Den-TP: A Density-Balanced Data Curation and Evaluation Framework for Trajectory Prediction

Supplementary Material

6. Implementation Details

During the dataset selection stage, we employ HiVT [53], HPNet [32], and QCNet [54] as backbones with batch sizes of 16, 4, and 4, respectively, selected using the NaiveGreedy algorithm. For model evaluation, one mainstream model and one SOTA model are chosen per dataset: HiVT and HPNet for Argoverse 1 [5], and QCNet and DeMo [43] for Argoverse 2 [38]. All models follow their official experimental settings and are trained on both selected subsets and complete datasets.

HiVT: Batch size 32, LR 3×10^{-4}, weight decay 1×10^{-4}, dropout 0.1. Its architecture includes 1 interaction module, 4 temporal and 3 global modules, 8-head attention (50 m radius), 6 prediction modes, and hidden sizes of 64/128.

HPNet: Batch size 16, LR 5×10^{-4} (same weight decay and dropout). It has 1 spatiotemporal and 2 tri-factor attention layers (50 m radius, 20-frame windows) with data augmentation: horizontal flipping (0.5), agent (0.05) and lane (0.2) occlusion.

QCNet: Trained for 64 epochs with AdamW (batch size 32, LR 5×10^{-4}, weight decay 1×10^{-4}, dropout 0.1). It uses a 128-dim hidden space, gated 8-head attention, layer normalization on MLP and attention layers, 3 iterations for the trajectory proposal, and 2 multi-context attention modules in both encoder and decoder.

DeMo: Trained for 60 epochs with AdamW (batch size 16 per GPU, LR 0.003, weight decay 0.01, cosine annealing with 10 warm-up epochs). No data augmentation is applied.

7. Additional Experiments

7.1. Additional Results on Argoverse 1 and Argoverse 2.

We present comprehensive results for different models and data retention ratios on Den-TP-selected datasets for both Argoverse 1 and Argoverse 2. The results, shown in Table 6 and Table 7, follow trends consistent with those observed in the 60%, 50%, and 40% retention settings, further validating the effectiveness of our approach.

7.2. Performance Enhancement on HPNet.

To further validate the generalization capability of the Den-TP method, we conducted the same experimental setup on HPNet. The test set was partitioned based on different scene densities, and the model was evaluated on these subdivided datasets. The results, presented in Table 8, demonstrate the effectiveness of our approach. When the agent density is below 40, our method achieves performance comparable to models trained on the full dataset, with only slight increases of 0.019 in minADE and 0.028 in minFDE, while MR remains nearly unchanged. However, in high-density scenarios where the number of agents exceeds 60, models trained on our selected subset exhibit notable improvements, with minADE and minFDE reduced by 0.001 and 0.002, respectively. This advantage becomes even more pronounced in scenarios with more than 80 agents, where minADE is reduced by approximately 0.022, minFDE by nearly 0.073, and MR by nearly 3.5%. These findings confirm that Den-TP effectively maintains dataset diversity while preserving model performance, regardless of the model used.

7.3. Low Data Retention Setting.

We further studied the extremely low-budget regime with $\alpha \in \{10, 5, 2, 1\}\%$. As shown in Table 9, the performance of all methods drops sharply below 10%. Nevertheless, our Den-TP method consistently outperforms random selection at the same budget levels. In particular, at 1% retention, Den-TP reduces minADE from 2.033 to 1.589 and MR from 0.452 to 0.386, showing that even with very limited data, informative subset selection remains beneficial. This advantage comes from prioritizing scarce high-density scenes during budget allocation, ensuring long-tailed scenarios remain represented.

7.4. Generalizability and Robustness.

To further assess the generalizability and robustness of our proposed method, we evaluate the impact of different data retention ratios on Den-TP-selected datasets using various backbone networks. The results, presented in Table 10, demonstrate that Den-TP is not only adaptable across a diverse range of trajectory prediction models but also effectively reduces dataset size while maintaining model performance across different feature extractors. These findings underscore Den-TP's robustness and broad applicability in real-world autonomous driving scenarios.

