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

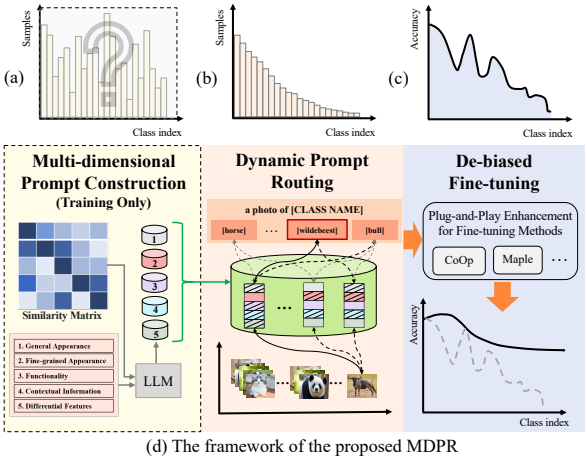
Yongju Jia[1] **Jiarui Ma**[1] **Xiangxian Li**[1,2†] **Baiqiao Zhang**[1,3] **Xianhui Cao**[4]
Juan Liu[1,2] **Yulong Bian**[1,2]
[1]Shandong University, Weihai, China
[2]Shandong Key Laboratory of Intelligent Electronic Packaging Testing and Application, Weihai, China
[3]The Hong Kong University of Science and Technology, Hong Kong, China
[4]AiLF Instruments, Weihai, China
jyjia@mail.sdu.edu.cn, jrma@mail.sdu.edu.cn, xiangxianli@sdu.edu.cn,
baiqiao.zhang@connect.ust.hk, hans@ailf.com.cn,
zzzliujuan@sdu.edu.cn, bianyulong@sdu.edu.cn

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 scenes. Recent works have introduced large language models (LLMs) to enhance VLM fine-tuning withsupplementaryy 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 multiple 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 operates with a slight increase in computational overhead, making MDPR a flexible and efficient enhancement for VLM fine-tuning under data imbalance. The codes are available in https://github.com/Sha843/MDPR.

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 and tuning (Khattak et al., 2023; Zhou et al., 2022a). However, 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). This challenge of learning from imbalanced data is not unique to VLM fine-tuning and has been extensively studied in various contexts. For instance, some works explore causal inference to disentangle robust features from spurious correlations (Meng et al., 2025), while others in federated learning tackle data heterogeneity across distributed clients through techniques like feature space alignment (Qi et al., 2025) and prototypical calibration (Qi et al., 2023; Meng et al., 2024).Lately, Large language Models (LLMs) are introduced to enhance VLM tuning, which faces two fundamental questions: (1) What



(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.

[†]Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 14809–14822
November 4-9, 2025

semantic information from LLMs is effective in alleviating distributional bias? (2) How to leverage augmented information during fine-tuning process?

To address VLMs' 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.

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 multiple 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.

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 pre-trained 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 multiple semantic dimensions, significantly mitigating inherent biases in pre-trained 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., 2024a), 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). Some works leverage text as privileged information during training to guide visual learning (Li et al., 2024b), while others explore causal inference to disentangle spurious correlations (Meng et al., 2025). **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). A related data-centric approach is adaptive data calibration, which rebalances the training data by down-sampling head concepts and synthesizing tail concepts (Song et al., 2025). 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 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 Motivation

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.

The core challenge in VLM fine-tuning arises from issues caused by imbalanced word distributions in pre-training and visual-language misalignment during fine-tuning.

In standard VLMs, the classification probability for a class c given an image x_b is formulated via a softmax over logits:

$$P(y=c|x_b) = \frac{\exp(\langle \mathbf{f}_{i,b}, \mathbf{f}_t^c \rangle)}{\sum_{j=1}^{C} \exp(\langle \mathbf{f}_{i,b}, \mathbf{f}_t^j \rangle)} \quad (1)$$

