

# TAROT: Targeted Data Selection via Optimal Transport

Lan Feng<sup>1\*</sup>, Fan Nie<sup>2\*</sup>, Yuejiang Liu<sup>2</sup>, Alexandre Alahi<sup>1</sup>  
<sup>1</sup>EPFL, <sup>2</sup>Stanford University

{lan.feng, alexandre.alahi}@epfl.ch, {niefan, yuejiang.liu}@stanford.edu

## Abstract

We propose TAROT, a *Targeted data selection framework grounded in Optimal Transport theory*. Previous targeted data selection methods primarily use influence-based greedy heuristics to enhance domain-specific performance. These methods perform well on limited, unimodal data (i.e., data following a single pattern) but become less effective as target data increases in complexity. Specifically, in multimodal distributions, these heuristics fail to account for multiple inherent patterns, leading to suboptimal data selection. This work identifies two primary factors contributing to this limitation: (i) the disproportionate impact of dominant feature components in high-dimensional influence estimation, and (ii) the restrictive linear additive assumptions inherent in greedy selection strategies. To address these challenges, TAROT incorporates whitened feature distance to mitigate dominant feature bias, offering a more reliable measure of data influence. Building on this, TAROT uses whitened feature distance to quantify and minimize the optimal transport distance between the selected data and target domains. Notably, this minimization also facilitates the estimation of optimal selection ratios. We evaluate TAROT across multiple tasks, including semantic segmentation, motion prediction, and instruction tuning. Results consistently show that TAROT outperforms state-of-the-art methods, highlighting its versatility across various deep learning tasks. Code is available at <https://github.com/vita-epfl/TAROT>.

## 1. Introduction

Expanding model capacity and data volume has emerged as a key strategy in both computer vision and large language models [1, 15, 33, 47]. However, training a model on all available data may not be optimal or feasible, as data quality varies and computational resources are often limited [2, 13, 25, 29]. To address this, recent research has introduced targeted data selection methods. These meth-

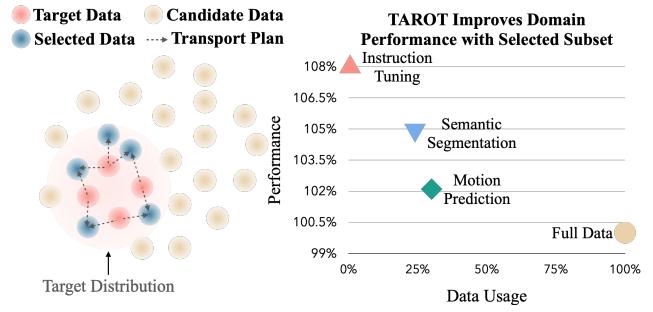


Figure 1. **Left:** this figure demonstrates the selection objective of TAROT, designed to minimize optimal transport costs. The method operates on a **target distribution** and a **candidate data pool**, selecting a **subset** from the candidate data to reduce transport costs. **Right:** the figure highlights the approximate performance improvements and data efficiency achieved by TAROT across various tasks. It illustrates that TAROT consistently enhances task performance in different domains while requiring less data. Detailed experimental results are provided in Sec. 5.

ods aim to select data from a candidate pool that maximizes model performance on specific target tasks. Recent methods [13, 42] address this challenge using a two-step approach: (1) **influence estimation**, which calculates data influence scores between the candidate and target datasets, and (2) **greedy heuristics**, which selects data with the highest average influence scores for the target dataset.

Although these selection paradigms have shown to be effective in large language model (LLM) applications [13, 42], they rely heavily on the assumption that influence functions are additive. This means that the influence of multiple training samples on a target sample can be added together. However, as noted by Hu et al. [22], this assumption, while efficient and broadly applicable, fails even in simple linear regression scenarios. One main reason why the additivity assumption works well in some cases is the simple and unimodal distribution of the target domain, such as question-answer pairs from a specific domain in LLMs. In these situations, adding up the influences provides a good estimate of how a candidate sample impacts the entire target dataset. However, in other contexts like motion prediction, where data distributions are often complex and multimodal, this

\*Equal contribution.

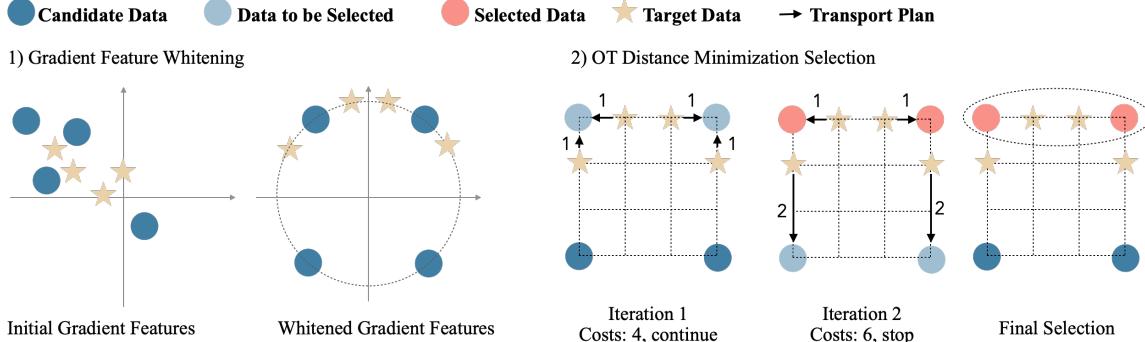


Figure 2. **Overview of TAROT.** **Left:** Raw gradient features are whitened and projected onto a unit hypersphere. **Right:** A greedy algorithm iteratively selects candidate samples. During the second iteration, adding two new distant data points increases the overall transport cost, prompting the algorithm to stop. As a result, only in-distribution data points are selected to minimize the OT distance.

approach falls short. In such cases, the summed influence does not capture the diverse nature of the target distribution, reducing its effectiveness.

To this end, we adopt a distribution-matching approach for targeted data selection. We first introduce *whitened feature distance (WFD)*, an influence estimation method designed to mitigate the bias from dominant features. Using **WFD** as pairwise cost estimation between data points, we develop TAROT, an Optimal Transport (OT)-based targeted selection framework. This method selects data that minimizes the OT distance between the target distribution and the selected samples. Fig. 1 illustrates the selection objective of our approach. Our choice of OT is also supported by findings in [24], which demonstrate that training models on source data with smaller OT distances to the target domain improve domain-specific performance.

To validate the general effectiveness of our method, we conduct experiments on semantic segmentation, motion prediction, and instruction tuning tasks. Extensive results demonstrate that our approach consistently outperforms previous methods. Notably, models trained on smaller, domain-relevant datasets often surpass those trained on larger, less relevant datasets.

In summary, our contributions are as follows:

- **Improved influence estimation:** we present **WFD** as a reliable measure of data influence.
- **Distribution matching-based data selection:** we propose TAROT, an OT-based pipeline for targeted data selection that reduces distributional differences between the target and selected datasets.
- **Broad applicability and improved performance:** extensive experiments in computer vision and LLMs demonstrate that TAROT consistently outperforms existing methods, allowing models trained on smaller, targeted datasets to match or surpass the performance of those trained on full datasets.

## 2. Related Work

**Data Influence Estimation.** Pruthi et al. [35] proposed a method to estimate data influence by tracing the gradient descent process. Another approach utilizes *datamodels*: Ilyas et al. [23] showed that simple linear datamodels can predict model outputs effectively. Building on this, Park et al. [34] introduced a scalable technique for estimating datamodel scores. Subsequent works have optimized these methods, focusing on checkpoint ensembles [11], efficient gradient storage [46], and scalable gradient projection [7]. Our method extends Pruthi et al. [35] by enhancing estimation robustness through feature covariance whitening.

