/6.77	2.29/5.93	2.11/5.37	0.76/1.46
√	√	√	√		3.45/9.86	2.85/7.80	2.53/6.69	2.27/5.88	2.10/5.36	0.72/1.41
√	√	√	√	√	**3.26/9.18**	**2.71/7.36**	**2.42/6.37**	**2.19/5.62**	**2.03/5.13**	0.72/**1.40**

supporting the proposed reformulation. We further evaluate SAE using the area under the receiver operating characteristic curve (ROC-AUC) on the aforementioned head and tail samples. As shown in Table 5, SAE achieves a higher ROC-AUC, demonstrating stronger tail identification capability.

4.6 Ablation Study

This section evaluates the key components of Tail-MDP using the Argoverse2 tail subsets identified by baseline prediction errors. As shown in Table 6, contrastive learning across the motion, interaction, and topology dimensions consistently improves long-tail prediction, demonstrating their complementarity in modeling heterogeneous tail patterns. Visualizations of the learned representations before and after contrastive learning are provided in **Appendix A.5**. Asymmetric contrastive learning further strengthens head-pattern modeling, reducing tail-oriented bias and improving overall performance. Prompt-guided decoding yields additional gains across all metrics, showing that subset-specific prompts mitigate decoder bias and improve adaptation to diverse tail scenarios.

4.7 Hyperparameter Sensitivity Study

(1) Ratios α and β: We study the selection ratios α and β for positive and negative samples in contrastive learning. As shown in Table 7, setting $\alpha = 10\%$ and $\beta = 40\%$ yields superior performance. A smaller α preserves greater consistency among positive samples, whereas a larger β increases the diversity of negative samples, facilitating discriminative representation learning. **(2) Prompt Length:** As shown in Table 8, a prompt length of 20 achieves the best performance. Shorter prompts provide insufficient semantic guidance, whereas overly long prompts introduce redundant noise. Therefore, we adopt a length of 20 as the optimal configuration, achieving the best trade-off.

Table 7: Sensitivity to proportion allocation.

Dim.	$\alpha : \beta$	Top 1%	Top 2%	Top 3%	All
Dyn.	10:40	**3.93/11.34**	**3.12/8.82**	**2.75/7.47**	**0.73/1.43**
	20:50	4.01/11.67	3.21/9.00	2.81/7.61	0.75/1.48
	30:30	4.22/12.28	3.35/9.44	2.88/7.87	0.73/1.43
Inter.	10:40	**4.02/11.63**	**3.22/9.05**	**2.82/7.68**	0.74/1.47
	20:50	4.12/11.90	3.27/9.19	2.86/7.78	0.73/1.45
	30:30	4.21/12.28	3.36/9.50	2.92/7.97	**0.73/1.44**
Topo.	10:40	**4.06/11.84**	**3.26/9.23**	**2.87/7.85**	0.75/1.49
	20:50	4.12/11.89	3.27/9.11	2.85/7.68	**0.74/1.46**
	30:30	4.27/12.46	3.35/9.44	2.92/7.97	0.75/1.48

Table 8: Sensitivity to prompt length.

Length	Top 1%	Top 2%	Top 3%	All
5	3.44/9.73	2.84/7.68	2.53/6.63	0.73/1.41
10	3.30/9.33	2.76/7.53	2.47/6.51	0.73/1.41
20	**3.26/9.18**	**2.71/7.36**	**2.42/6.37**	**0.72/1.40**
30	3.51/9.95	2.86/7.81	2.54/6.68	0.74/1.43
40	3.47/9.87	2.82/7.81	2.51/6.69	0.73/1.41

4.8 Decoder Attention Visualization

We visualize decoder attention in a representative tail scenario involving an unprotected left turn at a complex intersection in Fig. 2. Without prompts, the decoder mainly attends to straight lanes, which are more relevant to head-dominant cruising patterns. After prompt injection, attention shifts toward tail-relevant elements, including potentially traversable lanes and interacting traffic participants, demonstrating that prompt-guided decoding better exploits tail-specific information during trajectory generation.