However, due to factors like long-tailed pre-training data, this probability is often implicitly biased by the pre-training frequency N_c^{pre} of class c. This can be modeled as an additive bias term in the logit space, leading to a biased probability:

$$P(y=c|x_b) \approx \frac{\exp(\langle \mathbf{f}_{i,b}, \mathbf{f}_t^c \rangle + \alpha \log N_c^{\text{pre}})}{\sum_{j=1}^{C} \exp(\langle \mathbf{f}_{i,b}, \mathbf{f}_t^j \rangle + \alpha \log N_j^{\text{pre}})} \quad (2)$$

where $\mathbf{f}_{i,b}$ and $\mathbf{f}_t^c$ are the normalized image and text features, respectively; C is the total number of classes; N_c^{pre} denotes the pre-training frequency of class c; and $\alpha > 0$ represents the bias strength. The additive term $\alpha \log N_c^{\text{pre}}$ systematically favors head classes over tail classes.

Figure 2 shows the framework of MDPR, which is designed to mitigate this bias from two fundamental principles. First, by constructing a **multi-dimensional semantic representation** for each class, we aim to dilute the uni-dimensional bias inherent in any single description (f_{text}^c). Second, by employing a **dynamic routing mechanism**, we make the final prediction dependent on the *relevance* between the image and the diverse semantic facets, rather than the static, frequency-based bias. The subsequent sections detail how we implement these principles. MDPR can serve as a plug-and-play framework, 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

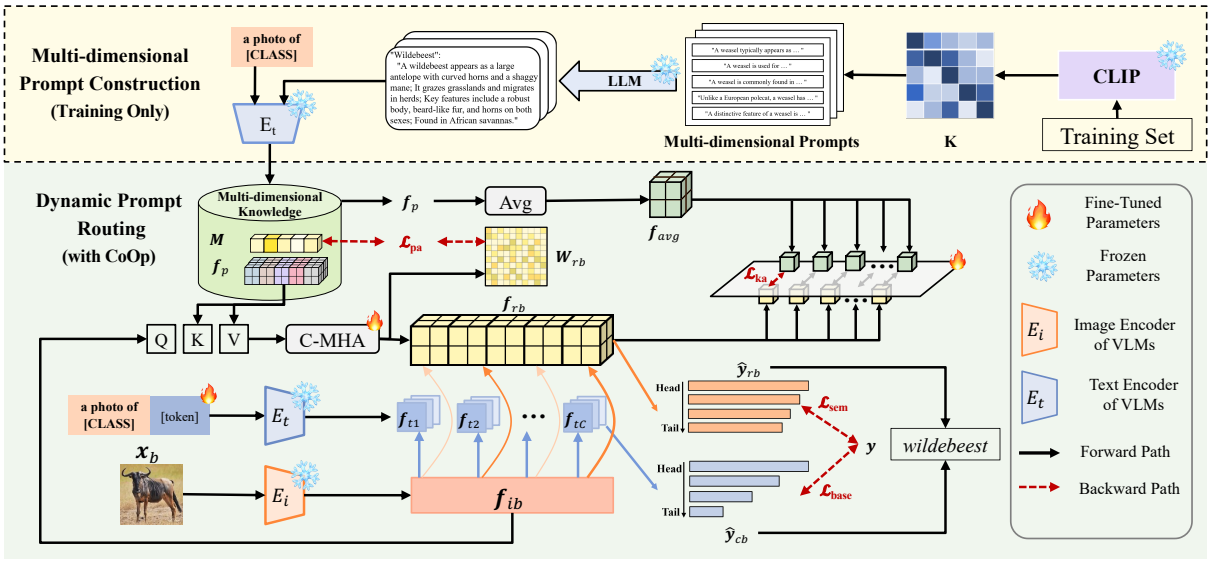


Figure 2: The framework of MDPR, which consists of two stages. The offline Multi-dimensional Prompt Construction builds a knowledge-base for enhancing semantics. The online Dynamic Prompt Routing aggregates the pre-learned knowledge to de-bias the predictions. Here we use CoOp (Zhou et al., 2022b) for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

VLMs with a 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 a class, especially for rare or nuanced concepts, we propose a structured, multi-dimensional prompt design. This design is inspired by recent VLM/LLM research on attribute-guided and context-aware prompting (Tian et al., 2024; Zheng et al., 2024; Tan et al., 2024), and also follows the cognitive process of human object recognition. The five dimensions are chosen to be complementary, constructing a comprehensive semantic space from three key perspectives: (1) **Semantic Granularity** (from coarse GA to fine-grained FA); (2) **Information Source** (from intrinsic properties like GA, FA, FT to extrinsic context CI); and (3) **Class Relationships** (from intra-class commonality to inter-class distinctions DF).

