LLM-empowered Dynamic Prompt Routing for Vision-Language Models Tuning under Long-Tailed Distributions

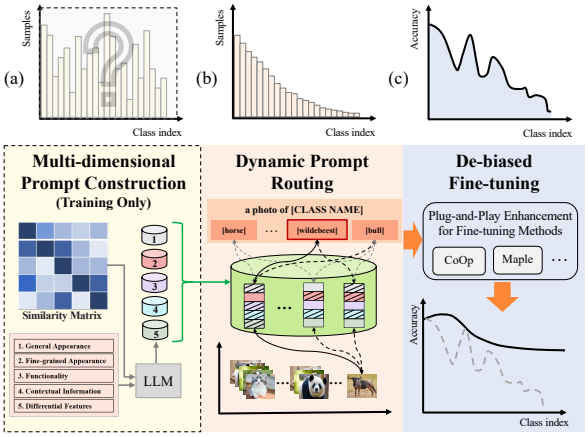
Anonymous ACL submission

Abstract

Pre-trained vision-language models (VLMs), such as CLIP, have demonstrated impressive capability in visual tasks, but their fine-tuning often suffers from bias in class-imbalanced scene. Recent works have introduced large language models (LLMs) to enhance VLM fine-tuning with supplementing semantic information. However, they often overlook inherent class imbalance in VLMs' pre-training, which may lead to bias accumulation in downstream tasks. To address this problem, this paper proposes a Multi-dimensional Dynamic Prompt Routing (MDPR) framework. MDPR constructs a comprehensive knowledge base for classes, spanning five visual-semantic dimensions. During fine-tuning, the dynamic routing mechanism aligns global visual classes, retrieves optimal prompts, and balances fine-grained semantics, yielding stable predictions through logits fusion. Extensive experiments on long-tailed benchmarks, including CIFAR-LT, ImageNet-LT, and Places-LT, demonstrate that MDPR achieves comparable results with current SOTA methods. Ablation studies further confirm the effectiveness of our semantic library for tail classes, and show that our dynamic routing incurs minimal computational overhead, making MDPR a flexible and efficient enhancement for VLM fine-tuning under data imbalance.

1 Introduction

Pretrained Vision-Language Models (VLMs), such as CLIP (Radford et al., 2021), have demonstrated remarkable capabilities in visual tasks by leveraging cross-modal knowledge with tuning techniques (Khattak et al., 2023; Zhou et al., 2022a). However, the fine-tuning of VLM under imbalanced downstream data exhibits significant bias (Wang et al., 2024), i.e., models favor many-sampled class optimization while under-performing on few-sampled classes, as shown in Figure 1(b) and (c). Lately,



(d) The framework of the proposed MDPR

Figure 1: Illustration of how MDPR alleviate the bias. To address the (a) unknown imbalance in the pre-training of VLMs and (b) the long-tailed distribution in downstream data, which jointly lead the (c) accuracy bias in fine-tuning, (d) MDPR constructs comprehensive knowledge using offline LLM generation, and designs a dynamic prompt routing mechanism to enhancing fine-tuning methods with de-biasing the predictions.

Large language Models (LLMs) are introduced to enhance VLM tuning, which faces two fundamental questions: (1) What semantic information from LLMs is effective in alleviating distributional bias? (2) How to leverage augmented information during the fine-tuning process?

To address VLMs' performance bottlenecks in data-scarce scenarios, prior works leverage LLMs for class-level semantic enhancement, sample synthesis, and open-world concept expansion. For semantic enhancement, LLMs generate discriminative prompts to improve inter-class separability (Zheng et al., 2024). For sample synthesis, LTGC (Zhao et al., 2024) guides diffusion models to synthesize tail-class samples. For concept expansion, PerVL (Cohen et al., 2022) and Custom Diffusion (Kumari et al., 2023) enable open-set generalization via text descriptions. However, these methods often overlook intrinsic VLMs' biases, leading to

cumulative bias during fine-tuning, and rely on static prompts or costly generative models, limiting adaptability. This necessitates an efficient framework for dynamic LLM-VLM interaction in fine-tuning.

To enhance the effectiveness of LLM knowledge in fine-tuning VLMs under imbalanced distributions, we propose Multi-dimensional Dynamic Prompt Routing (MDPR), as illustrated in Figure 1(d). Specifically, to address the implicit imbalance present during the VLM pre-training phase, MDPR firstly introduces a multi-dimensional prompt construction strategy. During training, it leverages zero-shot VLMs to extract and construct a prompt pool for each class, capturing five distinct dimensions: general appearance, fine-grained appearance, functionality, contextual information, and differential features. This multi-faceted prompt design helps mitigate prior biases for classes. Subsequently, in the dynamic prompt routing stage, it further alleviates the impact of imbalanced data by implementing global visual-class alignment, dynamic routing-based visual-prompt matching, and fine-grained semantic balancing. This process generates predictions from multiple perspectives, and robust results are achieved through a logit fusion mechanism. As an effective enhancement architecture, the proposed MDPR can be flexibly integrated with various VLM fine-tuning methods. The codes are available in https://anonymous.4open.science/r/MDPR-328C/README.md.

To evaluate the effectiveness of the proposed MDPR, we conducted extensive experiments on three long-tailed visual recognition benchmarks, namely CIFAR-100-LT, ImageNet-LT, and Places-LT. The experimental results demonstrate that MDPR, through the comprehensive prompt construction and dynamic routing mechanisms, effectively mitigates class imbalance biases in both pretrained models and downstream data, achieving robust performance improvements across head and tail classes while maintaining high compatibility with existing fine-tuning frameworks. The primary contributions of this work are:

- We propose a plug-and-play framework for VLM's fine-tuning, termed MDPR, which addresses the challenge of joint imbalance through dynamic prompt routing, achieving efficient performance enhancement.
- We propose a multi-dimensional prompt construction approach, which systematically enhances the semantic understanding of VLMs by integrating five semantic dimensions, significantly mitigating inherent biases in pretrained models.
- We validate the versatility of MDPR with different tuning methods, and it improves performance across three benchmarks with minimal additional parameters or time, particularly enhancing recognition of tail classes.