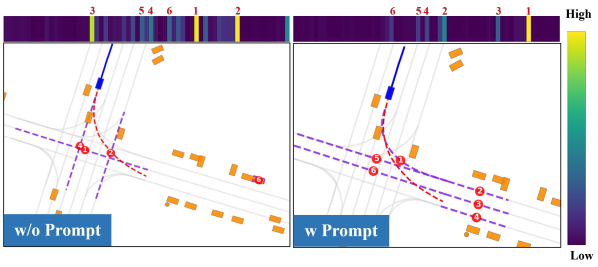

Figure 2: Prompt-guided attention visualization, with high-attention elements highlighted in purple.

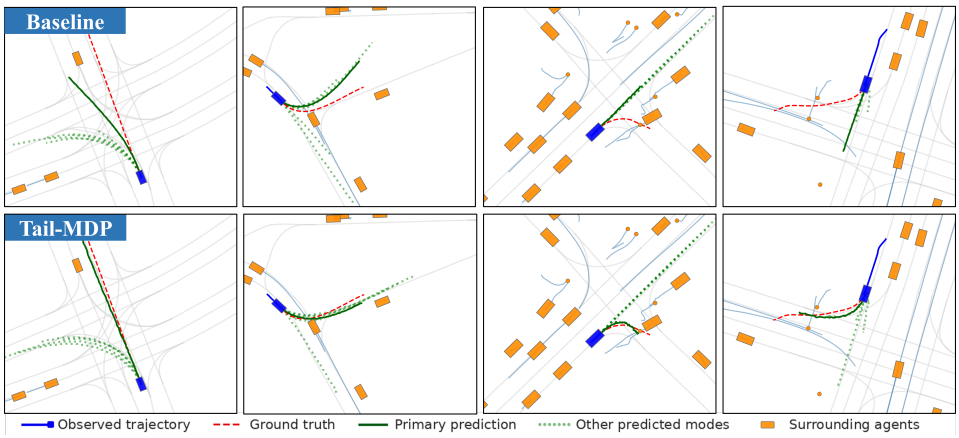

Figure 3: Visualization of prediction results in representative tail scenarios that involve (1) strong motion nonlinearity, (2) atypical topological constraints, and (3–4) complex interaction dynamics.

4.9 Predicted Trajectory Visualization

We further visualize predictions in four tail scenarios and compare Tail-MDP with the baseline, as shown in Fig. 3. Scenario 1 presents a strongly nonlinear motion pattern, where the target agent transitions from a stationary state to straight-line motion. Tail-MDP accurately captures this abrupt transition, whereas the baseline deviates from the ground truth and even predicts trajectories entering the opposing lane. Scenario 2 depicts a T-junction with strong topological constraints. Tail-MDP better respects the complex junction topology and generates road-compliant trajectories, while the baseline predictions extend beyond the drivable area. Scenarios 3 and 4 involve strong interactions with surrounding agents. Tail-MDP effectively accounts for these interactions and generates collision-free predictions, avoiding the simplistic motion extrapolation of the baseline. Overall, these results demonstrate the effectiveness of Tail-MDP in handling multi-dimensional long-tailedness.

5 Conclusion

In this paper, we present Tail-MDP, a multi-dimensional long-tail learning framework with tail-aware decoding for trajectory prediction. We reformulate tail sample identification as anomaly detection with respect to head-dominant patterns and employ AE reconstruction errors as a unified criterion to quantify tailness across motion dynamics, social interactions, and topological constraints. We further devise dimension-wise contrastive learning and prompt-guided decoding to enhance tail representations and enable tail-aware prediction. Extensive experiments on Argoverse2 and nuScenes demonstrate consistent improvements across diverse long-tail scenarios.