Moreover, considering the prior bias inherent in VLMs, we introduce the differential features dimension. 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. To this end, the knowledge base includes the following dimensions:

General Appearance (GA): Typical visual features of the class (e.g., color, shape) (Tian et al., 2022; Tan et al., 2024).

Fine-grained Appearance (FA): Specific local details and textures for distinguishing similar objects (Zheng et al., 2024; Tan et al., 2024; Zhao et al., 2024).

Functionality (FT): The primary function or purpose of the object (Tian et al., 2024).

Contextual Information (CI): Common background environments or associated objects (Tian et al., 2024; Zhao et al., 2024).

Differential Features (DF): Unique characteristics by contrasting with a confusable class.

This structured approach systematically provides the model with a more holistic understanding of class semantics, enhancing its discriminative capabilities. The core of our method is not the fixed number of dimensions, but a sufficiently rich prompt pool that enables the model to dynamically route the most relevant information. As our ablation study (Table 5) shows, each dimension provides indispensable, complementary information. Further details on the prompt generation process, including our structured query templates and verification-retry loop to mitigate LLM errors, are in Appendix B.

3.2.2 Knowledge Base Construction

The generated text prompts $\{\mathrm{p}_{c,v}\}$ are encoded into a knowledge base. The text encoder $E_t(\cdot)$ of a frozen CLIP model encodes each prompt $\mathrm{p}_{c,v}$ into a d-dimensional feature vector $\mathbf{f}_p^{c,v} = E_t(\mathrm{p}_{c,v})$.

These features form the multi-dimensional prompt tensor $\mathbf{F}_p \in \mathbb{R}^{C \times V_{\text{dim}} \times d}$. To obtain a general class-level semantic representation, we average

the features for each class c to get $\mathbf{f}_{\text{avg}}^c \in \mathbb{R}^d$:

$$\mathbf{f}_{\text{avg}}^c = \frac{1}{V_{\text{dim}}} \sum_{v=1}^{V_{\text{dim}}} \mathbf{f}_p^{c,v} \tag{3}$$

To introduce an inductive bias for routing, we construct a prior alignment matrix $\mathbf{M} \in \mathbb{R}^{C \times V_{\text{dim}}}$. An element $\mathbf{M}[c, v]$ represents the prior importance of the v-th prompt for class c, defined by its cosine similarity to a generic prompt (e.g., "a photo of a [class name c]"):

$$\mathbf{M}[c, v] = \text{Sim}(\mathbf{f}_p^{c,v}, E_t(prompt(c))) \tag{4}$$

3.3 Dynamic Prompt Routing

The Dynamic Prompt Routing (DPR) module dynamically selects and aggregates relevant semantic information from the knowledge base, conditioned on the input image's visual context.

3.3.1 Image-attentive Semantic Extraction

For an input image x_b with label y_b, its visual features $\mathbf{f}_{i,b} = E_i(x_b)$ are extracted using the image encoder $E_i(\cdot)$. A class-specific multi-head attention (C-MHA) module then computes attention weights $\mathbf{W}_r^{b,c}$ and forms an attentive semantic feature $\mathbf{f}_{rb}^{b,c}$ for each class c:

$$\mathbf{f}_{rb}^{b,c}, \mathbf{W}_r^{b,c} = \text{C-MHA}(\mathbf{f}_{i,b}, \mathbf{F}_p[c, :, :], \mathbf{F}_p[c, :, :]) \tag{5}$$

where $\mathbf{F}_p[c, :, :]$ are the V_{dim} prompt features for class c. This is performed for all classes in a matrix-wise manner for efficiency.

This relevance-based routing mechanistically contributes to de-biasing: it can increase the inter-class margin for tail classes by leveraging discriminative prompts (e.g., DF), and reduce the intra-class variance for head classes by adaptively selecting context-specific prompts (e.g., CI or FA).