**Coresets and Targeted Selection.** Coreset selection aims to identify a dataset subset for optimal in-domain performance [4, 17, 27, 31, 43, 44], assuming validation and candidate data share the same distribution. Our work shifts focus to targeted selection, optimizing performance on a distinct target domain. Prior methods often rely on influence-estimation heuristics: Engstrom et al. [13] leveraged linear datamodel scores [34] to select data for LLMs pretraining; Xia et al. [42] leveraged influence estimation to select data for LLMs fine tuning. Hu et al. [22] highlighted these methods’ limitations and proposed iterative heuristics. Our approach introduces a novel distribution-matching framework to address these challenges.

**Optimal Transport for Data Selection.** Optimal transport has been used for dataset similarity estimation [3] and sensitivity analysis in data valuation [24]. Kang et al. [25] applied it to optimize data selection ratios across multiple sources. We integrate influence estimation with optimal transport, enabling effective subset selection to boost target domain performance.

### 3. Preliminary

#### 3.1. Data Influence Formulation

Following Pruthi et al. [35], we summarize how to estimate the influence of a training data point on held-out data using a first-order approximation of training dynamics.

**First-order Approximation to Per-step Influence.** Suppose  $x_i$  is the only training sample at time step  $t$ , and the model  $\theta^t$  is updated based on the loss  $\mathcal{L}(x_i; \theta^t)$  using SGD with learning rate  $\eta_t$ . The per-step influence of  $x_i$  on a target sample  $x$  is defined as the reduction in the loss of  $x$  caused by training on  $x_i$ . This can be approximated using a first-order Taylor expansion. The per-step influence is given by:

$$-\eta_t \langle \nabla \mathcal{L}(x_i; \theta^t), \nabla \mathcal{L}(x; \theta^t) \rangle, \quad (1)$$

where  $\nabla \mathcal{L}(x_i, x; \theta^t)$  is the loss gradient.

**Trajectory Influence via Checkpoints.** For practicality, Pruthi et al. [35] compute the trajectory influence by summing the per-step influences over epochs:

$$\text{Inf}_{\text{SGD}}(x_i, x) \triangleq \sum_{i=1}^N \bar{\eta}_i \langle \nabla \mathcal{L}(x; \theta_i), \nabla \mathcal{L}(x_i; \theta_i) \rangle. \quad (2)$$

#### 3.2. Background on Optimal Transport

Given a complete, separable metric space  $\mathcal{X}$  and probability measures  $\alpha, \beta \in \mathcal{P}(\mathcal{X})$ , the Kantorovich formulation [26] of OT is defined as:

$$\text{OT}(\alpha, \beta) \triangleq \min_{\pi \in \Pi(\alpha, \beta)} \int_{\mathcal{X} \times \mathcal{X}} c(x, y) d\pi(x, y), \quad (3)$$

where  $c(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$  is a cost function, and  $\Pi(\alpha, \beta)$  is the set of couplings with marginals  $\alpha$  and  $\beta$ .

In practice, the measures  $\alpha$  and  $\beta$  are approximated using finite samples  $\{\mathbf{x}^{(i)}\}$  and  $\{\mathbf{y}^{(j)}\}$ , with corresponding discrete measures  $\alpha = \sum_{i=1}^n a_i \delta_{\mathbf{x}^{(i)}}$  and  $\beta = \sum_{j=1}^m b_j \delta_{\mathbf{y}^{(j)}}$ . The pairwise costs form an  $n \times m$  matrix  $\mathbf{C}$ , where  $C_{ij} = c(\mathbf{x}^{(i)}, \mathbf{y}^{(j)})$ . Solving this problem as a linear program is computationally expensive, with cubic complexity. In implementation, we use the Sinkhorn algorithm [10] for efficient calculation.

## 4. Method

The failure modes of *influence-based greedy heuristics* have been systematically studied in [22]. Specifically, errors in the influence function and the inability to account for the non-additive nature of collective influence can cause these heuristics to fail even in simple linear regression tasks.

In contrast, we frame targeted data selection as a distribution matching problem, aiming to select a subset of candidate data that closely aligns with the target distribution.

We use the OT distance to quantify the discrepancy between distributions.

Secs. 4.1 and 4.2 discuss how to measure the OT distance between datasets, and Sec. 4.3 discusses how to select data to minimize the OT distance. Fig. 2 shows an overview of these two processes.

#### 4.1. Optimal Transport Between Datasets

As discussed in [3], estimating OT distance between datasets begins with defining a distance metric between data pairs  $(x, y)$  and  $(x', y')$ . In Sec. 3.1, we introduced a gradient-based method for estimating data influence, which embeds both input features and labels into the model’s gradient. This gradient feature captures the task-specific relationship between data points and the model’s loss. By computing distances such as the L2 distance, it provides a viable metric for the OT problem. Furthermore, it aligns with the model’s optimization objective, ensuring that the OT distance reflects meaningful differences between datasets in terms of their impact on model training.

Formally, we define a dataset  $\mathcal{D}$  as a set of feature-label pairs  $(x, y) \in \mathcal{X} \times \mathcal{Y}$ , where  $\mathcal{X}$  is the feature space and  $\mathcal{Y}$  is the label set. For convenience, we use the shorthand  $z \triangleq (x, y)$  and  $\mathcal{Z} \triangleq \mathcal{X} \times \mathcal{Y}$ . We consider a candidate dataset  $\mathcal{D}_c = \left\{ z_c^{(i)} \right\}_{i=1}^N$  and a target dataset  $\mathcal{D}_t = \left\{ z_t^{(i)} \right\}_{i=1}^M$ .

We embed  $z$  using the model’s parameters trained on  $\mathcal{D}_c$ :

$$\phi(z) = \sum_{i=1}^T \nabla \mathcal{L}(z; \theta_i), \quad (4)$$

where  $T$  is the total number of checkpoints, and  $\nabla \mathcal{L}(z; \theta_i)$  represents the gradient of the loss function with respect to the model parameters at the  $i$ -th checkpoint  $\theta_i$ .

The distance between two samples is then defined as:

$$d_{\mathcal{Z}}(z, z') = \|\phi(z) - \phi(z')\|_2. \quad (5)$$

#### 4.2. Whitened Feature Distance

Directly computing the distance  $d_{\mathcal{Z}}(z, z')$  as defined in Eq. (5) presents significant challenges due to the correlation of gradient features. This correlation results in an ill-conditioned covariance matrix, allowing certain feature space directions to dominate distance estimation and distort sample relationships. Additionally, varying gradient component scales bias distance measurements, impairing the accuracy of distance estimation.

To address these issues, we propose a novel application of *whitening* to the gradient features. Whitening effectively decorrelates the features and scales them to have unit variance, ensuring that each feature contributes equally to the distance computation. Our method comprises three steps:

- 1. Random projection.** We adopt the same random projection technique from [34, 42] to reduce the dimensionality of the gradients. The projected gradient  $\phi(z)_i^{\text{proj}}$  is computed as:

$$\phi(z)_i^{\text{proj}} = \mathcal{P}^T \cdot \phi(z)_i, \quad (6)$$

where  $\mathcal{P}^T$  is the projection matrix. We leverage the efficient CUDA implementation from Park et al. [34].

- 2. Whitening.** Despite dimensionality reduction, projected gradients remain correlated, creating an anisotropic feature space. To address this, we perform Cholesky whitening to decorrelate features. First, we center the projected gradients  $\Phi(z)^{\text{proj}} = [\phi(z)_1^{\text{proj}}, \dots, \phi(z)_N^{\text{proj}}]$ :

$$\tilde{\phi}(z)_i^{\text{proj}} = \phi(z)_i^{\text{proj}} - \frac{1}{N} \sum_{i=1}^N \phi(z)_i^{\text{proj}}. \quad (7)$$

Next, we compute the covariance matrix:

$$\Sigma = \frac{1}{N} \sum_{i=1}^N \tilde{\phi}(z)_i^{\text{proj}} (\tilde{\phi}(z)_i^{\text{proj}})^\top, \quad (8)$$

and apply Cholesky decomposition,  $\Sigma = LL^\top$ , where  $L$  is lower triangular. The gradients are whitened as:

$$\phi(z)^w = L^{-1} \tilde{\phi}(z)^{\text{proj}}, \quad (9)$$

ensuring decorrelation.