AI USE STATEMENT

Generative AI tools were used solely for grammar checking. No generative AI tools were used for research methodology, experiment design, implementation, or result interpretation. We take responsibility for the final content of this work with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our work, we provide detailed descriptions of the overall Tail-MDP framework, including the tail scenario identification process, dimension-wise representation learning, and prompt-guided decoding, in the main paper. The appendix provides additional implementation details, training configurations, and experimental settings to facilitate the reproduction of our method and experimental results.

ETHICS STATEMENT

This research strictly complies with the ICLR Code of Ethics. Our empirical evaluations are conducted exclusively on standard, publicly available benchmarks and do not involve the collection or use of private or sensitive personal information. All datasets are used in accordance with their original licenses and terms of use. We do not identify any significant ethical concerns regarding privacy, security, or algorithmic bias arising from this work.

REFERENCES

Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, pp. 233–242, 2017.

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

Chongpu Chen, Xinbo Chen, Yi Yang, and Peng Hang. Sparse attention graph convolution network for vehicle trajectory prediction. *IEEE Transactions on Vehicular Technology*, 73(12):18294–18306, 2024.

Jie Cheng, Xiaodong Mei, and Ming Liu. Forecast-mae: Self-supervised pre-training for motion forecasting with masked autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8679–8689, 2023.

Mercurius Ray Coden, Ehsan Ahmadi, Soheil Mohamad Alizadeh Shabestary, and Amir Rasouli. Amend: A mixture of experts framework for long-tailed trajectory prediction. *arXiv e-prints*, pp. arXiv–2402, 2024. URL https://arxiv.org/pdf/2402.08698.

Nachiket Deo, Eric Wolff, and Oscar Beijbom. Multimodal trajectory prediction conditioned on lane-graph traversals. In *Conference on robot learning*, pp. 203–212. PMLR, 2022.

Lan Feng, Mohammadhossein Bahari, Kaouther Messaoud Ben Amor, Éloi Zablocki, Matthieu Cord, and Alexandre Alahi. Unitraj: A unified framework for scalable vehicle trajectory prediction. In *European Conference on Computer Vision*, pp. 106–123. Springer, 2024.

G Ganeshaaraj, Tharindu Fernando, Sridha Sridharan, and Clinton Fookes. Enhancing predictive performance on long-tail trajectories via clustering and specialized decoders. *Pattern Recognition*, pp. 112315, 2025.

Roger Girgis, Florian Golemo, Felipe Codevilla, Martin Weiss, Jim Aldon D'Souza, Samira Ebrahimi Kahou, Felix Heide, and Christopher Pal. Latent variable sequential set transformers for joint multi-agent motion prediction. *arXiv preprint arXiv:2104.00563*, 2021. URL https://arxiv.org/pdf/2104.00563.

Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1705–1714, 2019.

Mohinder S Grewal. Kalman filtering. In *International encyclopedia of statistical science*, pp. 1285–1289. Springer, 2025.

Guy Hacohen, Leshem Choshen, and Daphna Weinshall. Let's agree to agree: Neural networks share classification order on real datasets. In *International Conference on Machine Learning*, pp. 3950–3960. PMLR, 2020.

Yanjun Huang, Jiatong Du, Ziru Yang, Zewei Zhou, Lin Zhang, and Hong Chen. A survey on trajectory-prediction methods for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 7(3):652–674, 2022.

Zhengxing Lan, Lingshan Liu, Yilong Ren, Zhiyong Cui, and Haiyang Yu. Diffutory: A future feature and mode association augmented diffusion model for trajectory prediction in autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 2025.

Zhiqian Lan, Yuxuan Jiang, Yao Mu, Chen Chen, and Shengbo Eben Li. Sept: Towards efficient scene representation learning for motion prediction. In *The Twelfth International Conference on Learning Representations*, 2023.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, 2021.

Yuansheng Lian, Ke Zhang, and Meng Li. Cdkformer: Contextual deviation knowledge-based transformer for long-tail trajectory prediction. *Transportation Research Part C: Emerging Technologies*, 183:105430, 2026.