Partition Interval. Given the number of agents in different trajectory prediction scenarios varies significantly, we examine the impact of scenario density partition intervals on the selected subset to validate the generalizability of our method. Specifically, we divide the scenarios in the Argoverse 1 dataset based on agent counts with partition intervals τ of {5, 10, 20}, forming multiple scenario density categories. Within each category, we perform data selection. As shown in Table 11, the subsets selected using different

Methods	$\alpha(\%)$	HiVT-64			HiVT-128			HPNet		
		mADE↓	mFDE↓	MR↓	mADE↓	mFDE↓	MR↓	mADE↓	mFDE↓	MR↓
Random	30	0.76	1.21	0.13	0.73	1.13	0.12	0.69	0.98	0.09
Cluster		0.74	1.14	0.12	0.71	1.07	0.11	0.68	0.97	0.08
Herding		0.73	1.14	0.12	0.71	1.07	0.11	0.69	0.96	0.09
Den-TP		**0.72**	**1.12**	**0.12**	**0.70**	**1.06**	**0.11**	**0.68**	**0.95**	**0.07**
Random	20	0.78	1.25	0.14	0.75	1.18	0.13	0.71	1.01	0.10
Cluster		0.76	1.21	0.14	0.72	1.11	0.12	0.69	0.98	0.09
Herding		0.76	1.19	0.13	0.74	1.14	0.13	0.70	0.99	0.09
Den-TP		**0.74**	**1.16**	**0.12**	**0.72**	**1.11**	**0.11**	**0.69**	**0.97**	**0.08**
Random	10	0.84	1.40	0.16	0.80	1.29	0.15	0.88	1.40	0.18
Cluster		0.82	1.37	0.16	0.76	1.20	0.13	0.87	1.39	0.17
Herding		0.80	1.29	0.14	0.80	1.29	0.14	0.81	1.26	0.13
Den-TP		**0.78**	**1.25**	**0.13**	**0.75**	**1.17**	**0.12**	**0.78**	**1.17**	**0.12**

Table 6. Performance comparison results on Argoverse 1 with data retention ratios of 30%, 20%, and 10%. Pretrained HiVT-64 is used for sample selection. Evaluation conducted on HiVT-64, HiVT-128, and HPNet.

Methods	$\alpha(\%)$	QCNet			DeMo		
		mADE↓	mFDE↓	MR↓	mADE↓	mFDE↓	MR↓
Random	30	0.827	1.486	0.224	0.779	1.547	0.216
Cluster		0.821	1.523	0.222	0.765	**1.498**	0.207
Herding		0.819	1.505	0.213	0.762	1.506	0.204
Den-TP		**0.794**	**1.453**	**0.192**	**0.743**	1.501	**0.192**
Random	20	0.878	1.586	0.242	0.827	1.645	0.238
Cluster		0.843	1.561	0.222	0.794	1.618	0.229
Herding		0.844	1.545	0.222	0.801	1.625	0.227
Den-TP		**0.832**	**1.530**	**0.211**	**0.783**	**1.591**	**0.212**
Random	10	0.944	1.797	0.272	0.918	1.817	0.263
Cluster		0.911	1.772	0.251	0.878	1.779	0.247
Herding		0.908	1.743	0.246	0.882	1.774	0.243
Den-TP		**0.891**	**1.684**	**0.241**	**0.871**	**1.770**	**0.241**

Table 7. Performance comparison results on Argoverse 2 with data retention ratios of 30%, 20%, and 10%. Pretrained QCNet is used for sample selection. Evaluation conducted on QCNet and DeMo.

partition intervals result in comparable model performance, with minimal variations in minADE and minFDE. This consistency across different settings highlights the robustness of our method. This suggests that our approach generalizes well to other datasets. When applying this method, the partition interval can be adjusted based on the dataset characteristics: if the dataset has relatively few agents per scenario, a smaller interval is preferable; whereas for datasets with a high variance in agent count, a larger interval may be more suitable to better balance scenario density.

8. Ablation Study

Effectiveness of Submodular Gain. To isolate the effect of submodular gain, we conducted an experiment where data selection was based solely on submodular importance scores, without considering scene density balancing, as shown in Table 4 line 3. The results indicate that using only submodular selection achieves a minADE of 0.724, lower than the 0.741 obtained through random selection, demonstrating that submodular-based sample selection improves data quality. However, it still underperforms compared to our full method. This is because prioritizing sample informativeness without adjusting for scene density leads to a dataset biased toward certain complexity levels, ultimately hindering the model's generalization ability. In contrast, our full method, which integrates scene balancing with submodular gain, achieves the best performance across all metrics. These findings highlight the necessity of jointly considering both scene distribution balance and sample informativeness to construct an effective training dataset.

Impact of Pretrained Backbone Epochs. To examine the

HPNet	Agent<40			Agent>=40			Agent>=60			Agent>=80		
	mADE↓	mFDE↓	MR↓	mADE↓	mFDE↓	MR↓	mADE↓	mFDE↓	MR↓	mADE↓	mFDE↓	MR↓
Full	0.611	0.833	0.062	0.875	1.202	0.127	1.111	1.463	0.201	1.596	1.771	0.276
Random	0.652	0.903	0.071	0.932	1.336	0.147	1.267	1.850	0.274	1.649	2.086	0.375
Cluster	0.642	0.889	0.069	0.916	1.291	0.143	1.193	1.669	0.250	1.636	1.977	0.325
Herding	0.650	0.901	0.070	0.921	1.303	0.145	1.215	1.673	0.252	1.637	2.001	0.342
Den-TP	0.630	0.861	0.064	0.878	1.210	0.132	**1.110**	**1.461**	**0.198**	**1.574**	**1.698**	**0.241**

Table 8. Performance Comparison of HPNet across different scene densities when trained on different sample set with $\alpha = 50\%$.