- 3. Normalization.** Each whitened gradient vector  $\phi(z)_i^w$  is normalized to unit length to ensure consistent gradient scales:

$$\hat{\phi}(z)_i = \frac{\phi(z)_i^w}{\|\phi(z)_i^w\|}. \quad (10)$$

The **WFD**  $d_{\mathcal{Z}}^w(z, z')$  is then calculated as:

$$d_{\mathcal{Z}}^w(z, z') = \|\hat{\phi}(z) - \hat{\phi}(z')\|_2. \quad (11)$$

This gradient-based distance also applies to data influence estimation. Our experiments show that **WFD** outperforms the state-of-the-art method [34], as detailed in Sec. 5.1.

Using  $d_{\mathcal{Z}}^w$ , we lift the metric to a dataset-level OT distance:

$$d_{\text{OT}}(\mathcal{D}_c, \mathcal{D}_t) = \min_{\pi \in \Pi(\alpha, \beta)} \int_{\mathcal{Z} \times \mathcal{Z}} d_{\mathcal{Z}}^w(z, z') \pi(z, z'). \quad (12)$$

### 4.3. Data Selection via OT Distance Minimization

Our objective is to select a subset  $\mathcal{D}_s \subseteq \mathcal{D}_c$  of size  $S \leq N$ , such that the OT distance between the selected subset  $\mathcal{D}_s$  and  $\mathcal{D}_t$  is minimized.

Formally, let  $\mathcal{D}_s = \{z_s^{(i)}\}_{i=1}^S$  be a subset of  $\mathcal{D}_c$ . We aim to find  $\mathcal{D}_s$  such that:

$$d_{\text{OT}}(\mathcal{D}_s, \mathcal{D}_t) < d_{\text{OT}}(\mathcal{D}_c, \mathcal{D}_t), \quad (13)$$

where  $d_{\text{OT}}$  is subject to the constraint that the masses for each point are normalized.

Solving the subset selection problem to minimize the OT distance is inherently combinatorial, as the number of possible subsets  $\mathcal{D}_s \subseteq \mathcal{D}_c$  grows exponentially with the size of  $\mathcal{D}_c$ . An exhaustive search incurs prohibitive computational costs. To address this, we propose a greedy algorithm that efficiently approximates the solution by iteratively selecting elements that locally minimize the OT distance.

Our method exploits the sparsity of gradient-based features in high-dimensional space, which results in a sparse coupling matrix in the OT problem. Specifically, most of the transport mass in the OT plan is concentrated on the nearest neighbors of target points in  $\mathcal{D}_t$ . Leveraging this property, the algorithm iteratively selects the nearest neighbors from  $\mathcal{D}_c$  for each point in  $\mathcal{D}_t$ , providing a computationally efficient solution.

Based on this, we introduce two selection schemes tailored for different scenarios: (1) fixed-size selection for maximizing performance under a limited budget, and (2) OT distance minimization selection to find a potentially optimal selection ratio.

**Fixed-Size Selection.** This scheme selects a fixed number of points  $S$  from  $\mathcal{D}_c$ . For  $M$  target samples, we iteratively add their nearest candidates from  $\mathcal{D}_c$  to the selected set  $\mathcal{D}_s$ . In iteration  $k$ , the  $k$ -th nearest samples are considered. If adding all of them keeps  $|\mathcal{D}_s| \leq S$ , they are directly added, and the process continues. Otherwise, we compute the potential  $\phi_i$  of the new candidates using the dual OT problem:

$$\min_{\pi \in \Pi(\mathcal{D}_s \cup \{z_i\}, \mathcal{D}_t)} \int_{\mathcal{D}_s \cup \{z_i\} \times \mathcal{D}_t} c(z, z_t) d\pi(z, z_t), \quad (14)$$

where  $\Pi(\mathcal{D}_s \cup \{z_i\}, \mathcal{D}_t)$  is the set of couplings between  $\mathcal{D}_s \cup \{z_i\}$  and  $\mathcal{D}_t$ . As noted in [24],  $\phi_i$  estimates the benefit of adding  $z_i$  for minimizing the OT distance. We rank candidates by potential and select the top  $S - |\mathcal{D}_s|$  samples. The process ends once these samples are added.

**OT-Distance Minimization Selection (OTM).** This scheme minimizes the OT distance between the selected data  $\mathcal{D}_s$  and the target distribution. To avoid overfitting to specific points in  $\mathcal{D}_t$ , we use a  $k$ -fold selection strategy with cross-validation. The target dataset  $\mathcal{D}_t$  is randomly split into  $k$  equal subsets. In each fold,  $1/k$  of  $\mathcal{D}_t$  is used for selection, while the OT distance is evaluated against the remaining  $(k-1)/k$  data.

At each iteration, points from  $\mathcal{D}_c$  are selected based on their nearest-neighbor distances to  $1/k$  of  $\mathcal{D}_t$ , and the OT