3.3.2 Semantic-enhanced Class Prediction

The attentive semantic features are used to compute semantic logits $\hat{\mathbf{y}}_{rb}$. For an image b and class c, the logit is:

$$\hat{\mathbf{y}}_{rb}[b, c] = s \cdot \langle \mathbf{f}_{rb}^{b,c}, \mathbf{f}_{i,b} \rangle \tag{6}$$

where s is a learnable temperature. These logits are supervised by a dynamic semantic loss $\mathcal{L}_{\text{sem}}$ using Compensating Logit Adjusted Loss (CLA) (Shi et al., 2024) for imbalanced data:

$$\mathcal{L}_{\text{sem}} = \text{CLA}(\hat{\mathbf{y}}_{rb}, \mathbf{y}) \tag{7}$$

3.3.3 Regularization for Routing and Representation

To further stabilize the de-biasing process described above and enhance the quality of the learned representations, 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:

The **Prior Alignment Loss ($\mathcal{L}_{\text{pa}}$)** encourages the learned attention weights $\mathbf{W}_r$ to align with the prior importance matrix $\mathbf{M}$, preventing the model from overfitting to spurious correlations in the data. It is formulated as:

$$\mathcal{L}_{\text{pa}} = \frac{1}{B \cdot C} \sum_{b=1}^{B} \sum_{c=1}^{C} \Big(1 - \text{Sim}(\mathbf{W}_r^{b,c}, \mathbf{M}[c, :])\Big) \tag{8}$$

The **Knowledge Alignment Loss ($\mathcal{L}_{\text{ka}}$)** uses knowledge distillation to enhance the quality of tail-class representations. It aligns the dynamic semantic representation (student) with the globally-averaged semantic representation (teacher). Let $z_s = \text{Proj}(\mathbf{f}_{rb}^{b,y_b})$ and $z_t = \text{Proj}(\mathbf{f}_{\text{avg}}^{y_b})$, the loss is:

$$\mathcal{L}_{\text{ka}} = D_{\text{KL}}(\text{softmax}(z_s/T) \,||\, \text{softmax}(z_t/T)) \tag{9}$$

where y_b is the ground-truth class and T is the distillation temperature.

3.4 Training Strategy

The training objective of MDPR is to optimize the model end-to-end via a multi-task loss function. For clarity, the complete training procedure is summarized in Algorithm 1 in the Appendix.

3.4.1 Learnable Parameters

The learnable parameters include: (1) **Base VLM Framework Parameters** (e.g., CoOp's context vectors or MaPLe's multi-level prompts); and (2) **MDPR Module Parameters** (the C-MHA network and the projection layer $\text{Proj}(\cdot)$). The knowledge base ($\mathbf{F}_p, \mathbf{f}_{\text{avg}}, \mathbf{M}$) is fixed during training.

3.4.2 Optimization

The total loss function $\mathcal{L}_{\text{total}}$ is a weighted sum of the base VLM loss, the semantic loss, and the regularization losses:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{base}}\mathcal{L}_{\text{base}} + \lambda_{\text{sem}}\mathcal{L}_{\text{sem}} + \mathcal{L}_{\text{reg}} \tag{10}$$

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

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.

3.5 Logits-fused Inference

During inference, MDPR combines predictions from the base VLM pathway (logits $\hat{\mathbf{y}}_{cb}$) and the dynamic semantic pathway (logits $\hat{\mathbf{y}}_{rb}$). The final fused logits $\hat{\mathbf{y}}_{\text{fuse}}$ are computed as:

$$\hat{\mathbf{y}}_{\text{fuse}} = (1-\beta) \cdot \hat{\mathbf{y}}_{cb} + \beta \cdot \hat{\mathbf{y}}_{rb} \tag{11}$$