2 Related Works

2.1 Pretrained Model Fine-tuning under Long-tailed Distribution

Long-tailed data distributions challenge pretrained model fine-tuning, often leading to a bias towards head classes and impairing generalization to tail classes. Traditional strategies such as re-balancing (Shi et al., 2024; Tan et al., 2020; Cui et al., 2019), information augmentation (Xu et al., 2023; Li et al., 2024), and Mixture-of-Experts (MoE) models (Fedus et al., 2022; Zhang et al., 2023) offer foundational solutions. More recently, novel fine-tuning approaches for multimodal pretrained models have been explored. **Cross-modal collaborative fine-tuning** enhances minority class representations via visual-semantic contrastive learning and feature alignment (Chen et al., 2024). **Parameter-efficient fine-tuning (PEFT)** techniques, including adapter tuning (Kim et al., 2024) and prompt tuning (Dong et al., 2022), aim to adjust for minority classes with minimal backbone alteration, mitigating overfitting. Furthermore, **knowledge transfer and distillation** leverage priors from large pretrained models, employing teacher-student paradigms or cross-domain transfer to bolster tail class robustness (Rangwani et al., 2024).While these fine-tuning strategies address long-tailed distributions from various angles, many focus on re-weighting samples/losses or adapting model parameters. In contrast, MDPR introduces an explicit, structured semantic knowledge base and a dynamic routing mechanism, offering a complementary pathway to directly enhance the semantic understanding and discriminative capability for classes, especially those in the tail.

2.2 LLM-Enhanced Visual Representation Learning with Limited Samples

Large Language Models (LLMs) have enriched visual learning in data-scarce scenarios like few-shot and long-tailed recognition. Research primarily

explores three directions: **Category semantic enhancement.** For fine-grained or underspecified labels, LLaMP (Zheng et al., 2024) employs LLMs to generate descriptive prompts, improving inter-class separability. ArGue (Tian et al., 2024) integrates visual attributes and common sense semantics to guide prompt refinement. These methods typically yield a single, albeit enhanced, textual representation per class. MDPR, however, constructs a multi-dimensional prompt pool for each class, capturing diverse semantic facets, and dynamically selects from this pool based on image context, offering greater representational richness and adaptability. **Sample generation.** In imbalanced settings, LLMs produce detailed descriptions to steer text-to-image (T2I) models for synthesizing tail-class samples, as in LTGC (Zhao et al., 2024). While effective for data augmentation, such approaches often incur significant computational overhead from generative models and may not directly enhance the VLM's intrinsic understanding. MDPR, instead, focuses on efficiently enriching the VLM with pre-computed semantic knowledge, rather than relying on external sample generation. **Concept expansion.** LLMs facilitate modeling novel concepts in open-world settings. PerVL (Cohen et al., 2022) uses LLMs to generate personalized descriptions, extending VLM vocabularies. These methods primarily target open-set generalization or T2I generation. MDPR, while also leveraging LLM-derived knowledge, is specifically designed as a plug-and-play module to improve fine-tuning performance on closed-set, long-tailed recognition tasks by dynamically routing pre-defined, multi-faceted class semantics.

3 Method

3.1 Overview

To alleviate biases in the fine-tuning of VLMs, the proposed Multi-dimensional Dynamic Prompt Routing (MDPR) routes a visual-semantic knowledge base to enhance representation learning. Figure 2 shows the framework of MDPR, which comprises two synergistic modules: 1) The multi-dimensional prompt construction module generates a comprehensive knowledge base by designing five-dimensional prompts for LLM. 2) The dynamic prompt routing module enhances the utilization of the knowledge and provides de-biasing predictions. MDPR could serve as a flexible plug-and-play module, capable of seamless integration into existing fine-tuning methods.

3.2 Multi-dimensional Prompt Construction

To address potential inherent class biases in pre-trained VLMs, MDPR designs a class-specific prompt knowledge base spanning multiple semantic dimensions. The knowledge base endows VLMs with deeper understanding of classes, especially for distinguishing similar classes or prior tail classes in pre-training data.

3.2.1 Visual-Language Prompt Design

Since a single class name often fails to capture the complex visual attributes of certain class, especially rare or nuanced concepts in real-world data. Recent works (Tan et al., 2024; Zheng et al., 2024) have explored using prompts that incorporate knowledge from LLMs or word knowledge to enhance visual representation learning. Based on this line of research, we conduct a review of existing prompting strategies and make a five-dimensional classification of prompts.

Moreover, considering the prior bias inherent in VLMs, we introduce an additional prompt dimension termed differential features. Given a dataset with C classes, we first construct a confusion matrix $\mathbf{K}$ using CLIP's zero-shot predictions on the training set. For each class, the most frequently confused class is selected as the target for generating differential features, which may reflect biases in the model's pretraining distribution. To this end, the knowledge we includes following dimensions:
General Appearance (GA): Typical visual features of the class, such as color, shape, and size (Tian et al., 2022; Tan et al., 2024).
Fine-grained Appearance (FA): Focuses on more specific local details and textures crucial for distinguishing sub-class or similar objects (Zheng et al., 2024; Tan et al., 2024; Zhao et al., 2024).
Functionality (FT): Articulates the primary function, purpose, or role of the class object in specific activities (Tian et al., 2024).
Contextual Information (CI): Depicts the common background environments, associated objects, or typical scenarios where the class object is usually found (Tian et al., 2024; Zhao et al., 2024).
Differential Features (DF): Highlights unique characteristics by contrasting the class with one or more easily confusable similar classes.

For each class c, we obtain a set of prompts $\mathcal{P}_c = \{\mathrm{p}_{c,v}\}_{v=1}^{V_{dim}}$, covering V_{dim} prompts. The detailed prompt generation procedure and the choice of LLM are elaborated in Appendix B.