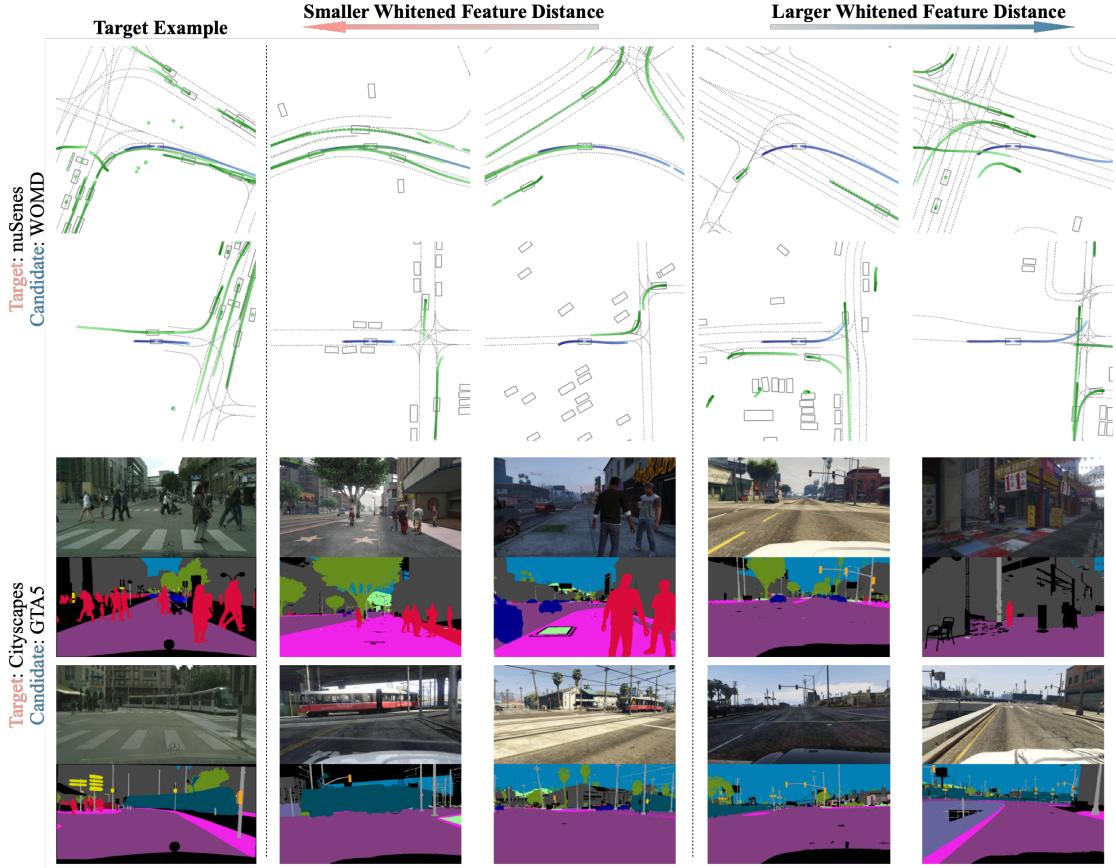


Figure 3. Illustration of target examples and their corresponding closest (most helpful) and furthest (most detracting) candidate examples, identified by **WFD**. **Top two rows:** Visualization of motion prediction tasks using the nuScenes and Waymo datasets. The dark-to-light blue lines indicate the past (input) and future (ground truth) trajectories for the target vehicle, while green lines represent surrounding vehicle trajectories. **Bottom two rows:** Semantic segmentation tasks with examples from the Cityscapes and GTA5 datasets. For each target example, the top image shows the RGB scene, and the bottom image depicts the corresponding ground truth semantic segmentation mask. The **WFD** method selects training examples that are semantically aligned with the target, demonstrating its effectiveness in identifying positive and negative samples across different tasks.

distance is monitored. The process stops when the OT distance increases:

$$d_{\text{OT}}(\mathcal{D}_s, \mathcal{D}_t \setminus \mathcal{D}_t^{(fold)}) > d_{\text{OT}}(\mathcal{D}_s^{(prev)}, \mathcal{D}_t \setminus \mathcal{D}_t^{(fold)}), \quad (15)$$

where  $\mathcal{D}_t^{(fold)}$  is the  $1/k$  subset used for selection.

After processing all  $k$  folds, the selected subsets are combined to form the final dataset  $\mathcal{D}_s$ . The choice of  $k$  balances distribution alignment and generalization. In our experiments,  $k = 10$  ensures a good match with the target distribution while avoiding overfitting.

#### 4.4. Data Weighting with OT Potential

To highlight the importance of individual data points, we assign weights based on their OT potentials from Eq. (14). These potentials are scaled to positive integer weights, determining the number of times each point is duplicated within an epoch. The scaled weights  $\{w_i\}$  are normalized

to satisfy  $\sum_{i=1}^S w_i = R$ , where  $R$  is a customizable repetition factor. For instance,  $R$  can be set to  $N + M$  to match the same training steps as using the full dataset, or adjusted based on specific training requirements, e.g., limited training budget.

## 5. Experiments

This section starts by evaluating the influence estimation performance of **WFD** in Sec. 5.1. Following that, we evaluate TAROT across diverse tasks, including semantic segmentation (Sec. 5.2), motion prediction (Sec. 5.3), and instruction tuning (Sec. 5.4).

### 5.1. WFD for Influence Estimation

**Evaluation Metric.** Following Park et al. [34], we use the Linear Data Modeling Score (LDS) as our evaluation metric. LDS measures the effectiveness of an attribution or in-

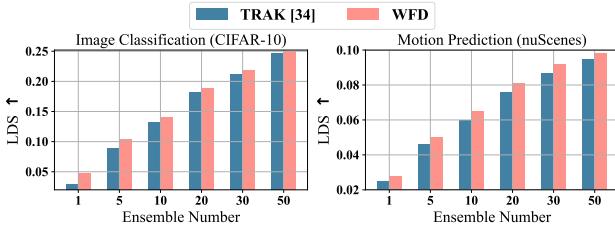


Figure 4. **WFD** outperforms baseline methods on both tasks. The x-axis indicates the number of model checkpoints that a given method uses to compute attribution scores. The y-axis indicates the method’s efficacy as measured by the linear datamodeling score (LDS).

fluence estimation method in predicting changes in model outputs when the training set is altered. Further details of LDS are provided in the supplementary materials.

**Target and Candidate Datasets.** Following Park et al. [34], we evaluate image classification using ResNet-9 classifiers trained on the CIFAR-10 dataset. For motion prediction, we adopt AutoBots [16], training on the nuScenes [5] dataset (32k samples) and validating on 9k target samples.

**Baselines.** We compare **WFD** against the state-of-the-art data attribution method **TRAK** [34] using the same ensemble approach. Since **WFD** quantifies distance (where smaller values indicate greater influence), we evaluate it with the LDS score by considering negative **WFD** values. Additionally, we employ ZCA whitening instead of Cholesky whitening to preserve data structure for improved LDS performance. Differences between these two whitening methods are discussed in the supplementary materials.

**Main Results.** Fig. 4 compares the performance of **WFD** with **TRAK** on the LDS score. Key findings include:

- **WFD outperforms TRAK regardless of ensemble sizes.** This indicates that **WFD** exhibits a higher correlation with datamodels, which serves as an “oracle” of sorts for the LDS objective.
- **Inspecting WFD-identified examples.** In Fig. 3 we display, for four randomly selected target samples from the nuScenes [5] and Cityscapes [9] dataset, the training examples from WOMD [14] and GTA5 [36] dataset corresponding to the smallest (most positive) and largest (most negative) whitened feature distance.

## 5.2. TAROT for Semantic Segmentation

Semantic segmentation, a key task in computer vision, involves assigning semantic labels to every pixel in an image. Prior studies show that synthetic data can significantly boost segmentation model performance on real-world data [36]. In this section, we address two research questions:

- Is all synthetic data equally effective for improving real-world segmentation performance?
- Can targeted selection methods identify synthetic data

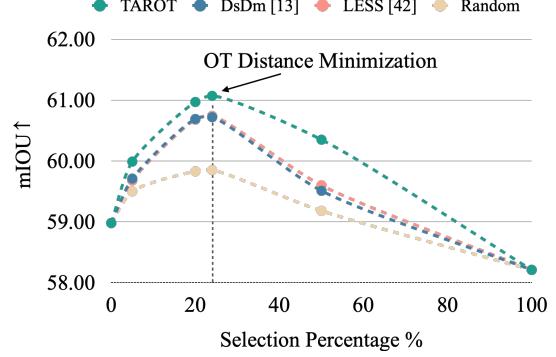


Figure 5. **Results of Targeted Selection for Semantic Segmentation.** Performance curves of DeepLabV3 trained using different data selection methods.

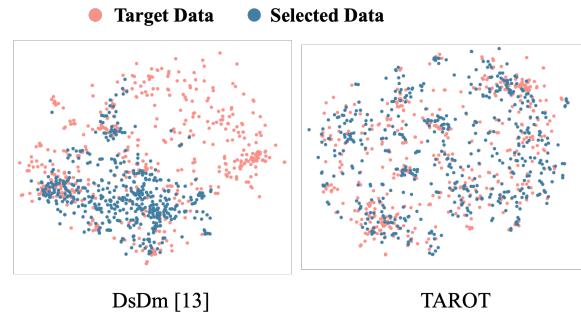


Figure 6. **Data selection results visualized in the gradient feature space using T-SNE.** We use Cityscapes’ validation set as the target, we select 10% of its training set. **Left:** DsDm, where selected samples cluster tightly around the target distribution center. **Right:** TAROT, where selected samples are distributed more broadly, capturing the complexity of the target distribution.

that better align with real-world distributions to enhance model performance?

**Target and Candidate Datasets.** The GTA5 dataset [36] serves as the candidate dataset, while the Cityscapes [9] training split (2975 samples) is used as the target dataset, with its validation split for evaluation. GTA5 contains 24,966 synthetic images with pixel-level annotations of urban scenes, whereas Cityscapes provides finely annotated real-world urban images.

**Baselines.** We compare TAROT with state-of-the-art targeted selection methods LESS and DsDm [13, 42], as well as random selection.