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, including our reproduced baselines, are trained using the AdamW optimizer with a weight decay of 1×10^{-4}. The initial learning rate for learnable prompts and MDPR-specific modules 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 $\lambda_{\text{base}}, \lambda_{\text{sem}}, \lambda_{\text{pa}}, \lambda_{\text{ka}}$ in Equation (10) are determined through systematic tuning, with λ_{base} fixed at 1.0. The weights for $\mathcal{L}_{\text{sem}}$ and $\mathcal{L}_{\text{ka}}$ are linearly warmed up from 0 to their target values over the first 5 epochs. The logit fusion coefficient β (for combining logits $\hat{\mathbf{y}}_{cb}$ and $\hat{\mathbf{y}}_{rb}$ during inference, see Equation (11) 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 the Zero-Shot CLIP (ZS CLIP) (Radford et al., 2021) baseline, a range of prompt tuning methods (CoOp (Zhou et al., 2022b), CoCoOp (Zhou et al., 2022a), MaPLe (Khattak et al., 2023), LASP (Bulat and Tzimiropoulos, 2023), PLOT (Chen et al., 2022)), the general VLM enhancement method TextRefiner (Xie et al., 2025), and techniques designed specifically for long-tail recognition such as CPRL (Yan et al., 2024) and 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,

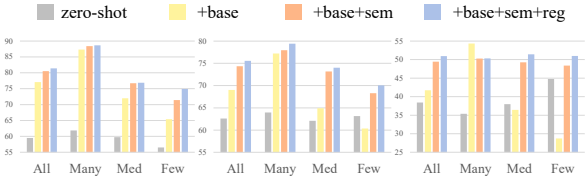
Model	IR=10			IR=50				IR=100			
	All	Many	Med	All	Many	Med	Few	All	Many	Med	Few
CLIP-ViT-B/16 (ICML'21)	59.50	61.09	55.97	59.50	64.05	57.27	54.22	59.50	61.83	59.74	56.50
CoOp (IJCV'22)	70.88	75.06	61.58	65.70	79.63	58.44	50.50	64.34	79.43	64.51	46.53
CoCoOp (CVPR'22)	72.29	76.75	62.35	66.38	80.20	60.20	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
PLOT++ (ICLR'23)	75.52	78.83	68.16	70.73	82.95	64.54	57.00	68.37	81.89	67.54	53.57
LASP (CVPR'23)	68.57	72.64	59.52	63.76	76.49	57.15	49.83	61.29	76.49	60.17	44.87
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.

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% 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-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%) compared to that for CoOp (approx. 18%) 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.



(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 C.1.

4.4 Generalization on Different Backbones

To validate MDPR's generalization beyond the ViT-B/16 backbone, we conducted additional experiments on Places-LT using CLIP-RN50 and CLIP-RN101. As shown in Table 4, the standard CoOp method struggles significantly on these ResNet backbones, with its performance dropping well below the zero-shot baseline for tail classes. In contrast, our MDPR (CoOp) consistently provides substantial improvements, not only recovering the performance but surpassing the zero-shot baseline across most metrics. This demonstrates MDPR's robustness and effectiveness as a backbone-agnostic enhancement. The significant gains, particularly in Few-shot categories (e.g., +26.45% on RN50), underscore MDPR's capability to effectively leverage semantic knowledge to guide representation learning, regardless of the underlying visual encoder architecture.

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.

Model	CLIP-RN50 Backbone				CLIP-RN101 Backbone			
	All	Many	Med	Few	All	Many	Med	Few
CLIP-ViT-B/16 (ICML'21)	33.37	32.88	32.29	36.75	33.37	30.42	33.12	39.39
CoOp (IJCV'22)	20.74	33.47	14.79	10.92	25.01	39.98	18.91	11.31
Ours (CoOp)	**34.98**	**31.50**	**35.70**	**37.37**	**37.85**	**35.53**	**38.66**	**40.25**

Table 4: Generalization results of MDPR (CoOp) on the Places-LT dataset with different ResNet backbones. Best results for each backbone are **bolded**. MDPR consistently provides significant gains over the standard CoOp baseline, which struggles on these backbones.

4.5 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**) on CIFAR-100-LT (IR=50), ImageNet-LT, and Places-LT. As shown in Figure 3, 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.

4.6 In-depth Analysis of Knowledge-base Construction

An ablation study on the multi-dimensional semantic knowledge base using **Ours (CoOp)** on Places-LT (Table 5) 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, confirming 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.