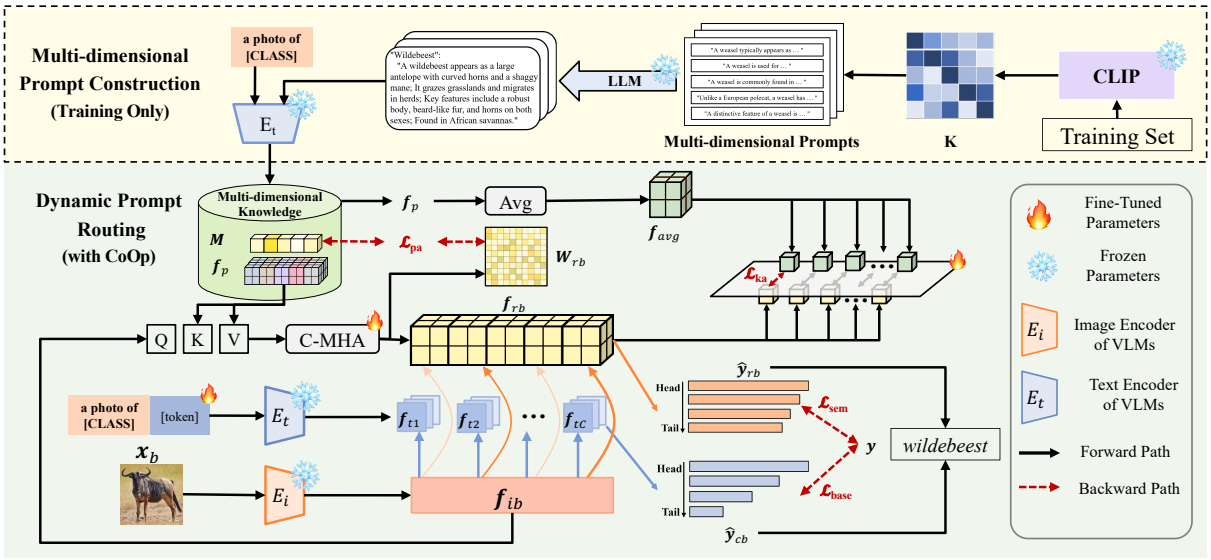


Figure 2: The framework of MDPR, which consists of two stages. The offline Multi-dimensional Prompt Construction builds a knowledge-base for enhancing semantics. The online Dynamic Prompt Routing aggregation the pre-learned knowledge for debias the predictions. Here we use CoOp (Zhou et al., 2022b) for an example.

3.2.2 Knowledge Base Construction

The generated prompts are diverse from classes to dimensions, we then send the prompts to VLMs for organizing a class-specific knowledge base. The text encoder $E_t(\cdot)$ of a frozen CLIP encodes $\mathrm{p}_{c,v}$ for v-th dimension prompt of class c into a d-dimensional feature $\mathbf{f}_p^{cv} = E_t(\mathrm{p}_{c,v})$, where d is the latent dimension of CLIP.

The encoded features combine as multi-dimensional prompt features $\mathbf{f}_p \in \mathbb{R}^{C\times V_{dim}\times d}$. To obtain a more general class-level semantic representation, the V_{dim} dimensional prompt features $\mathbf{f}_p^{cv}$ are averaged to $\mathbf{f}_{avg}^c \in \mathbb{R}^d$:

$$\mathbf{f}_{avg}^c = \frac{1}{V_{dim}} \sum_{v=1}^{V_{dim}} \mathbf{f}_p^{cv} \tag{1}$$

To introduce a beneficial inductive bias during dynamic routing and to guide the learning of attention weights, we construct a prior alignment matrix $\mathbf{M} \in \mathbb{R}^{C\times V_{dim}}$. An element $\mathbf{M}[c, v]$ represents the prior importance of the v-th dimensional prompt for class c. This importance is defined by the similarity between the encoded v-th dimensional prompt $\mathbf{f}_p^{c,v}$ and the encoding of a standard generic *prompt* for that class (e.g., "a photo of a [class name c]"):

$$\mathbf{M}[c, v] = \mathrm{Sim}(\mathbf{f}_p^{c,v}, E_t(prompt(c))) \tag{2}$$

3.3 Dynamic Prompt Routing

The Dynamic Prompt Routing (DPR) module designs relevant semantic information from the pre-constructed multi-dimensional prompts for each class, conditioned on the visual context of the input image. This process generates fine-grained, class-aware semantic representations for tail classes.

3.3.1 Image-attentive Semantic Extraction

The aim of this stage is to learn an instance-aware, dynamically adjusted semantic representation for visual inputs. In a long-tailed visual recognition dataset $D = \{X, Y\}$, for an image $x_b \in X$ labeled y_b, its d-dimensional visual features $\mathbf{f}_{ib} = E_i(x_b)$ are extracted by the image encoder $E_i(\cdot)$ of pre-trained CLIP.

The learned $\mathbf{f}_i^b$ underlines certain aspects of the semantics of class, a class-specific multi-head attention (C-MHA) module computes attention weights and forms attentive semantic features. For an image x_b in class c, the C-MHA outputs $\mathbf{W}_r^c$ for routing and attentive semantic features $\mathbf{f}_{rb}^c$, which can be formulated as:

$$\mathbf{f}_{rb}^c, \mathbf{W}_r^c = \text{C-MHA}(\mathbf{f}_{ib}, \mathbf{f}_p^c, , \mathbf{f}_p^c) \tag{3}$$

This C-MHA allows the model to dynamically focus on the most discriminative semantic aspects for each class conditioned on the image. In practice, we accelerate this procedure by a matrix manner, and collect the weights $\mathbf{W}_r$ and $\mathbf{f}_{rb}$ for image x_b across classes.

3.3.2 Semantic-enhanced Class Prediction

Leveraging the image features $\mathbf{f}_{ib}$ and the attentive semantic representations $\mathbf{f}_{rb}$, we formally compute semantic logits $\hat{\mathbf{y}}_{\mathrm{rb}}$ for classification. For an image

b and class c, the $\hat{\mathbf{y}}_{\text{rb}}$ is defined as:

$$\hat{\mathbf{y}}_{\text{rb}} = s \cdot \langle \mathbf{f}_{rb}, \mathbf{f}_{ib} \rangle \tag{4}$$

where s is a learnable temperature parameter. The logits $\hat{\mathbf{y}}_{\text{rb}}$ directly reflect the model's confidence in assigning an image to each class based on the dynamically aggregated semantic information.

Correspondingly, we introduce a dynamic semantic loss ($\mathcal{L}_{\text{sem}}$) to supervise this classification branch. This loss holds comparable importance to the base VLM's classification loss ($\mathcal{L}_{\text{ce}}$) and jointly drives the primary classification task:

$$\mathcal{L}_{\text{sem}} = \text{CLA}(\mathbf{z}_{\text{sem}}, \mathbf{y}) \tag{5}$$

where $\mathbf{y}$ represents the ground-truth labels, and Compensated Cross Entropy refers to loss function designed for imbalanced data (Shi et al., 2024).

To resist the bias accumulation under long-tailed scene, we introduce a regularization loss $\mathcal{L}_{\text{reg}} = \lambda_{\text{pa}}\mathcal{L}_{\text{pa}} + \lambda_{\text{ka}}\mathcal{L}_{\text{ka}}$, where λ_{pa} and λ_{ka} are weights of losses. The $\mathcal{L}_{\text{pa}}$ and $\mathcal{L}_{\text{ka}}$ target the attention routing strategy and the quality of the generated dynamic semantic representations, respectively:

Prior Alignment Loss for Routing Strategy Optimization ($\mathcal{L}_{\text{pa}}$): This loss aims to guide the learning of weights $\mathbf{W}_r$ using the prior alignment matrix $\mathbf{M}$, and $\mathbf{M}[c, v]$ signifies the prior importance of the v-th semantic dimension for class c.