**Data Selection and Model Training.** We employ DeepLabV3 [6] with a ResNet50 [19] backbone, using gradients from the last four checkpoints to guide data selection. We evaluate selection ratios of 5%, 20%, 50% and OTM (Sec. 4.3). OTM selects approximately 24% of the data. The repetition factor is set as:  $R = N + M$  for data selected by TAROT (Sec. 4.4), while other methods employ uniform repetition to match the same size.

A new DeepLabV3 model is then trained from scratch on the selected subsets for 50 epochs with a batch size of 16. No data augmentation is applied, and images are resized to  $920 \times 460$ .

**Evaluation Metric.** Mean Intersection over Union (mIoU) is used as the primary evaluation metric. We report the best validation performance during training.

**Main Results.** Fig. 5 compares the performance of TAROT and other baselines. Key findings include:

- **Some synthetic data can degrade model performance.** Training model on the full synthetic GTA5 dataset lowers mIoU from 58.98 to 58.21, highlighting the negative impact of misaligned synthetic data.
- **Targeted data selection enhances domain-specific performance.** Model performance improves with increased selection ratios but declines beyond an optimal point. Notably, TAROT consistently outperforms baselines, achieving the highest mIoU with about 24% of the GTA5 dataset. This demonstrates the effectiveness of OT distance in guiding optimal selection ratios.
- **Additional insights from T-SNE visualizations.** Fig. 6 shows T-SNE visualizations of gradient features for selected data. The visualizations reveal differing selection behaviors: DsDm tightly clusters around the target distribution center, failing to capture its complexity. In contrast, TAROT selects data that better aligns with the broader target distribution, highlighting a key limitation of averaging-based influence methods in handling complex distributions.

### 5.3. TAROT for Motion Prediction

Motion prediction, the task of forecasting future trajectories from past data, is a critical component of autonomous driving. This section addresses two key research questions:

- **Is TAROT transferable?** For example, can a model trained on data selected by a smaller model improve the performance of a larger model?
- **Can targeted selection improve the performance of state-of-the-art motion prediction models on a target domain?**

**Target and Candidate Datasets.** We use the UniTraj framework [15] for unified training and evaluation across multiple datasets, including Waymo Open Motion (WOMD) [14], Argoverse 2 [41], nuScenes [5], and nuPlan [18].

This experiment focuses on improving model performance on the nuScenes validation set through targeted data selection. The nuScenes training set ( $32k$  samples) serves as the target dataset, while the candidate pool comprises WOMD, Argoverse 2, and nuPlan. From nuPlan, we filter trajectories with a moving distance over 2 meters, yielding  $1000k$  samples. WOMD and Argoverse 2 use their official training splits, containing  $1800k$  and  $180k$  samples, respectively.

Table 1. **nuScenes Leaderboard.** Evaluated with minADE5 metric: the average of L2 distances between the predicted trajectory and ground truth over the 5 most likely predictions. *Wayformer-All* denotes Wayformer trained with the full candidate dataset (WOMD, Argoverse 2 and nuPlan).

Method	Ranking (↓)	minADE5 (↓)
Wayformer-TAROT	1	<b>0.92</b>
Wayformer-All [32]	2	0.94
MTR-UniTraj [15]	3	0.96
Goal-LBP [45]	4	1.02
SemanticFormer [37]	5	1.14

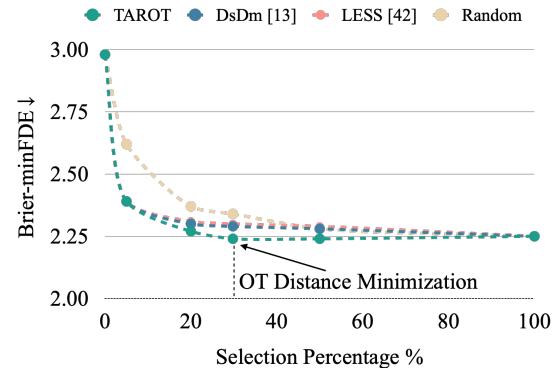


Figure 7. **Results of Targeted Selection for Motion Prediction.** The figure illustrates the performance of Wayformer under various data selection methods.

tively. The evaluation is conducted on the nuScenes validation set. This cross-dataset task involves challenges like non-overlapping geographic regions and varying annotation formats. We explore whether TAROT can effectively curate multi-source data to enhance nuScenes validation performance.

**Baselines.** We compare against the same baselines in Sec. 5.2.

**Data Selection and Model Training.** We employ AutoBots [16], a lightweight ( $1.5M$  parameters) attention-based prediction model, for data selection. AutoBots is pretrained on candidate and target datasets for 50 epochs, and gradients from the final checkpoint guide selection.

We evaluate selection ratios of 5%, 20%, 50%, and OTM. OTM selects approximately 31%, 53%, and 22% from WOMD, Argoverse 2, and nuPlan, covering 29.8% of the full dataset. The selected data is used to train the Wayformer [32] model ( $15M$  parameters) for 100 epochs. To highlight that TAROT can also achieve performance gains with reduced training data and iterations, the data weighting method (Sec. 4.4) is not applied.

**Evaluation Metric.** We use the Brier Minimum Final Displacement Error (Brier-minFDE) metric for evaluation, which is used in the Argoverse 2 [41] challenge. We report

Table 2. Performance of TAROT on instruction tuning for LLMs. ‘Transfer’ indicates that QWEN-2.5-7B is trained on data selected by LLAMA-3.1-8B. Otherwise, the same base model is used for both selection and fine-tuning. ‘All’ denotes employing the full dataset. 5% indicates a selection ratio of 5%. The percentages within parentheses represent the [selection ratios](#) determined by OT-Minimization. ‘Subtask label’ specifies whether subtask labels are required during data selection.

Dataset	LLAMA-3.1-8B		QWEN-2.5-7B		Transfer		Subtask label
	MMLU ↑	BBH ↑	MMLU ↑	BBH ↑	MMLU ↑	BBH ↑	
All	63.8	63.3	74.1	63.8	74.1	63.8	-
5% Random	64.7	60.8	74.1	63.5	74.1	63.5	✗
5% LESS	65.7	62.6	<b>74.3</b>	66.3	<b>74.3</b>	65.0	✓
5% TAROT	<b>66.0</b>	<b>65.0</b>	74.1	66.9	74.2	65.3	✗
TAROT-OTM	65.7 (0.13%)	63.6 (0.21%)	<b>74.3 (0.09%)</b>	<b>68.9 (0.13%)</b>	<b>74.3 (0.13%)</b>	<b>68.7 (0.21%)</b>	✗

the best validation performance for each method.

**Main Results.** Fig. 7 compares the performance of Wayformer trained on datasets with different selection ratios. Key findings include:

- **Effectiveness of TAROT:** TAROT consistently outperforms other baselines. Influence-estimation based heuristics, such as LESS and DsDm, perform well at low selection ratios (5%) but lose effectiveness as the ratio increases, emphasizing the benefits of the distribution-matching approach. Additionally, OTM selection outperforms training on the entire dataset and other selection ratios, delivering global optimal results.
- **Transferability of TAROT:** Data selected by Autobots significantly enhances the performance of Wayformer, a model with  $10\times$  the capacity. Furthermore, despite differences in loss functions and model architectures between the two models, the selected data still yields performance improvements. This highlights that although TAROT leverages a pretrained model for selection, the selected dataset retains strong model-agnostic properties.
- **Improvement on leaderboard performance.** Notably, Tab. 1 shows that Wayformer trained on TAROT-selected data achieves first place on the nuScenes leaderboard, surpassing MTR-UniTraj [15], which relies on large-scale datasets ( $2000k$  samples). These results demonstrate that targeted data selection offers a promising alternative to data scaling for breaking performance boundaries.