Mengmeng Liu, Hao Cheng, Lin Chen, Hellward Broszio, Jiangtao Li, Runjiang Zhao, Monika Sester, and Michael Ying Yang. Laformer: Trajectory prediction for autonomous driving with lane-aware scene constraints. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2039–2049, 2024.

Ziheng Lu, Yingfeng Cai, Hai Wang, Long Chen, Yang Wang, and Xinxin Zuo. Protoaug: Prototype-guided uncertainty-aware augmentation for long-tail motion prediction. *IEEE Internet of Things Journal*, 2026.

Osama Makansi, Özgün Çiçek, Yassine Marrakchi, and Thomas Brox. On exposing the challenging long tail in future prediction of traffic actors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13147–13157, 2021.

Maximilian Müller and Matthias Hein. Logex: Improved tail detection of extremely rare histopathology classes via guided diffusion. *arXiv preprint arXiv:2409.01317*, 2024.

Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. *arXiv preprint arXiv:2207.05844*, 2022. URL `https://arxiv.org/pdf/2207.05844`.

Daehee Park, Monu Surana, Pranav Desai, Ashish Mehta, Reuben John, and Kuk-Jin Yoon. Generative active learning for long-tail trajectory prediction via controllable diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27839–27850, 2025.

Alexander Prutsch, Horst Bischof, and Horst Possegger. Efficient motion prediction: A lightweight & accurate trajectory prediction model with fast training and inference speed. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9411–9417. IEEE, 2024.

Alexander Prutsch, Christian Fruhwirth-Reisinger, David Schinagl, and Horst Possegger. Sharp: Short-window streaming for accurate and robust prediction in motion forecasting. *arXiv preprint arXiv:2603.28091*, 2026a.

Alexander Prutsch, David Schinagl, and Horst Possegger. Streaming real-time trajectory prediction using endpoint-aware modeling. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3005–3014. IEEE, 2026b.

Bin Rao, Haicheng Liao, Yanchen Guan, Chengyue Wang, Bonan Wang, Jiaxun Zhang, and Zhenning Li. Amd: Adaptive momentum and decoupled contrastive learning framework for robust long-tail trajectory prediction. *arXiv preprint arXiv:2507.01801*, 2025. URL https://arxiv.org/pdf/2507.01801.

Bin Rao, Haicheng Liao, Chengyue Wang, Keqiang Li, Zhenning Li, and Hai Yang. Sail: Scene-aware adaptive iterative learning for long-tail trajectory prediction in autonomous vehicles. *Transportation Research Part C: Emerging Technologies*, 190:105757, 2026.

Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. *Advances in Neural Information Processing Systems*, 35:6531–6543, 2022.

James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11909–11919, 2023.

Nan Song, Bozhou Zhang, Xiatian Zhu, and Li Zhang. Motion forecasting in continuous driving. *Advances in Neural Information Processing Systems*, 37:78147–78168, 2024.

Divya Thuremella and Lars Kunze. Prediction of social dynamic agents and long-tailed learning challenges: a survey. *Journal of Artificial Intelligence Research*, 77:1697–1772, 2023.

Jifeng Wang, Kaouther Messaoud, Yuejiang Liu, Juergen Gall, and Alexandre Alahi. Forecast-peft: Parameter-efficient fine-tuning for pre-trained motion forecasting models. *arXiv preprint arXiv:2407.19564*. URL https://arxiv.org/pdf/2407.19564?

Yuning Wang, Pu Zhang, Lei Bai, and Jianru Xue. Fend: A future enhanced distribution-aware contrastive learning framework for long-tail trajectory prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1400–1409, 2023.

Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Burhaneddin Yaman, Xinyue Wang, Zhaoyin Jia, Min Cai, Thomas Lampo, Jin Sun, Vivasvat Keswani, and Danhua Guo. Remedying the curse of autonomous driving: Vlm driven training-free framework for efficient long-tail video detection. 2026.