It encourages the learned attention distribution to align with this desirable prior distribution, which may be more balanced or incorporate domain knowledge, thereby optimizing the information routing strategy:

$$\mathcal{L}_{\text{pa}} = \frac{1}{C}\sum_{c=1}^{C}\Big(1 - \text{Sim}(\mathbf{w}_{\text{r}}^{(c)}, \mathbf{M}[c, :])\Big) \tag{6}$$

This alignment helps the model to focus on semantically crucial dimensions, mitigating the influence of statistical biases in the data during dynamic routing.

Knowledge Alignment Loss for Representation Quality Enhancement ($\mathcal{L}_{\text{ka}}$): To improve the stability and generalization of the instance-aware dynamic semantic features $\mathbf{f}_{rb}$ via knowledge distillation. DPR encourages the distribution of the dynamic semantic features for the ground-truth class y_b (after a learnable linear projection $\text{Proj}(\cdot)$) to align with the distribution of the averaged semantic features for that class $\mathbf{f}_{avg}^{y_b}$ (i.e., $\mathbf{f}_{avg}^{y_b}[y_b, :]$, passed through the same projection layer) in a high-dimensional space:

$$\mathcal{L}_{\text{ka}} = \text{KL}\left(\text{Proj}(\mathbf{f}_{rb}, \text{Proj}(\mathbf{f}_{avg}^{y_b}[y_b, :]))\right) \tag{7}$$

where $\text{KL}(\cdot, \cdot)$ denotes the KL divergence.

3.4 Training Strategy

The training objective of the MDPR framework is to optimize the entire model end-to-end, enabling it to effectively leverage the multi-dimensional semantic knowledge base through dynamic routing for superior performance on imbalanced downstream visual tasks. The multi-task loss synergistically optimizes the representational capacity of the base VLM and the semantic enhancement and regularization mechanisms of the MDPR module.

3.4.1 Learnable Parameters

During the training process, the following parameters are subject to optimization:

Base VLM Framework Parameters: Depending on the chosen base VLM fine-tuning paradigm (e.g., CoOp or MaPLe), this may include its learnable prompt parameters (e.g., CoOp's context vectors $\mathbf{ctx}$, MaPLe's multi-level prompt parameters).

MDPR Module Parameters: This encompasses the parameters of the C-MHA, and the parameters of the learnable linear projection layer $\text{Proj}(\cdot)$.

The pre-computed multi-dimensional prompt features $\mathbf{f}_p$, class-level averaged semantic features $\mathbf{f}_{avg}$, and the prior alignment matrix $\mathbf{M}$ serve as fixed inputs during the training phase and are not updated.

3.4.2 Optimization

The MDPR model is optimized by minimizing the sum of losses. The total loss function $\mathcal{L}_{\text{total}}$ is formulated as:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{base}}\mathcal{L}_{\text{base}} + \lambda_{\text{sem}}\mathcal{L}_{\text{sem}} + \mathcal{L}_{\text{reg}} \tag{8}$$

where λ_{base} and λ_{sem} are the weights of losses, and λ_{base} is typically set to 1.0.

3.5 Logits-fused Inference

During inference, MDPR combines the predictions from the base VLM framework and the dynamic semantic routing pathway to yield final predictions. Given the class name-based logits $\hat{\mathbf{y}}_{cb}$ and the routing-based logits $\hat{\mathbf{y}}_{rb}$, the final fused logits $\hat{\mathbf{y}}_{fuse}$ are computed as:

$$\hat{\mathbf{y}}_{fuse} = (1 - \beta) \cdot \hat{\mathbf{y}}_{cb} + \beta \cdot \hat{\mathbf{y}}_{rb} \tag{9}$$

Dataset	#Class	IR	#Train	#Test
CIFAR-100-LT	100	10	19,573	10,000
		50	12,608	
		100	10,847	
ImageNet-LT	1,000	256	115,846	50,000
Places-LT	365	996	62,500	7,300

Table 1: Statistics of long-tailed datasets, where "#" means the number of item.

where $\beta \in [0, 1]$ is a hyperparameter balancing the two sources, typically set to 0.5 in our experiments (Section 4.2.2). This fusion allows MDPR to benefit from both the general representations of the base VLM and the instance-specific insights from the DPR module.

4 Experiments

To comprehensively evaluate the efficacy of MDPR, we conduct extensive experiments on three long-tailed image recognition benchmarks.

4.1 Datasets

Our experiments are conducted on three widely adopted long-tailed image recognition benchmarks: CIFAR-100-LT (Cao et al., 2019), ImageNet-LT (Liu et al., 2019), and Places-LT (Liu et al., 2019). Detailed statistics for these datasets are presented in Table 1.

4.2 Experimental Settings

4.2.1 Evaluation Metrics

Following the evaluation protocol proposed in (Liu et al., 2019), we report accuracies of all classes and three class subsets: Many-classes (>100 images), Medium-classes (20-100 images), and Few-classes (<20 images). This detailed breakdown allows for a more nuanced understanding of model behavior across varying class data densities.

4.2.2 Implementation Details

Base VLM Framework and Backbone: The MDPR is implemented and evaluated on top of two prominent prompt learning frameworks: CoOp (Zhou et al., 2022b) and MaPLe (Khattak et al., 2023). These are referred to as MDPR-CoOp/Ours(CoOp) and MDPR-MaPLe/Ours(MaPLe), respectively. All experiments except the CPRL (Yan et al., 2024) utilize the pre-trained CLIP ViT-B/16 model as the visual backbone.

Training Hyperparameters: All models are trained using the AdamW optimizer with a weight decay of 1×10^{-4}. The initial learning rate is set to 1×10^{-3}, decayed using a cosine annealing schedule over 20 epochs. A batch size of 128 is used for all datasets. The loss weights $\lambda_{\text{base}}, \lambda_{\text{sem}}, \lambda_{\text{upd}}, \lambda_{\text{kl}}$ in Equation (8) are determined through systematic tuning, with λ_{base} fixed at 1.0. The weights for $\mathcal{L}_{\text{sem}}$ and $\mathcal{L}_{\text{kl}}$ are linearly warmed up from 0 to their target values over the first 5 epochs. The logit fusion coefficient β (for combining $\mathbf{z}_{\text{base}}$ and $\mathbf{z}_{\text{sem}}$ during inference, see Equation A.1.5 is set to 0.5 by default. All experiments were conducted on a single NVIDIA RTX 3090 GPU. Further details on hyperparameter tuning ranges, final selected values, and the KL temperature T are provided in Appendix A.1.