#### 5.4. TAROT for Instruction Tuning

Previous experiments demonstrate that TAROT is particularly effective when dealing with complex target distributions. In this section, we evaluate whether TAROT can also deliver strong performance in large language model (LLM) tasks, where the target distribution is relatively simple and existing methods have already shown high effectiveness.

**Target and Candidate Datasets.** To ensure a fair comparison and maintain consistency with established benchmarks, we replicate the experimental setup described by Xia et al. [42]. We utilize the same candidate dataset, comprising

FLAN V2 [30], CoT [40], DOLLY [8] and OPEN ASSISTANT 1 [28], with MMLU [20, 21] and BBH [38] serving as the target tasks for evaluation.

**Data Selection and Model Training.** We conduct experiments using LLAMA-3.1-8B [12] and QWEN-2.5-7B [39] as the base models for both data selection and fine-tuning. To evaluate the transferability of our selection method, we also fine-tune a QWEN-2.5-7B using data samples selected by LLAMA-3.1-8B. Data selected by TAROT-OTM is weighted using OT potentials (Sec. 4.4) with  $R = 0.5\%N$ . More details are provided in the supplementary materials.

**Evaluation Metric.** We employ Exact Matching (EM) for BBH and Accuracy for MMLU as evaluation metrics.

**Main Results.** Tab. 2 represents the performance of TAROT compared with baselines. Among them, LESS requires the subtask labels of target samples, whereas TAROT operates without this requirement. Our key findings include:

- **Effectiveness and transferability of TAROT.** TAROT consistently outperforms all baseline methods and models trained on the full dataset without relying on subtask labels. Besides, data samples selected by LLAMA-3.1-8B enhances performance for QWEN-2.5-7b, indicating strong transferability across different model architectures.
- **Effectiveness of OTM.** Remarkably, TAROT-OTM selects less than 0.5% of the data while achieving performance that is comparable to or even surpasses that of methods selecting 5% of the data. The results suggest that only a small fraction of the candidate dataset is pertinent to the target tasks, and TAROT-OTM successfully identifies these critical samples, thereby significantly reducing training costs without compromising performance.

#### 6. Conclusion

This paper presents TAROT, a general targeted data selection framework leveraging OT theory. By introducing **WFD**, TAROT mitigates biases from dominant features and enhances influence estimation. Through OT-

based selection, it aligns subsets closely with target distributions, consistently outperforming traditional methods in tasks such as semantic segmentation, motion prediction, and instruction tuning. These results demonstrate TAROT’s effectiveness in improving domain-specific performance with smaller, more relevant datasets. **Limitations:** In cases with small or highly specific target datasets, TAROT may overfit by overly tailoring the selected data, potentially compromising generalization. A promising future direction is incorporating distribution diversification into the selection objective.

## 7. Acknowledgements

The project was partially funded by Honda R&D Co., Ltd, Valeo Paris, and the Swiss National Science Foundation under the Grant P500PT 222293.

## References

- [1] 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. 1
- [2] Alon Albalak, Yanai Elazar, Sang Michael Xie, Shayne Longpre, Nathan Lambert, Xinyi Wang, Niklas Muenmighoff, Bairu Hou, Liangming Pan, Haewon Jeong, et al. A survey on data selection for language models. *arXiv preprint arXiv:2402.16827*, 2024. 1
- [3] David Alvarez-Melis and Nicolo Fusi. Geometric dataset distances via optimal transport. *Advances in Neural Information Processing Systems*, 33:21428–21439, 2020. 2, 3
- [4] Zalán Borsos, Mojmir Mutny, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. *Advances in neural information processing systems*, 33: 14879–14890, 2020. 2
- [5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Gi-an carlo Baldan, and Oscar Beijbom. nuscenes: A multi-modal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 6, 7
- [6] Liang-Chieh Chen. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 6
- [7] Sang Keun Choe, Hwijeon Ahn, Juhan Bae, Kewen Zhao, Minsoo Kang, Youngseog Chung, Adithya Pratapa, Willie Neiswanger, Emma Strubell, Teruko Mitamura, et al. What is your data worth to gpt? Ilm-scale data valuation with influence functions. *arXiv preprint arXiv:2405.13954*, 2024. 2
- [8] Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free dolly: Introducing the world’s first truly open instruction-tuned llm. *Company Blog of Databricks*, 2023. 8
- [9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 6
- [10] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013. 3
- [11] Junwei Deng, Ting-Wei Li, Shichang Zhang, and Jiaqi Ma. Efficient ensembles improve training data attribution. *arXiv preprint arXiv:2405.17293*, 2024. 2
- [12] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 8
- [13] Logan Engstrom, Axel Feldmann, and Aleksander Madry. Dsdm: Model-aware dataset selection with datamodels. *arXiv preprint arXiv:2401.12926*, 2024. 1, 2, 6
- [14] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R. Qi, Yin Zhou, Zoey Yang, Aurélien Chouard, Pei Sun, Jiquan Ngiam, Vijay Vasudevan, Alexander McCauley, Jonathon Shlens, and Dragomir Anguelov. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9710–9719, 2021. 6, 7
- [15] Lan Feng, Mohammadhossein Bahari, Kaouther Messaoud Ben Amor, Éloi Zablocki, Matthieu Cord, and Alexandre Alahi. Unitraj: A unified framework for scalable vehicle trajectory prediction. *arXiv preprint arXiv:2403.15098*, 2024. 1, 7, 8, 2
- [16] 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. 6, 7
- [17] Chengcheng Guo, Bo Zhao, and Yanbing Bai. Deepcore: A comprehensive library for coresnet selection in deep learning. In *International Conference on Database and Expert Systems Applications*, pages 181–195. Springer, 2022. 2
- [18] K. Tan et al. H. Caesar, J. Kabzan. Nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. In *CVPR ADP3 workshop*, 2021. 7
- [19] 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. 6
- [20] Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. Aligning ai with shared human values. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 8
- [21] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings*

- of the International Conference on Learning Representations (ICLR)*, 2021. 8
- [22] Yuzheng Hu, Pingbang Hu, Han Zhao, and Jiaqi W Ma. Most influential subset selection: Challenges, promises, and beyond. *arXiv preprint arXiv:2409.18153*, 2024. 1, 2, 3
- [23] Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Predicting predictions from training data. In *Proceedings of the 39th International Conference on Machine Learning*, 2022. 2
- [24] Hoang Anh Just, Feiyang Kang, Tianhao Wang, Yi Zeng, Myeongseob Ko, Ming Jin, and Ruoxi Jia. Lava: Data valuation without pre-specified learning algorithms. In *The Eleventh International Conference on Learning Representations*. OpenReview, 2023. 2, 4
- [25] Feiyang Kang, Hoang Anh Just, Anit Kumar Sahu, and Ruoxi Jia. Performance scaling via optimal transport: Enabling data selection from partially revealed sources. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 2
- [26] Leonid Kantorovich. On the translocation of masses. *Management science*, 5(1):1–4, 1958. 3
- [27] Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8110–8118, 2021. 2
- [28] Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. Openassistant conversations-democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36, 2024. 8
- [29] Zikang Liu, Kun Zhou, Wayne Xin Zhao, Dawei Gao, Yaliang Li, and Ji-Rong Wen. Less is more: Data value estimation for visual instruction tuning. *arXiv preprint arXiv:2403.09559*, 2024. 1
- [30] Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR, 2023. 8
- [31] Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *International Conference on Machine Learning*, pages 6950–6960. PMLR, 2020. 2
- [32] Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2980–2987. IEEE, 2023. 7
- [33] Maxime Oquab, Timothée Darcret, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1
- [34] Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale. In *International Conference on Machine Learning*, pages 27074–27113. PMLR, 2023. 2, 4, 5, 6, 1
- [35] Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33:19920–19930, 2020. 2, 3
- [36] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II* 14, pages 102–118. Springer, 2016. 6
- [37] Zhigang Sun, Zixu Wang, Lavdim Halilaj, and Juergen Luettin. Semanticformer: Holistic and semantic traffic scene representation for trajectory prediction using knowledge graphs. *arXiv preprint arXiv:2404.19379*, 2024. 7
- [38] Mirac Suzgun, Nathan Scales, Nathanael Schärlí, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, 2023. 8
- [39] Qwen Team. Qwen2.5: A party of foundation models, 2024. 8
- [40] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. 8
- [41] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemel Pontes, Deva Ramanan, Peter Carr, and James Hays. 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)*, 2021. 7
- [42] Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. In *Forty-first International Conference on Machine Learning*. 1, 2, 4, 6, 8
- [43] Xiaobo Xia, Jiale Liu, Jun Yu, Xu Shen, Bo Han, and Tongliang Liu. Moderate coresnet: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*, 2022. 2
- [44] Weiwei Xiao, Yongyong Chen, Qiben Shan, Yaowei Wang, and Jingyong Su. Feature distribution matching by optimal transport for effective and robust coresnet selection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(8): 9196–9204, 2024. 2
- [45] Zhen Yao, Xin Li, Bo Lang, and Mooi Choo Chuah. Goal-lbp: Goal-based local behavior guided trajectory prediction for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2023. 7
- [46] Huayi Zhang, Binwei Yan, Lei Cao, Samuel Madden, and Elke Rundensteiner. Metastore: Analyzing deep learning