Bozhou Zhang, Nan Song, and Li Zhang. Decoupling motion forecasting into directional intentions and dynamic states. *Advances in Neural Information Processing Systems*, 37:106582–106606, 2024a.

Bozhou Zhang, Nan Song, Bingzhao Gao, and Li Zhang. Relative position matters: Trajectory prediction and planning with polar representation. *arXiv preprint arXiv:2508.11492*, 2025.

Cong Zhang, Bangyang Wei, Yang Liu, and Samuel Labi. World model-based long-tail and scenario-specific generation for autonomous driving. *Journal of Intelligent and Connected Vehicles*, 2026.

Junrui Zhang, Mozhgan Pourkeshavarz, and Amir Rasouli. Tract: A training dynamics aware contrastive learning framework for long-tail trajectory prediction. *arXiv preprint arXiv:2404.12538*, 2024b. URL https://arxiv.org/pdf/2404.12538.

Lu Zhang, Peiliang Li, Sikang Liu, and Shaojie Shen. Simpl: A simple and efficient multi-agent motion prediction baseline for autonomous driving. *IEEE Robotics and Automation Letters*, 9(4): 3767–3774, 2024c.

Yang Zhou, Hao Shao, Letian Wang, Steven L Waslander, Hongsheng Li, and Yu Liu. Smartrefine: A scenario-adaptive refinement framework for efficient motion prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15281–15290, 2024.

Zikang Zhou, Luyao Ye, Jianping Wang, Kui Wu, and Kejie Lu. Hivt: Hierarchical vector transformer for multi-agent motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8823–8833, 2022.

Zikang Zhou, Jianping Wang, Yung-Hui Li, and Yu-Kai Huang. Query-centric trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17863–17873, 2023.

Zhenyu Zong, Yuchen Wang, Haohong Lin, Lu Gan, and Huajie Shao. A generalizable physics-guided causal model for trajectory prediction in autonomous driving. *arXiv preprint arXiv:2602.13936*, 2026.

A Appendix

A.1 Training Details

We adopt Forecast-MAE Cheng et al. (2023) as the baseline while retaining its original model configuration and pretrained weights. Training is conducted for 100 epochs on nuScenes using the UniTraj platform Feng et al. (2024) and for 60 epochs on Argoverse2, with a batch size of 64. The learning rate is linearly warmed up from 10^{-6} to 10^{-3} over the first 10 epochs and decayed following a cosine schedule. All experiments are implemented on NVIDIA RTX 4090 GPUs.

For the SAEs, the number of prototypes in the memory bank is set to $N = 64$. The shrinkage threshold ϵ is set to 2×10^{-2}, 3×10^{-2}, and 3×10^{-2} for the motion-specific, interaction-specific, and topology-specific AEs, respectively, while the entropy loss coefficient is set to 2×10^{-4}. The number of primitives in the Prompt Pool is set to $M = 10$.

For multi-expert consensus, we construct an expert pool of open-source predictors for each benchmark. Samples whose cross-expert prediction error mean and variance both fall within the lowest 80% are selected as representative head samples for autoencoder training. To mitigate model-specific bias in representative head sample selection, we deliberately construct a diverse expert pool spanning different network architectures and training strategies. For Argoverse2, we employ Forecast-MAE Cheng et al. (2023), EMP-D and EMP-M Prutsch et al. (2024), DeMo Zhang et al. (2024a), QCNet Zhou et al. (2023), SmartRefine Zhou et al. (2024), RealMotion Song et al. (2024), Forecast-PEFT Wang et al., SimPL Zhang et al. (2024c), Polaris Zhang et al. (2025), and SEAM Prutsch et al. (2026b) as experts. For nuScenes, the expert pool consists of Forecast-MAE Cheng et al. (2023), EMP Prutsch et al. (2024), MTR Shi et al. (2022), AutoBot Girgis et al. (2021), Wayformer Nayakanti et al. (2022), PCM Zong et al. (2026), LAformer Liu et al. (2024), PGP Deo et al. (2022), and HiVT Zhou et al. (2022).