4.3 Comparison Results

To comprehensively evaluate the MDPR framework's effectiveness in addressing long-tailed distributions, this section presents a comparative performance analysis against a range of representative methods on CIFAR-100-LT, ImageNet-LT, and Places-LT. The compared methods include Zero-Shot CLIP (ZS CLIP) (Radford et al., 2021), mainstream VLM prompt tuning approaches such as CoOp (Zhou et al., 2022b), CoCoOp (Zhou et al., 2022a), and MaPLe (Khattak et al., 2023), and recent VLM enhancement or long-tail recognition techniques (e.g., TextRefiner (Xie et al., 2025), CPRL (Yan et al., 2024), Candle(Shi et al., 2024)). All experiments were conducted under fair conditions. MDPR integrated with CoOp and MaPLe is denoted as **Ours (CoOp)** and **Ours (MaPLe)**. Detailed classification accuracies are in Tables 2 and 3

The proposed MDPR framework substantially enhances base VLM fine-tuning performance, achieving consistent and significant gains across all class groups and scenarios, particularly reaching SOTA levels for Few-shot classes on multiple benchmarks. For instance, on the challenging CIFAR-100-LT (IR=100), Ours (CoOp) and Ours (MaPLe) improve Few-shot accuracy from CoOp's 46.53% and MaPLe's 58.43% to 57.40% and 67.17%, respectively. On larger datasets like ImageNet-LT and Places-LT, MDPR also demonstrates strong efficacy; notably, Ours (MaPLe) boosts Few-shot accuracy on Places-LT by over 22 p.p. compared to MaPLe, securing top performance in Overall, Medium-shot, and Few-shot metrics on several datasets. These results robustly validate that MDPR, via structured multi-

Model	IR=10			IR=50				IR=100			
	All	Many	Med	All	Many	Med	Few	All	Many	Med	Few
CLIP-ViT-B/16 (ICML'21)	59.50	61.09	55.97	59.50	64.05	57.27	54.22	59.50	61.83	59.74	56.50
CoOp (IJCV'22)	70.88	75.06	61.58	65.70	79.63	58.44	50.50	64.34	79.43	64.51	46.53
CoCoOp (CVPR'22)	72.29	76.75	62.35	66.38	80.20	60.20	49.00	63.90	80.69	65.20	42.80
Maple (CVPR'23)	81.98	84.58	76.19	77.09	87.34	71.98	65.39	74.09	88.14	73.46	58.43
TextRefiner (AAAI'25)	74.22	78.12	65.55	67.70	81.83	62.51	47.33	64.32	83.00	66.03	40.53
CPRL (MM'24)	81.75	84.97	74.58	71.16	86.61	65.22	49.50	68.20	**88.74**	71.97	39.83
Candle (KDD'24)	75.77	76.71	73.68	73.14	77.15	70.17	70.78	72.42	76.14	72.54	**67.93**
Ours (CoOp)	76.25	78.51	71.23	72.44	81.76	67.93	61.50	70.33	81.17	70.57	57.40
Ours (Maple)	**84.73**	**86.32**	**81.19**	**81.38**	**88.68**	**76.90**	**74.94**	**79.25**	87.60	**81.26**	67.17

Table 2: Comparison results on CIFAR-100-LT dataset, where best results are **bolded** and suboptimal results are underlined. According to the split standard of dataset, CIFAR-100-LT with IR=10 contains no few-sampled classes.

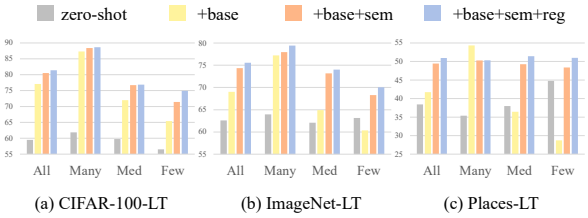
dimensional semantics and image-conditioned dynamic routing, effectively supplements VLMs with discriminative information, enhancing representation learning for balanced performance on long-tailed data.

MDPR demonstrates universality and effectiveness as an enhancement module across different base frameworks and datasets. While MaPLe inherently outperforms CoOp on some datasets, MDPR consistently delivers significant gains when combined with either framework. The substantial Few-shot improvement MDPR brings to MaPLe on Places-LT (over 22 p.p.) compared to that for CoOp (approx. 18 p.p.) suggests its particular effectiveness in unlocking the potential of advanced frameworks under extreme imbalance. Furthermore, unlike some specialized long-tail methods (e.g., Candle) that might excel on tail classes for specific datasets at the cost of head/medium class performance, MDPR promotes more balanced improvements.

MDPR's relative advantage tends to be more pronounced at higher imbalance ratios. Comparing results on CIFAR-100-LT across increasing IRs shows that while all methods' absolute performance declines, MDPR's (especially Ours (MaPLe)) improvement margin over baselines often widens. This further substantiates the crucial role of MDPR's multi-dimensional semantic understanding and dynamic routing in tackling extreme data imbalance.

4.4 Ablation Study

Our algorithm achieves performance gains through stacking multi-dimensional semantic prompts and regularization modules. To assess their contributions, we compared zero-shot CLIP, base MaPLe(**base**), MaPLe with semantic prompts (**base+Sem**), and full MDPR (**base+sem+reg**)



(a) CIFAR-100-LT (b) ImageNet-LT (c) Places-LT

Figure 3: Ablation results across datasets. Results of other IR of CIFAR-100-LT see in Appendix B.

on CIFAR-100-LT (IR=50), ImageNet-LT, and Places-LT. As shown in Figure 4, performance improves progressively with each module. Adding semantic prompts (**base+sem**) significantly boosts tail-class accuracy, e.g., Few on Places-LT from 28.73% to 48.39% (+19.66%). Regularization (**base+sem+reg**) further raises Few to 50.99% and slightly improves head and mid classes (e.g., Many to 79.42% on ImageNet-LT). Semantic prompts substantially mitigate tail-class bias via comprehensive semantic representations, while regularization enhances prediction consistency across head, mid, and tail classes by stabilizing dynamic routing.