$\alpha(\%)$	Method	mADE↓	mFDE↓	MR↓
10	Random	0.818	1.329	0.167
	Den-TP	**0.781**	**1.253**	**0.132**
5	Random	0.869	1.484	0.183
	Den-TP	**0.843**	**1.416**	**0.161**
2	Random	1.055	1.902	0.245
	Den-TP	**1.015**	**1.871**	**0.220**
1	Random	2.033	4.613	0.452
	Den-TP	**1.589**	**3.416**	**0.386**

Table 9. Performance comparison under extremely low data retention settings ($\alpha \leq 10\%$).

Backbone	$\alpha(\%)$	mADE↓	mFDE↓	MR↓	mADE↓	mFDE↓	MR↓
HiVT-64	10	0.786	1.254	0.138	0.782	1.172	0.121
	20	0.749	1.167	0.125	0.691	0.970	0.081
	30	0.727	1.120	0.120	0.678	0.951	0.079
	40	0.712	1.089	0.114	0.671	0.931	0.076
	50	0.704	1.073	0.111	0.661	0.913	0.074
	60	0.703	1.065	0.111	0.654	0.901	0.072
HPNet	10	0.796	1.277	0.143	0.739	1.050	0.095
	20	0.753	1.193	0.125	0.699	0.975	0.082
	30	0.732	1.143	0.119	0.681	0.943	0.078
	40	0.717	1.087	0.112	0.670	0.924	0.075
	50	0.708	1.079	0.111	0.664	0.917	0.074
	60	0.703	1.067	0.108	0.657	0.910	0.075

Table 10. Performance of HiVT-64 and HPNet trained on subsets selected at varying data retention ratios.

Epoch	0	5	8	10	15	64
mADE↓	0.713	**0.704**	0.707	0.708	0.710	0.712
mFDE↓	1.083	**1.073**	1.076	1.074	1.083	1.080
MR↓	0.112	**0.111**	0.111	0.111	0.111	0.111

Table 12. Performance of models pretrained for different numbers of epochs before Den-TP selection (50% subset).

Partition	mADE↓	mFDE↓	MR↓
$\tau = 5$	0.703	**1.056**	**0.110**
$\tau = 10$	**0.702**	1.064	0.111
$\tau = 20$	0.707	1.081	0.113

Table 11. Performance of Den-TP with different density partition intervals τ on Argoverse 1.

influence of the pretrained backbone on subset selection, we conducted a series of experiments using models initialized identically but trained with different numbers of pretraining epochs. Taking HiVT-64 as an example, the official training setup involves training the model for 64 epochs using the full dataset. In our experiments, we varied the number of pretraining epochs as {0, 5, 8, 10, 15, 64} and analyzed its impact on subset selection, as shown in Table 12. The results indicate that moderate pretraining is crucial for effective subset selection. When the number of pretraining epochs is set to 5, the subset selection achieves optimal performance, consistently outperforming other configurations across all data retention ratios. As the pretraining epochs increase, subset selection continues to provide significant advantages over other data selection methods but does not surpass the performance observed at epoch 5. For models without pretraining, minADE degrades noticeably compared to models pretrained for 5 epochs. In contrast, at 64 pretraining epochs, as the model has already converged, the impact of sample selection on gradient updates diminishes. Although subset selection performance remains competitive, it does not yield further improvements over moderate pretraining.

9. Qualitative Results

In Figure 7 and Figure 8, we present trajectory samples from three different scene densities in Argoverse 1 and Argoverse 2. In the top row, the scenes feature only a few agents, making them relatively straightforward for the model to predict. Because low-density scenarios dominate the dataset, the model tends to be biased toward these simpler cases, and the lower interaction complexity naturally reduces motion-forecasting uncertainty. In the middle rows,