- meta-data at scale. *Proceedings of the VLDB Endowment*, 17(6):1446–1459, 2024. [2](#)
- [47] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. *Advances in Neural Information Processing Systems*, 36, 2024. [1](#)

# TAROT: Targeted Data Selection via Optimal Transport

## Supplementary Material

### 8. More Implementation Details

#### 8.1. Influence Estimation

**ZCA Whitening.** As discussed in Sec. 5.1, we use ZCA whitening to achieve higher Local Data Scaling (LDS) scores due to its ability to preserve the data structure. However, our empirical analysis revealed that ZCA whitening has minimal impact on data selection performance. Therefore, for the data selection experiments, we opted for the more computationally efficient Cholesky whitening.

To apply ZCA whitening, we first compute the covariance matrix:

$$\Sigma = \frac{1}{N} \sum_{i=1}^N \tilde{\phi}(z)_i (\tilde{\phi}(z)_i)^\top, \quad (16)$$

where  $\tilde{\phi}(z)_i$  represents the centered input features. Next, we perform eigenvalue decomposition:

$$\Sigma = U \Lambda U^\top, \quad (17)$$

where  $U$  is the matrix of eigenvectors, and  $\Lambda$  is the diagonal matrix of eigenvalues. The ZCA whitening transformation is then defined as:

$$\phi(z)^{\text{ZCA}} = U \Lambda^{-\frac{1}{2}} U^\top \tilde{\phi}(z), \quad (18)$$

which ensures that the transformed features are uncorrelated and have unit variance.

The key difference between ZCA whitening and Cholesky-based whitening lies in the structure of the transformation matrix. In Cholesky whitening, the lower triangular matrix  $L$  introduces decorrelation sequentially, while ZCA whitening uses a symmetric transformation to minimize distortion of the original data structure. This preservation of the data's original geometric structure may result in a better LDS score. However, this structural similarity has a limited impact on data selection tasks, where the primary goal is to identify representative or diverse samples. Since both whitening methods achieve decorrelation and normalization, the relative distances and informativeness of the samples, which are key factors in data selection, remain largely unaffected by the choice of whitening method.

**Linear Data Modeling Score (LDS).** As mentioned in Sec. 5.1, we employ LDS as evaluation metric for influence estimation. LDS is computed by the Spearman rank correlation between the model outputs of models trained on randomly selected subsets and the average of attribution scores as predictions of those model outputs. For Cifar-10, we directly reuse the provided outputs of models trained on different subsets in the Github repository of TRAK [34], and

Table 3. Training Hyperparameters for Semantic Segmentation

Hyperparameter	Value
Batch Size	16
Learning Rate	$1 \times 10^{-4}$
Optimizer	SGD
Momentum	0.9
Weight Decay	$1 \times 10^{-4}$
Learning Rate Scheduler	WarmupPolyLR
Warmup Factor	1/3
Warmup Method	Linear

Table 4. Experiment Settings for Motion Prediction

Data Configurations	
History Trajectory Time Length	2.1 seconds
Future Trajectory Time Length	6 seconds
Object Type	Vehicle
Line Type	Center Lane

Model Configurations	
<b>Autobots</b>	
Batch Size	256
Learning Rate	0.00075
Optimizer	Adam
Gradient Clipping	5
<b>Wayformer</b>	
Batch Size	128
Learning Rate	$1 \times 10^{-4}$
Optimizer	AdamW
Gradient Clipping	5
Learning Rate Schedule	[10, 20, 30, 40, 50]

Table 5. Training Hyperparameters for Instruction Tuning

Hyperparameter	Value
Batch Size	128
Learning Rate	$2 \times 10^{-5}$
Optimizer	Adam
LoRA rank	128
LoRA $\alpha$	512
dropout	0.1
Warmup Epoch	4

utilize these outputs to calculate LDS. For nuScences, we randomly select 100 subsets of the candidate datasets. Each

subset is sampled to be 50% of the size of the candidate dataset. Then we independently train 100 models on these subsets and utilize their model outputs to calculate LDS.

#### Algorithm 1 Fixed-Size Selection

---

**Require:** • Candidate dataset  $\mathcal{D}_c = \{z_c^{(i)}\}_{i=1}^N$

- Target dataset  $\mathcal{D}_t = \{z_t^{(i)}\}_{i=1}^M$
- Desired subset size  $S = N$
- Distance metric  $d_{\mathcal{Z}}$

**Ensure:** Selected subset  $\mathcal{D}_s \subseteq \mathcal{D}_c$  of size  $S$

- 1: Initialize  $\mathcal{D}_s \leftarrow \emptyset$
- 2: Initialize iteration counter  $k \leftarrow 1$
- 3: **while**  $|\mathcal{D}_s| < S$  **do**
- 4:   Identify the  $k$ -th nearest candidates for each  $z_t \in \mathcal{D}_t$  based on  $d_{\mathcal{Z}}$
- 5:   Let  $\mathcal{D}_k$  be the set of these candidates
- 6:   **if**  $|\mathcal{D}_s| + |\mathcal{D}_k| \leq S$  **then**
- 7:      $\mathcal{D}_s \leftarrow \mathcal{D}_s \cup \mathcal{D}_k$
- 8:   **else**
- 9:     Compute potential  $\phi_i$  for each  $z_i \in \mathcal{D}_k$  using:
$$\phi_i = \min_{\pi \in \Pi(\mathcal{D}_s \cup \{z_i\}, \mathcal{D}_t)} \int_{\mathcal{D}_s \cup \{z_i\} \times \mathcal{D}_t} d_{\mathcal{Z}}(z, z_t) d\pi(z, z_t)$$
- 10:   Rank candidates in  $\mathcal{D}_k$  by  $\phi_i$  in ascending order
- 11:   Add  $z_i \in \mathcal{D}_k$  to  $\mathcal{D}_s$  if it is among the top  $S - |\mathcal{D}_s|$  candidates in  $\mathcal{D}_k$
- 12:   **Break** the loop
- 13:   **end if**
- 14:    $k \leftarrow k + 1$
- 15: **end while**
- 16: Compute potential  $\phi_i$  for each  $z_i \in \mathcal{D}_s$  to serve as weight  $w_i$  and scale to integers.
- 17: **return**  $\mathcal{D}_s, \{w_i\}$

---

## 8.2. Semantic Segmentation

Tab. 3 shows detailed hyperparameters for the semantic segmentation experiments.