4.5 In-depth Analysis of Knowledge-base Construction

An ablation study on the multi-dimensional semantic knowledge base using **Ours (CoOp)** on Places-LT (Table 4) reveals each dimension's distinct contribution. Removing the Differential Features (DF) dimension caused the largest overall accuracy drop, highlighting the critical role of distinguishing unique characteristics via comparison with similar classes. The removal of Contextual Information (CI) or General Appearance (GA) also significantly impacted performance, underscoring the importance of scene understanding and fundamental visual features.In contrast, lacking Fine-grained Appearance (FA) or Functionality (FT) had a smaller, yet noticeable, negative effect, confirm-

Model	ImageNet-LT				Places-LT			
	All	Many	Med	Few	All	Many	Med	Few
CLIP-ViT-B/16 (ICML'21)	62.95	63.96	62.08	63.15	38.40	35.49	37.97	44.77
CoOp (IJCV'22)	68.69	74.82	65.75	61.75	40.73	53.10	35.58	29.72
CoCoOp (CVPR'22)	-	-	-	-	41.12	52.95	35.84	31.39
Maple (CVPR'23)	69.02	77.20	64.90	60.37	41.37	**54.33**	36.45	28.73
TextRefiner (AAAI'25)	66.74	**81.75**	62.45	39.40	38.01	52.76	32.79	22.79
Candle (KDD'24)	71.28	76.38	69.55	62.91	45.81	46.97	45.42	44.56
Ours (CoOp)	74.57	77.67	73.42	69.87	48.89	49.45	48.72	47.96
Ours (Maple)	**75.57**	79.42	**74.02**	**70.04**	**50.94**	50.32	**51.42**	**50.99**

Table 3: Comparison results on ImageNet-LT dataset and Places-LT dataset, where best results are **bolded** and suboptimal results are underlined. The "-" in results means out of memory in our devices.

ing the supplementary value of specific visual details and object function information. Notably, all dimensions positively contributed to few-shot class recognition; removing any single dimension decreased few-shot accuracy by 1.3% to 2%, with CI removal having the most pronounced effect.

These findings demonstrate that a comprehensive, multi-dimensional knowledge base with complementary semantic dimensions is essential for MDPR to effectively address long-tailed distributions and enhance learning of data-scarce classes.

Knowledge	All	Many	Mid	Few
All	**48.89**	**49.45**	**48.72**	**48.24**
w/o GA	47.23	48.24	46.70	46.58
w/o FA	47.88	48.67	47.53	47.24
w/o FT	47.87	49.10	47.04	47.48
w/o CI	47.28	49.40	46.05	46.21
w/o DF	46.82	48.11	45.74	46.92

Table 4: Different knowledge base on Places-LT.

4.6 Comparative Analysis of Model Efficiency

To assess the practical applicability of our proposed MDPR framework, this section briefly analyzes the additional parameter count and its impact on training efficiency. As summarized in Table 5, our MDPR module introduces approximately 1.1M trainable parameters. This increment is substantially smaller than the total parameter count of the CLIP ViT-B/16 backbone (representing less than 0.74% of the backbone's parameters), positioning MDPR within the realm of parameter-efficient fine-tuning. For training on the ImageNet-LT dataset, integrating MDPR results in a slight increase in per-epoch training time of approximately 14 seconds for the CoOp baseline and 114 seconds for the MaPLe baseline.

In summary, while MDPR introduces a modest number of additional parameters and a slight increase in computation, these are well-justified by the significant performance gains, particularly in recognizing few-shot classes. The marginal overhead is especially low when MDPR is integrated with more complex frameworks like MaPLe, underscoring its practicality as an efficient enhancement module for VLMs addressing imbalanced data.

Metric	CoOp	CoOp +MDPR	MaPLe	MaPLe +MDPR
Param (M)	0.008	1.108	3.6	4.7
Time (s)	1115	1229	1361	1375

Table 5: Trainable Parameters (Param) and Training Time per Epoch (Time) on ImageNet-LT.

5 Conclusion

Addressing the class bias in fine-tuning vision-language models under long-tailed distributions, we propose the Multi-dimensional Dynamic Prompt Routing (MDPR) framework. Unlike traditional static prompt or high-cost sample generation methods, MDPR leverages a structured multi-dimensional semantic knowledge base and an image-driven dynamic routing mechanism to efficiently mitigate biases from pre-training and downstream data. First, MDPR constructs a five-dimensional prompt pool, providing comprehensive class understanding to counter prior biases. Second, an image-guided dynamic routing module, combined with regularization, generates instance-adaptive class representations by optimizing routing and representation stability. Experiments on CIFAR-100-LT, ImageNet-LT, and Places-LT demonstrate that MDPR significantly enhances tail-class performance while balancing head and medium-class robustness, achieving SOTA or highly competitive results. As a lightweight plug-and-play module, MDPR offers an effective paradigm for open-world long-tailed recognition.

Limitations

First, the effectiveness of MDPR has been primarily validated on the CLIP ViT-B/16 backbone integrated with CoOp and MaPLe. Its generalizability and performance on larger-scale or different VLM architectures require further examination in future work. Second, MDPR's prediction balancing, while benefiting from the rich multi-dimensional semantic library, still partially relies on known class distribution information from the training set. This dependency might limit its robustness in real-world scenarios with unknown or dynamic class distributions. Future research could explore integrating methods like causal inference to enhance adaptability to open environments. Third, the current multi-dimensional semantic knowledge base is constructed offline for predefined classes, potentially posing scalability challenges in incremental or open-set learning scenarios requiring rapid adaptation to new classes or domains. Future work could draw from continual learning principles to explore mechanisms for dynamic construction and updating of the semantic knowledge base.

References

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. 2019. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32.

J. Chen, J. Zhao, J. Gu, and 1 others. 2024. Multimodal framework for long-tailed recognition. *Applied Sciences*, 14(22):10572.

Niv Cohen, Rinon Gal, Eli A Meirom, Gal Chechik, and Yuval Atzmon. 2022. "this is my unicorn, fluffy": Personalizing frozen vision-language representations. In *European conference on computer vision*, pages 558–577. Springer.

Y. Cui, M. Jia, T. Y. Lin, and 1 others. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277.

B. Dong, P. Zhou, S. Yan, and 1 others. 2022. Lpt: Long-tailed prompt tuning for image classification. *arXiv preprint arXiv:2210.01033*.

W. Fedus, B. Zoph, and N. Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19113–19122.

J. Kim, D. Kim, H. Jung, and 1 others. 2024. Long-tailed recognition on binary networks by calibrating a pre-trained model. *arXiv preprint arXiv:2404.00285*.

Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. 2023. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1931–1941.

D. Li, J. Yan, T. Zhang, and 1 others. 2024. On the role of long-tail knowledge in retrieval augmented large language models. *arXiv preprint arXiv:2406.16367*.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. 2019. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2537–2546.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

H. Rangwani, P. Mondal, M. Mishra, and 1 others. 2024. Deit-lt: Distillation strikes back for vision transformer training on long-tailed datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23396–23406.