Since subset labels (i.e., Head, Middle, or Tail) are unavailable at inference time, a lightweight classification head to directly predict the subset category from the scene representation is introduced, thereby avoiding the need to run multiple autoencoders during inference. During training, the corresponding Prompt Pool is activated according to the ground-truth subset label. In parallel, a two-layer MLP classification head takes the scene representation as input and predicts its subset category. To prevent the auxiliary classification objective from affecting the learned scene representation, the scene features are detached before being fed into the classification head. The classifier is optimized using the cross-entropy loss:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} y_{i,c} \log p_{i,c},$$

where $C = 3$ corresponds to the Head, Middle, and Tail subsets. During inference, the Prompt Pool is selected directly according to the subset category predicted by the classification head, thereby achieving prompt routing.

A.2 Datasets and Metrics

Argoverse2: The Argoverse2 Motion Forecasting Dataset Wilson et al. contains 250,000 driving scenarios collected from six U.S. cities across diverse traffic environments. The dataset is officially split into approximately 200,000 training scenarios, 25,000 validation scenarios, and 25,000 test scenarios. Each scenario spans 11 seconds and is sampled at 10 Hz, where the first 5 seconds are provided as historical observations and the subsequent 6 seconds constitute the prediction horizon. Each scenario contains trajectories of surrounding traffic participants, including their positions, headings, velocities, and object categories, together with a scenario-specific high-definition map containing lane, crosswalk, and drivable-area geometry. The interaction-rich and actor-dense scenarios make AV2 a widely used benchmark for evaluating motion forecasting methods under complex traffic conditions.

nuScenes: The nuScenes dataset Caesar et al. (2020) contains 1,000 urban driving scenes collected in Boston and Singapore, covering diverse traffic conditions and road environments. Each scene spans approximately 20 seconds, with agent annotations provided at 2 Hz. For the motion forecasting task, the official prediction split contains 32,186 training instances, 8,560 validation instances, and 9,041 test instances. Each prediction instance provides 2 seconds of historical observations and requires forecasting the subsequent 6 seconds of future motion. The dataset provides trajectories and annotations for surrounding traffic participants, together with high-definition map information describing lane geometry, road boundaries, pedestrian crossings, and other semantic elements.

Metrics: We evaluate the prediction performance using $\textbf{minADE}_6$ and $\textbf{minFDE}_6$, where minADE measures the average L2 distance between the best of six predicted trajectories and the ground truth over the prediction horizon, while minFDE measures the L2 distance between the endpoint of the best predicted trajectory and the ground-truth endpoint.

A.3 Dimension-Specific Perturbation Analysis

To validate whether the reconstruction errors in the three dimensions can characterize the degree of long-tailedness in their corresponding dimensions, we conduct a dimension-specific perturbation experiment, as shown in Fig. 4.

For motion dynamics, we modify the target vehicle's historical speed profile from normal cruising to rapid deceleration, thereby introducing a motion pattern that substantially deviates from normal driving behavior. The reconstruction error of the motion-dynamics AE increases significantly, by 339% relative to the reference scenario, while the errors in the other two dimensions remain nearly unchanged. **For social interactions**, we modify the position and motion state of a neighboring agent such that it decelerates in front of the target vehicle, thereby strengthening the interaction between the two agents and constructing a more complex interaction scenario. The reconstruction error of the social-interaction AE increases by 51% compared with the reference scenario, whereas the errors in the motion-dynamics and topological-constraint dimensions exhibit only minor changes. **For topological constraints**, we add virtual lanelets to transform the original road topology into a double-intersection configuration, thereby introducing a more complex topological structure. The reconstruction error of the topological-constraint AE increases by 20% relative to the reference scenario, while the errors in the other two dimensions remain nearly unchanged. This experiment verifies that the three reconstruction errors are not merely correlated anomaly scores, but exhibit clear dimension-specific responses to controlled perturbations.