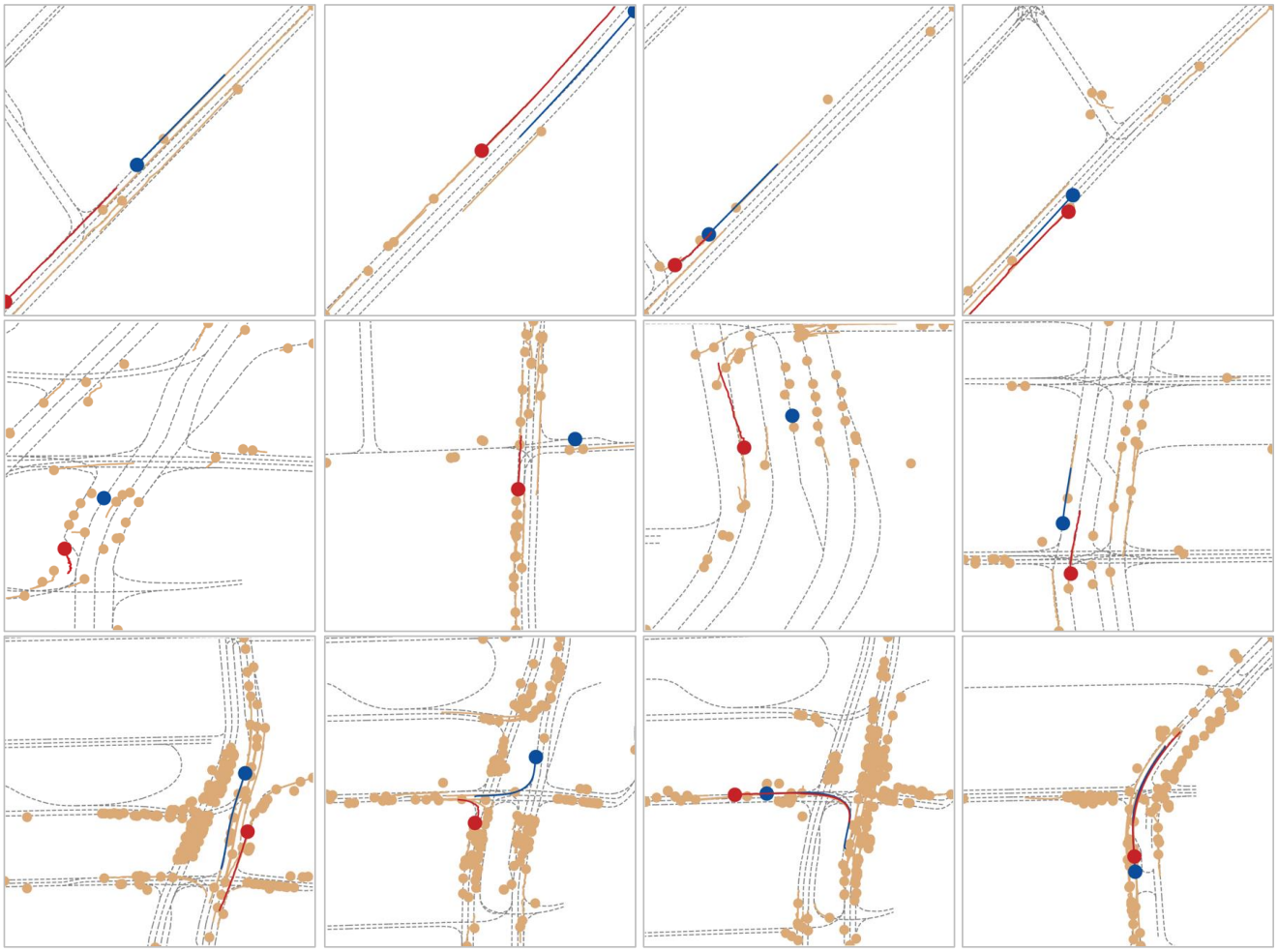

Figure 7. Visualization of different agent density scenarios in Argoverse 1.

moderate-density scenarios display an increased number of agents. Here, interactions among vehicles become more frequent, creating added complexity for trajectory prediction. In the bottom row, high-density scenes pose significant challenges for trajectory prediction models. These crowded urban intersections and multi-agent interactions are underrepresented in standard datasets. The large number of dynamic agents in these scenarios compounds uncertainty, making it harder for the model to produce accurate predictions. Yet, these are precisely the scenarios that are most critical for ensuring safe autonomous driving. Our Den-TP method addresses this imbalance head-on by emphasizing high-density samples. Through a more balanced yet compact selection of training data, the model becomes well-prepared for both common and complex scenarios. As a result, Den-TP bolsters model robustness in high-density environments, leading to more reliable trajectory predictions in real-world urban traffic conditions.

LLM Usage

We used ChatGPT solely for grammar correction and LaTeX formatting. They were not involved in research ideation, experiment design, or data analysis. All scientific contributions, methodology, and results are entirely the work of the authors.

Figure 8. Visualization of different agent density scenarios in Argoverse 2.