J. X. Shi, C. Zhang, T. Wei, and 1 others. 2024. Efficient and long-tailed generalization for pre-trained vision-language model. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2663–2673.

Hao Tan, Jun Li, Yizhuang Zhou, Jun Wan, Zhen Lei, and Xiangyu Zhang. 2024. Compound text-guided prompt tuning via image-adaptive cues. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5061–5069.

J. Tan, C. Wang, B. Li, and 1 others. 2020. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11662–11671.

Changyao Tian, Wenhai Wang, Xizhou Zhu, Jifeng Dai, and Yu Qiao. 2022. Vl-ltr: Learning class-wise visual-linguistic representation for long-tailed visual recognition. In *European conference on computer vision*, pages 73–91. Springer.

Xinyu Tian, Shu Zou, Zhaoyuan Yang, and Jing Zhang. 2024. Argue: Attribute-guided prompt tuning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28578–28587.

Y. Wang, Z. Yu, J. Wang, and 1 others. 2024. Exploring vision-language models for imbalanced learning. *International Journal of Computer Vision*, 132(1):224–237.

Jingjing Xie, Yuxin Zhang, Jun Peng, Zhaohong Huang, and Liujuan Cao. 2025. Textrefiner: Internal visual feature as efficient refiner for vision-language models prompt tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8718–8726.

Z. Xu, R. Liu, S. Yang, and 1 others. 2023. Learning imbalanced data with vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15793–15803.

Jiexuan Yan, Sheng Huang, NanKun Mu, Luwen Huangfu, and Bo Liu. 2024. Category-prompt refined feature learning for long-tailed multi-label image classification. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 2146–2155.

Y. Zhang, R. Wang, D. Z. Cheng, and 1 others. 2023. Empowering long-tail item recommendation through cross decoupling network (CDN). In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5608–5617.

Qihao Zhao, Yalun Dai, Hao Li, Wei Hu, Fan Zhang, and Jun Liu. 2024. Ltgc: Long-tail recognition via leveraging llms-driven generated content. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19510–19520.

Zhaoheng Zheng, Jingmin Wei, Xuefeng Hu, Haidong Zhu, and Ram Nevatia. 2024. Large language models are good prompt learners for low-shot image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28453–28462.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348.

A Appendix

A.1 Detailed Hyperparameter Settings

This section provides a comprehensive overview of the hyperparameter settings used for training our MDPR models and the baseline VLM frameworks, supplementing the details in Section 4.2.2 of the main paper. All experiments were conducted on a single NVIDIA RTX 3090 GPU.

A.1.1 Common Training Settings

The following settings were applied to all trained models (both baselines and our MDPR variants) unless specified otherwise:

- **Optimizer:** AdamW (Loshchilov and Hutter, 2017).
- **Weight Decay:** 1×10^{-4}.
- **Base Learning Rate (for prompts and MDPR modules):** 1×10^{-3}.
- **Learning Rate Schedule:** Cosine annealing schedule.
- **Total Training Epochs:** 20.
- **Batch Size:** 128 for all datasets.
- **Visual Backbone:** Pre-trained CLIP ViT-B/16 (Radford et al., 2021) for all experiments. The backbone parameters were kept frozen during fine-tuning of CoOp, MaPLe, and their MDPR-enhanced counterparts, consistent with standard prompt tuning practices.

A.1.2 Base VLM Framework Parameters

When MDPR is integrated, the parameters of the underlying base VLM frameworks were set as follows:

CoOp (Zhou et al., 2022b):

- **Number of Context Tokens (N_{ctx}):** 16.
- **Class Token Position:** "end".
- **Context Initialization:** Random initialization (standard for CoOp).

MaPLe (Khattak et al., 2023):

- **Number of Context Tokens (N_{ctx}):** 2 for both visual and language shallow prompts.
- **Deep Prompt Depth (Vision & Language):** 9 layers for both vision and language encoders.
- **Context Initialization:** Random initialization.

A.1.3 MDPR Module Parameters

The specific parameters for our MDPR module were configured as:

- **Semantic Prompt Embedding Dimension (d):** 512 (consistent with CLIP ViT-B/16 text encoder output).
- **Multi-Head Attention (MHA) in DPR:**
  - Number of Attention Heads: 8.
  - Dropout Rate (during training): 0.1.
- **KL Projection Layer (Proj$(\cdot)$):** This linear layer projects features from $d = 512$ to an intermediate dimension of 128 before KL divergence calculation.

A.1.4 Loss Weights and Temperatures

The weights for the individual loss components in the total loss function (Equation 8 in the main paper: $\mathcal{L}_{\text{total}} = \lambda_{\text{base}}\mathcal{L}_{\text{base}} + \lambda_{\text{sem}}\mathcal{L}_{\text{sem}} + \lambda_{\text{upd}}\mathcal{L}_{\text{upd}} + \lambda_{\text{kl}}\mathcal{L}_{\text{kl}}$) and the KL distillation temperature T were determined through systematic tuning on a validation split (e.g., a subset of the training data or a dedicated validation set if available).

- **λ_{base}:** Fixed at 1.0 for all experiments to give primary importance to the base VLM's objective.
- **Tuning Strategy for $\lambda_{\text{sem}}, \lambda_{\text{upd}}, \lambda_{\text{kl}}, T$:** A two-stage tuning process was generally followed:
  1. **Stage 1 (Tuning λ_{sem}):** λ_{upd} and λ_{kl} were initially set to 0. λ_{sem} was tuned by exploring values in the set $\{0.1, 0.5, 1.0, 2.0\}$. Preliminary experiments on CIFAR-100-LT (IR=100) suggested the following as strong starting points, which were then validated or slightly adjusted for other datasets:
     - For MDPR-CoOp: $\lambda_{\text{sem}} = 0.1$.
     - For MDPR-MaPLe: $\lambda_{\text{sem}} = 1.0$.
  2. **Stage 2 (Joint Tuning $\lambda_{\text{upd}}, \lambda_{\text{kl}}, T$):** With the selected λ_{sem}, λ_{upd} was tuned from the set $\{0.01, 0.05, 0.1, 0.5, 1.0\}$, λ_{kl} from $\{0.001, 0.005, 0.01, 0.05, 0.1\}$, and the KL distillation temperature T from $\{1.0, 2.0, 5.0\}$.
- **Loss Weight Warm-up:** The weights for $\mathcal{L}_{\text{sem}}$ and $\mathcal{L}_{\text{kl}}$ (i.e., λ_{sem} and λ_{kl}) were linearly warmed up from 0 to their target tuned values over the first 5 training epochs. This strategy was found to stabilize early training.