A.4 Stability of Representative Head-Sample Identification

To assess whether a limited number of experts can stably identify representative head samples, we progressively expand a heterogeneous expert pool comprising models with diverse architectures and training strategies on Argoverse2. We first use 6 models to select a subset, DS_2, according to the multi-expert consensus criterion. We then add new models to the pool and apply the same criterion to DS_2 as the candidate set to obtain DS_3. This procedure is repeated until all 11 selected models

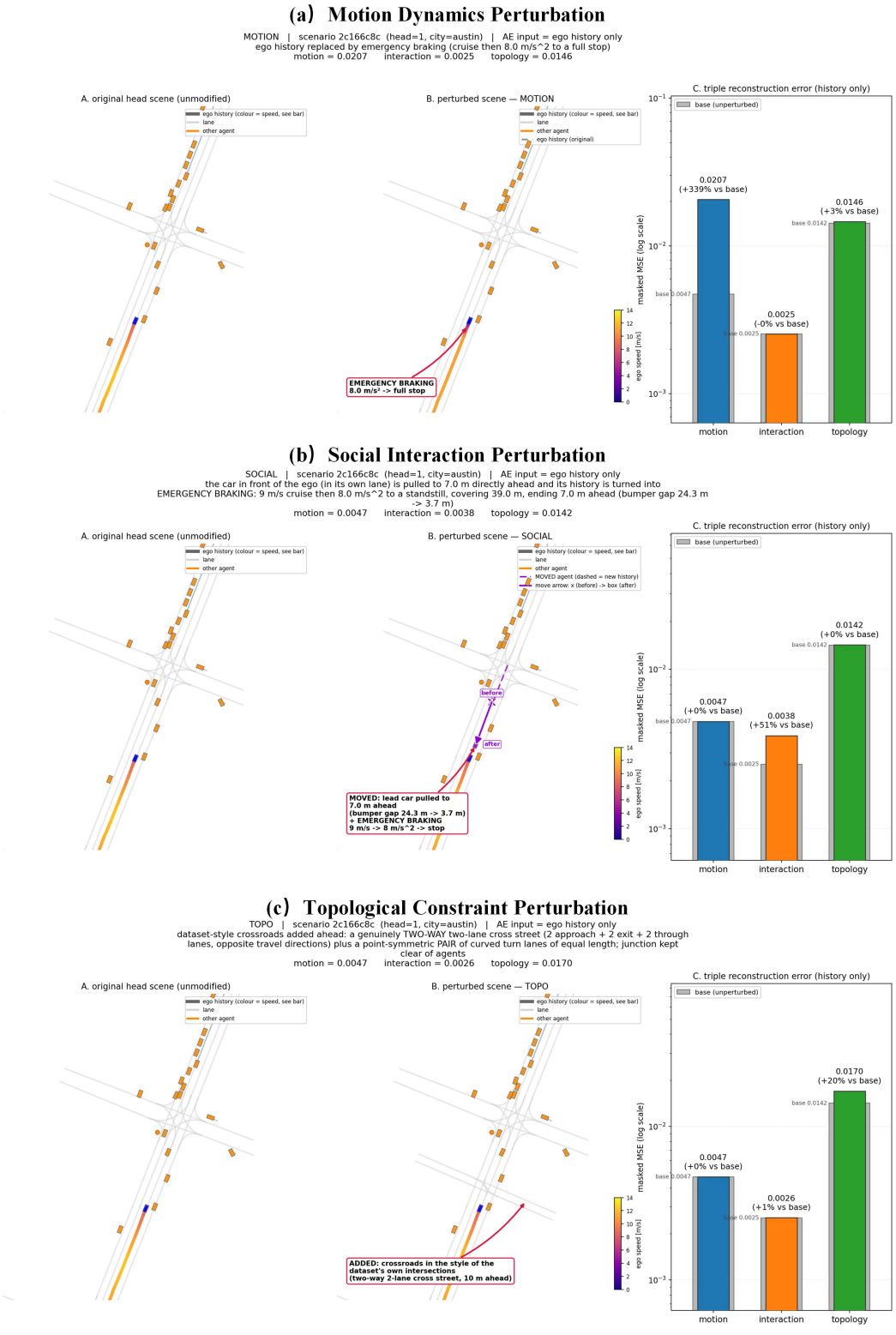

Figure 4: Dimension-specific perturbation analysis.

are included. Finally, we compute the sample overlap between consecutive subsets:

$$\text{Overlap}(DS_i, DS_{i+1}) = \frac{|DS_i \cap DS_{i+1}|}{|DS_i|}.$$