4.7 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 6, 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.

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 5: Different knowledge base on Places-LT.

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

Table 6: 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 multi-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

- Limited by the devices, the effectiveness of MDPR has been primarily validated on the CLIP ViT-B/16 backbone and further verified on ResNet backbones. Its generalizability and performance on larger-scale or different VLM architectures require further examination in future work.
- 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.
- The current multi-dimensional semantic knowledge base is constructed offline for predefined classes, raising challenges in both scalability and reliability. Regarding scalability, offline construction limits rapid adaptation in incremental or open-set learning scenarios. Regarding reliability, the quality of the knowledge base depends heavily on the external LLM. Future work could explore mechanisms for dynamic construction and updating of the knowledge base, while enhancing robustness by cross-validating across multiple LLMs or integrating real-world metadata.

Acknowledgments

This work was partially supported by the Project SDCX-ZG-202502017 funded by Postdoctoral Innovation Program of Shandong Province, the Ministry of Education Humanities and Social Sciences Research Project (Grant No. 22YJCZH007), and the Science and Technology Support Plan for Youth Innovation of Colleges and Universities of Shandong Province of China (Grant No. 2022KJN028).

References

Adrian Bulat and Georgios Tzimiropoulos. 2023. Lasp: Text-to-text optimization for language-aware soft prompting of vision & language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23232–23241.

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.

Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. 2022. Plot: Prompt learning with optimal transport for vision-language models. *arXiv preprint arXiv:2210.01253*.

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. 2024a. On the role of long-tail knowledge in retrieval augmented large language models. *arXiv preprint arXiv:2406.16367*.

Xiangxian Li, Yuze Zheng, Haokai Ma, Zhuang Qi, Xiangxu Meng, and Lei Meng. 2024b. Cross-modal learning using privileged information for long-tailed image classification. *Computational Visual Media*, 10(5):981–992.

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*.

Lei Meng, Xiangxian Li, Xiaoshuo Yan, Haokai Ma, Zhuang Qi, Wei Wu, and Xiangxu Meng. 2025. Causal inference over visual-semantic-aligned graph for image classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19449–19457.

Lei Meng, Zhuang Qi, Lei Wu, Xiaoyu Du, Zhaochuan Li, Lizhen Cui, and Xiangxu Meng. 2024. Improving global generalization and local personalization for federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36.

Zhuang Qi, Lei Meng, Zitan Chen, Han Hu, Hui Lin, and Xiangxu Meng. 2023. Cross-silo prototypical calibration for federated learning with non-iid data. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3099–3107.

Zhuang Qi, Lei Meng, Zhaochuan Li, Han Hu, and Xiangxu Meng. 2025. Cross-silo feature space alignment for federated learning on clients with imbalanced data. In *The 39th Annual AAAI Conference on Artificial Intelligence (AAAI-25)*, pages 19986–19994.

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.

Mingyang Song, Xiaoye Qu, Jiawei Zhou, and Yu Cheng. 2025. From head to tail: Towards balanced representation in large vision-language models through adaptive data calibration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9434–9444.

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, 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.

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.
- **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.

A.1.4 Loss Weights and Temperatures

The weights for the individual loss components in the total loss function (Equation 10) and the KL distillation temperature T were determined through systematic tuning on a validation split.

- λ_{base}**:** Fixed at 1.0 for all experiments.
- **Tuning Strategy:** A two-stage tuning process was generally followed:
  1. **Stage 1 (Tuning λ_{sem}):** With $\lambda_{\text{pa}} = 0$ and $\lambda_{\text{ka}} = 0$, λ_{sem} was tuned from $\{0.1, 0.5, 1.0, 2.0\}$.
  2. **Stage 2 (Joint Tuning $\lambda_{\text{pa}}, \lambda_{\text{ka}}, T$):** With the selected λ_{sem}, λ_{pa} was tuned from $\{0.01, 0.05, 0.1, 0.5, 1.0\}$, λ_{ka} from $\{0.001, 0.005, 0.01, 0.05, 0.1\}$, and the temperature T from $\{1.0, 2.0, 5.0\}$.
- **Typical Final Values:** While optimal values could slightly vary per dataset, the following settings demonstrated robust performance across most experiments:
  - For MDPR-CoOp: $\lambda_{\text{sem}} = 0.1, \lambda_{\text{pa}} = 0.05, \lambda_{\text{ka}} = 0.01, T = 2.0$.
  - For MDPR-MaPLe: $\lambda_{\text{sem}} = 1.0, \lambda_{\text{pa}} = 0.05, \lambda_{\text{ka}} = 0.005, T = 2.0$.
- **Loss Weight Warm-up:** The weights λ_{sem} and λ_{ka} were linearly warmed up from 0 to their target values over the first 5 epochs to stabilize early training.