A.1.5 Inference Settings

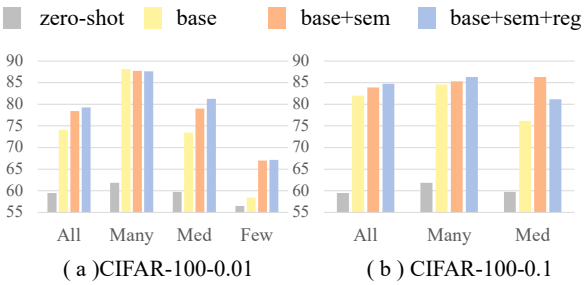
- **Logit Fusion Coefficient (β):** The coefficient β in the logit fusion equation ($\mathbf{z}_{\text{final}} = (1 - \beta)\mathbf{z}_{\text{base}} + \beta\mathbf{z}_{\text{sem}}$) was set to 0.5 by default for all reported results. This implies an equal contribution from the base VLM's predictions and the MDPR's dynamic semantic pathway predictions.

B Language Model Selection for Prompt Generation

visual features:
Provide a concise English phrase describing the key visual appearance features of a "{class-name}".
Focus on what it looks like (e.g., shape, color, texture, notable parts).
The phrase should be approximately {target-word-count} words and suitable to complete the sentence: "A {class-name} typically appears as {YOUR PHRASE HERE}."
Output ONLY the descriptive phrase. Do NOT include "A {class-name} typically appears as".
Descriptive phrase for "{class-name}":
functional-use: Provide a concise English phrase describing the primary function or purpose of a "{class-name}".
Focus on what it is used for.
The phrase should be approximately {target-word-count} words and suitable to complete the sentence: "A {class-name} is used for [YOUR PHRASE HERE]."
Output ONLY the descriptive phrase. Do NOT include "A {class-name} is used for".
Descriptive phrase for "{class-name}":
contextual-scene: Provide a concise English phrase describing the common environments or contexts where a "{class-name}" is typically found.
Focus on its usual surroundings or scenarios.
The phrase should be approximately {target-word-count} words and suitable to complete the sentence: "A {class-name} is commonly found in [YOUR PHRASE HERE]."
Output ONLY the descriptive phrase. Do

NOT include "A {class-name} is commonly found in".
Descriptive phrase for "{class-name}":
differential-comparison: Describe the key visual differences of a "{class-name}" when compared to a "{confusing-class-name}".
Focus on features that distinguish a "{class-name}" from a "{confusing-class-name}".
The description should be in English, concise, and approximately target-word-count words.
Output ONLY the descriptive phrase itself, suitable for completing the sentence: "Unlike a {confusing-class-name}, a {class-name} has [YOUR PHRASE HERE]."
Output ONLY the descriptive phrase of differences. Do NOT include "Unlike a {confusing-class-name}, a {class-name} has".
Descriptive phrase of differences for "class-name" compared to "confusing-class-name":
fine-grained-attribute: Provide a concise English phrase describing one or two highly distinctive or fine-grained visual attributes of a "{class-name}" that make it unique or easily identifiable.
Focus on specific, detailed characteristics.
The description should be in English, concise, and approximately target-word-count words.
Output ONLY the descriptive phrase itself, suitable for completing the sentence: "A distinctive feature of a {class-name} is [YOUR PHRASE HERE]."
Output ONLY the descriptive phrase of the attribute(s). Do NOT include "A distinctive feature of a {class-name} is".
Descriptive phrase of attribute(s) for "{class-name}":

For generating the multi-dimensional semantic prompts required by MDPR, we evaluated several Large Language Models (LLMs), including Qwen2.5, LLaMa4, and DeepSeek-V3. The evaluation primarily considered the statistical properties of the semantic similarities (forming the prior alignment matrix $\mathbf{M}$) between the CLIP-encoded LLM-generated prompts and generic class descriptions. Considering a comprehensive comparison of key metrics (summarized in Table 6) and a qualitative assessment of the generated text, we selected **Qwen2.5** for prompt generation due to its favorable overall performance in semantic alignment and distributional stability.



(a)CIFAR-100-0.01 (b) CIFAR-100-0.1

Figure 4: Ablation results across datasets.

Table 6: Key statistics for the prior alignment matrix $\mathbf{M}$ (semantic similarities) from prompts by different LLMs on CIFAR-100.

LLM	Mean	Std	Median
Qwen2.5	**0.8371**	**0.0370**	0.8452
LLaMa4	0.8360	0.0371	**0.8457**
DeepSeek-V3	0.8354	0.0373	0.8438

Algorithm 1 Multi-dimensional Dynamic Prompt Routing (MDPR)

Require: Training set $\mathcal{D} = \{(x_b, y_b)\}_{b=1}^{B}$
Ensure: Trained model ϕ
Initialize CLIP with pre-trained weights
Build confusion matrix $\mathbf{K} \xleftarrow{\text{CLIP}} \mathcal{D}$
Generate prompts $\mathcal{P}_c \xleftarrow{\text{LLM}} (\mathbf{K}, \text{prompts})$
Compute $\mathbf{M} \in \mathbb{R}^{C\times 5} \xleftarrow{\text{CLIP}} \mathcal{P}_c$
Encode $\mathbf{f}_p \in \mathbb{R}^{C\times 5\times d} \xleftarrow{\text{CLIP}} \mathcal{P}_c$
for x_b, y_b **do** in D, Compute $\mathbf{f_{ib}} = \phi_{\mathbf{v}}(\mathbf{x_b})$
    Calculate $\hat{\mathbf{y}}_{cb}$, constrained by $\mathcal{L}_{\text{base}}$
    Initialize $\mathbf{f}_{rb}$ and $\mathbf{W}_r$
    for class $c = 1$ to C **do**
        $\mathbf{f}_{rb}^c, \mathbf{W}_r^c = \text{C-MHA}(\mathbf{f}_{ib}^c, \mathbf{f}_p^c, \mathbf{f}_p^c)$
        Calculate $\mathcal{L}_{\text{reg}}$
        Append $\mathbf{f}_{rb}^c$ to $\mathbf{f}_{rb}$, $\mathbf{W}_r^c$ to $\mathbf{W}_r$
    end for
    Calculate $\hat{\mathbf{y}}_{rb}$, Constraint $\mathcal{L}_{\text{sem}}$
    Optimize $\mathcal{L}_{\text{total}}$
    Update $\phi \leftarrow \phi - \eta\nabla_\phi\mathcal{L}_{\text{total}}$
end for
return ϕ