As shown in Fig. 5, the number of selected samples decreases and stabilizes as the expert pool grows, while the overlap between consecutive subsets increases, reaching 99.01% at the final expansion. This indicates that the last expansion removes only a small fraction of previously selected samples. These results suggest that the constructed heterogeneous expert pool is sufficient to identify a stable, representative head subset for subsequent autoencoder training.

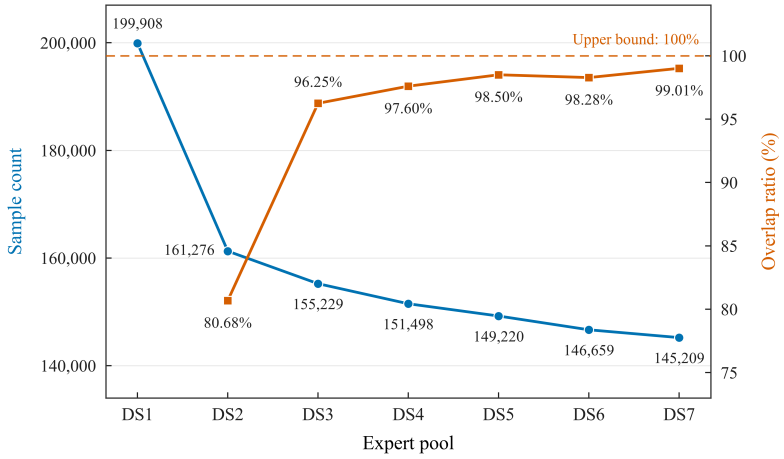

Figure 5: Curves of selected head-sample count and consecutive-subset overlap across expert pool sizes.

A.5 Feature Space Visualization

We use t-SNE to visualize the encoded representations in 2D feature space. As shown in Fig. 6, Tail-MDP produces more compact tail clusters and clearer head-tail separation across all dimensions than the baseline, providing more informative representations for prediction in tail scenarios.

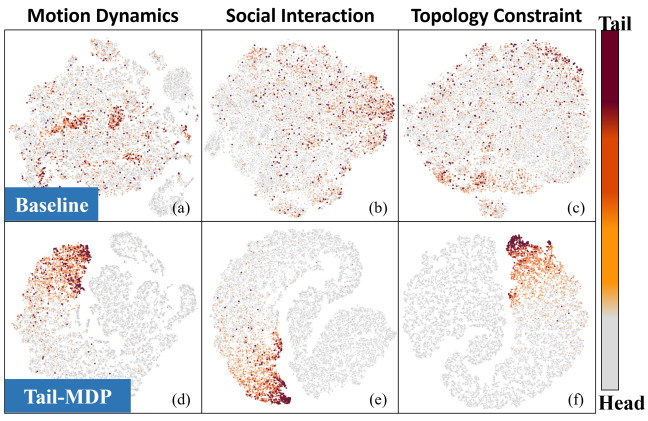

Figure 6: Visualization of feature spaces. Darker colors indicate larger reconstruction errors.