A.2 Algorithm Pseudo-code

Algorithm 1 provides a detailed overview of the MDPR framework's training process, corresponding to the methodology described in Section 3.

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

B LLM-based Knowledge Base Construction

B.1 Prompt Templates and Generation Process

We used structured query templates to guide the LLM in generating the five semantic dimensions for each class. The templates provided to the LLM are shown below.

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}":

Differential Comparison via Confusion Matrix. To generate meaningful prompts for the **Differential Features (DF)** dimension, we first construct a confusion matrix $\mathbf{K}$ using CLIP's zero-shot predictions on the training set. For each class c, its most frequently confused class c' (where $c' \neq c$) is identified from $\mathbf{K}$. This confused class c' is then used to fill the '{confusing-class-name}' placeholder in the query template, guiding the LLM to generate highly relevant and discriminative comparisons.

B.2 LLM Selection, Error Handling, and Construction Cost

Beyond designing prompt templates, constructing a reliable knowledge base involves selecting a suitable LLM, handling potential generation errors, and assessing computational costs. This section details our approach to these practical considerations.

B.2.1 LLM Selection.

To ensure the quality of the generated knowledge base, 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 7 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.

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

Table 7: Key statistics for the prior alignment matrix $\mathbf{M}$ (semantic similarities) from prompts by different LLMs on CIFAR-100.

B.2.2 Handling Potential LLM Errors.

To mitigate potential errors or hallucinations from the LLM, we implemented a verification and regeneration mechanism. After an initial prompt is generated, we perform an automated check for format compliance and basic relevance. If the output is invalid (e.g., empty, or fails a simple keyword check), the system automatically triggers a secondary generation attempt for that specific prompt. This offline process ensures a higher quality and robustness of the final knowledge base.

B.2.3 Offline Construction Cost.

The construction of the multi-dimensional knowledge base is a one-time, offline process. We report its resource cost for transparency.

- **LLM Generation:** Using the Qwen2.5 API, generating five prompts per class takes approximately **19** seconds and uses **105** tokens on average. For large-scale datasets like ImageNet-LT (1000 classes), we employ parallel generation with 5 processes, reducing the total time to approximately **3900** seconds.
- **CLIP Encoding and Matrix Construction:** The memory footprint for encoding prompts and building the confusion matrix is comparable to standard CLIP zero-shot inference. With a batch size of 128 on a single NVIDIA RTX 3090, the peak memory usage is approximately **1.9** GB.

C Additional Experimental Results

C.1 Ablation Results for Different Imbalance Ratios

Figure 4 provides additional ablation study results on CIFAR-100-LT with different imbalance ratios (IR=10 and IR=100), supplementing Figure 3 in the main paper. The trend is consistent: performance improves progressively as each MDPR component (base+sem, base+sem+reg) is added.

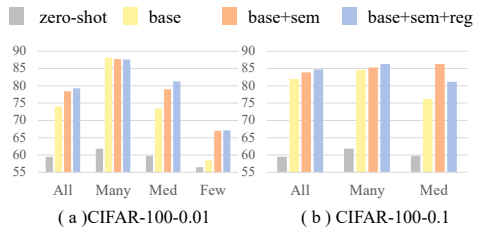


Figure 4: Ablation results on CIFAR-100-LT with IR=10 (labeled as -0.1) and IR=100 (labeled as -0.01). This supplements the main ablation figure.