## 8.3. Motion Prediction

We use the UniTraj [15] framework for motion prediction experiments. Detailed experiment settings are provided in Tab. 4.

## 8.4. Instruction Tuning

We follow the settings and hyperparameters in LESS [42]. Details are provided in Tab. 5

## 9. Details for Algorithms

We present the detailed data selection algorithm described in Sec. 4.3

---

#### Algorithm 2 OT-Distance Minimization Selection (OTM)

---

**Require:** • Candidate dataset  $\mathcal{D}_c = \{z_c^{(i)}\}_{i=1}^N$

- Target dataset  $\mathcal{D}_t = \{z_t^{(i)}\}_{i=1}^M$
- Number of folds  $K$
- Distance metric  $d_{\text{OT}}$

**Ensure:** Selected subset  $\mathcal{D}_s \subseteq \mathcal{D}_c$

- 1: Initialize  $\mathcal{D}_s \leftarrow \emptyset$
- 2: Split  $\mathcal{D}_t$  into  $K$  equal subsets:  $\mathcal{D}_t^{(1)}, \mathcal{D}_t^{(2)}, \dots, \mathcal{D}_t^{(K)}$
- 3: **for**  $fold = 1$  to  $K$  **do**
- 4:   Let  $\mathcal{D}_t^{(fold)}$  be the current fold
- 5:   Let  $\mathcal{D}_t^{(\text{tar})} \leftarrow \mathcal{D}_t \setminus \mathcal{D}_t^{(fold)}$
- 6:   Initialize temporary subset  $\mathcal{D}_s^{(fold)} \leftarrow \emptyset$
- 7:   Initialize iteration counter  $k \leftarrow 1$
- 8:   Initialize Distance  $d_{\text{pre}} \leftarrow \text{'inf'}$
- 9:   **while** True **do**
- 10:   Let  $\mathcal{D}_k$  be the set of  $k$ -th nearest candidates for each  $z_t \in \mathcal{D}_t^{(\text{tar})}$  based on  $d_{\mathcal{Z}}$
- 11:   **if**  $d_{\text{OT}}(\mathcal{D}_s^{(fold)} \cup \mathcal{D}_k, \mathcal{D}_t^{(fold)}) > d_{\text{pre}}$  **then**
- 12:     **Break** the loop
- 13:   **end if**
- 14:    $\mathcal{D}_s^{(fold)} \leftarrow \mathcal{D}_s^{(fold)} \cup \mathcal{D}_k$
- 15:    $d_{\text{pre}} \leftarrow d_{\text{OT}}(\mathcal{D}_s^{(fold)}, \mathcal{D}_t^{(fold)})$
- 16:    $k \leftarrow k + 1$
- 17: **end while**
- 18:    $\mathcal{D}_s \leftarrow \mathcal{D}_s \cup \mathcal{D}_s^{(fold)}$
- 19: **end for**
- 20: Compute potential  $\phi_i$  for each  $z_i \in \mathcal{D}_s$  to serve as weight  $w_i$  and scale to integers.
- 21: **return**  $\mathcal{D}_s, \{w_i\}$

---

## 10. More Experiment Results

### 10.1. Ablation Study on Whitened Feature Distance

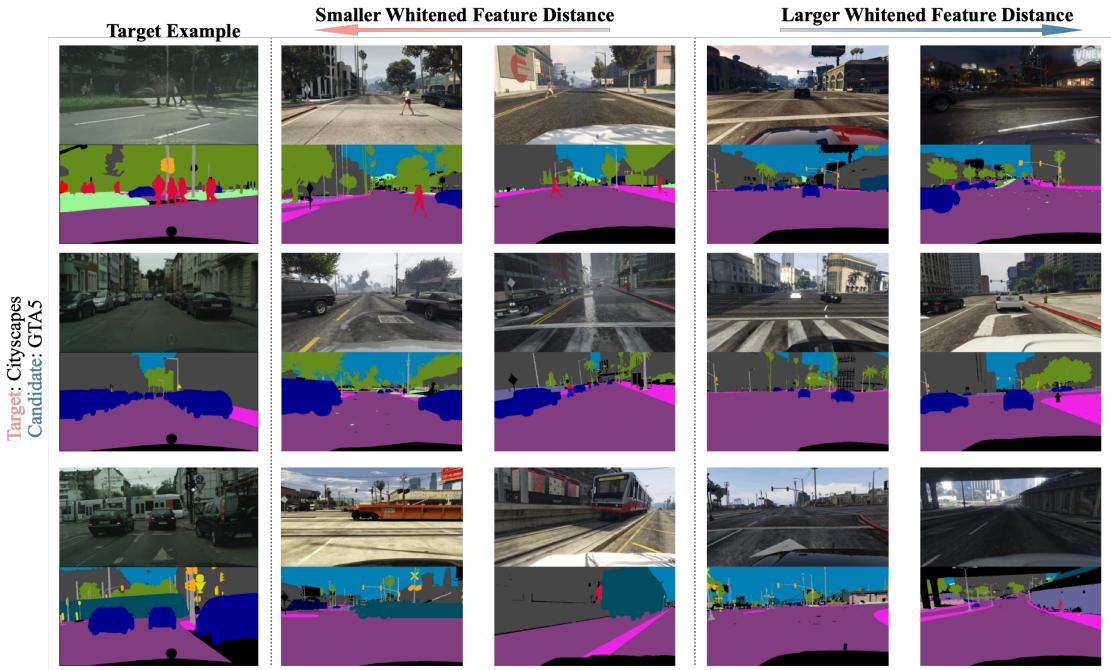
Table 6. Ablation Study of **WFD** on nuScenes with ensemble size ranging from 1 to 50. The evaluation metric is LDS.

	1	10	50
w/o whitened	0.007	0.010	0.017
w/o norm	0.024	0.060	0.097
w/o whitened+norm	0.005	0.008	0.015
<b>WFD</b>	<b>0.028</b>	<b>0.065</b>	<b>0.098</b>

In Tab. 6, we conduct ablation experiments by removing key components to explore their impacts on **WFD**. ‘w/o whitened’ means we do not perform whitening to the projected gradients and utilize the original projected gradients to calculate the distance. ‘w/o norm’ means we compute the L2 distance of the gradient vectors without normalization. Results demonstrate that removing the proposed com-



**Figure 8. More Qualitative Results of WFD in Motion Prediction.** Visualization of motion prediction tasks using the nuScenes and Waymo datasets. The dark-to-light blue lines indicate the past (input) and future (ground truth) trajectories for the target vehicle, while green lines represent surrounding vehicle trajectories.



**Figure 9. More Qualitative Results of WFD in Semantic Segmentation.** Semantic segmentation tasks with examples from the Cityscapes and GTA5 datasets. For each target example, the top image shows the RGB scene, and the bottom image depicts the corresponding ground truth semantic segmentation mask.

ponents of our whitened feature distance will lead to a performance drop, demonstrating the effectiveness of our in-

fluence estimation design.

## 10.2. More Qualitative Results of WFD

We provide additional qualitative results of **WFD** in Figs. 8 and 9. One clear observation is that the most supportive samples tend to closely resemble the target sample in both the input and output spaces. In motion prediction, the most negative samples also share similarities with the target sample in the input space (e.g., map and history trajectories); however, their ground truth trajectories often exhibit minor differences. For instance, in the first row of samples, the most negative examples demonstrate behaviors such as turning left or driving in the right-most lanes, whereas the target sample follows a trajectory in the lane adjacent to the right-most lane. These qualitative findings align with the definition of influence estimation: the most positive samples guide the model to make predictions similar to the target sample, while the negative samples encourage divergent predictions.