KALE: Knowledge Aggregation for Label-free Model Enhancement

Yuebin Xu
yxu349@connect.hkust-gz.edu.cn
HKUST (Guangzhou)
Guangzhou, Guangdong, China

Xuemei Peng
xpeng558@connect.hkust-gz.edu.cn
HKUST (Guangzhou)
Guangzhou, Guangdong, China

Zhiyi Chen
zchen986@connect.hkust-gz.edu.cn
HKUST (Guangzhou)
Guangzhou, Guangdong, China

Zeyi Wen*
wenzeyi@hkust-gz.edu.cn
HKUST (Guangzhou)
Guangzhou, Guangdong, China

Abstract

Large foundation models have demonstrated remarkable success in natural language processing and computer vision. Applying the large models to downstream tasks often requires fine-tuning, in order to boost the predictive accuracy. However, the fine-tuning process relies heavily on labeled data and extensive training. This dependency makes fine-tuning impractical for niche applications, such as rare object detection or specialized medical tasks. To overcome these limitations, we propose KALE: Knowledge Aggregation for Label-free model Enhancement, a label-free method for model enhancement, leveraging knowledge aggregation via model fusion and adaptive representation alignment. Our method is powered by a carefully designed joint self-cooperative optimization function that considers (i) multi-granularity optimization (task-specific and layer-specific), (ii) self and cooperative supervision integration, and (iii) mitigation of error accumulation caused by entropy minimization. Additionally, we introduce a class cardinality-aware sample filtering to ensure the stability of the fusion process. We also design a lightweight representation alignment technique to refine the fusion coefficient in a few shots for quality enhancement. We evaluate our method on multiple image classification datasets using ViT-B/32 and ViT-L/14 backbones. Experimental results demonstrate that our label-free method consistently outperforms state-of-the-art unsupervised approaches, including TURTLE and supervised full fine-tuning, in terms of average performance. Specifically, compared to TURTLE, our method achieves average improvements of 20.7% with ViT-B/32 and 19.5% with ViT-L/14. Furthermore, on the challenging SUN397 dataset, our method surpasses supervised full fine-tuning by 4% and 2.3% with ViT-B/32 and ViT-L/14, respectively.

CCS Concepts

• **Computing methodologies → Learning paradigms**.

Keywords

Knowledge Aggregation, Unsupervised Learning, Model Fusion, Foundation Models

ACM Reference Format:
Yuebin Xu, Xuemei Peng, Zhiyi Chen, and Zeyi Wen. 2025. KALE: Knowledge Aggregation for Label-free Model Enhancement. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM '25), November 10–14, 2025, Seoul, Republic of Korea.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3746252.3761019

1 Introduction

Foundation models in computer vision (CV) and natural language processing (NLP) have seen rapid, transformative progress driven by large-scale datasets and growing computational devices. Prominent examples like BERT [5] in NLP and Vision Transformer (ViT) [6] in CV demonstrate remarkable generalization across many tasks, serving as foundational backbones for modern AI. Despite their versatility, adapting these models to downstream tasks typically relies on supervised fine-tuning, which demands extensive labeled data and substantial computational resources. This dependence creates a major barrier to practical deployment, especially in scenarios where high-quality annotations are scarce or prohibitively costly.

Fine-tuning is the dominant strategy for adapting foundation models to specific tasks, relying heavily on *learning from labeled data.* Full fine-tuning (FFT) updates all model parameters to fit task-specific distributions, typically yielding strong performance. Parameter-efficient fine-tuning (PEFT) methods like LoRA [13] reduce trainable parameters via lightweight modules, but still require labels to guide updates. Despite improved efficiency, both FFT and PEFT depend on annotations, limiting their use in domains with scarce labels—such as medical imaging [19], low-resource languages [32], or rare object detection [29]. Label dependence remains a major bottleneck, as high-quality annotation is often costly, or infeasible [39], restricting scalability and generalization.

In light of the challenges of data dependency in supervised fine-tuning, recent research has begun to explore the *learning-from-model* [43] paradigm. Existing model-based approaches, such as model fusion [16, 25], has emerged as an effective technique to

[*] Corresponding author.

CIKM'25, Seoul, Korea

ACM ISBN 979-8-4007-2040-6/2025/11
https://doi.org/10.1145/3746252.3761019

enable knowledge transferring. These methods aggregate knowledge from multiple pretrained models, enabling the construction of more powerful multi-task learning (MTL) models. Prior model fusion methods primarily focus on building universal MTL models [9] that perform reasonably well across tasks; however, they often suffer from significant performance degradation on individual tasks [9, 16, 26, 28, 40], limiting their practical applicability where task-specific accuracy is critical. Building upon these insights, our key contribution lies in enhancing the existing model fusion methods to not only achieve effective knowledge aggregation but also improve the performance of the fused model on single tasks.

Motivated by the advantages and limitations of model fusion, we propose KALE (Knowledge Aggregation for Label-free model Enhancement). On one hand, model fusion enables label-free knowledge aggregation by combining multiple pretrained models, offering a promising alternative to data-dependent fine-tuning. On the other hand, existing fusion methods often suffer from performance degradation on individual tasks, limiting their practical use where task-specific accuracy is critical. KALE is designed to harness the benefits of model fusion while addressing its shortcomings by explicitly optimizing the fusion process to maintain and improve task-specific performance. Additionally, KALE supports efficient downstream specialization without requiring labeled data or full parameter updates, effectively balancing multi-task knowledge integration with task-focused refinement.

KALE operates in two stages. In the first stage, KALE performs model fusion to aggregate diverse knowledge from multiple pretrained models. To ensure effective integration, we introduce a principled fusion optimization strategy called Joint Self-Cooperative Optimization, which captures knowledge at both task-specific and layer-specific granularities. This approach effectively mitigates the common issue of error accumulation found in entropy-based methods [23]. To further enhance the adaptation stability, we propose a class cardinality-aware sample filtering technique that selectively removes high-uncertainty samples according to class cardinality, ensuring robust and balanced performance across tasks. In the second stage, KALE specializes the fused model for a target task through adaptive representation alignment. This unsupervised adaptation enables the fused model to retain its aggregated knowledge while improving task-specific performance.

We conduct extensive experiments on diverse image classification benchmarks with varying numbers of classes using Vision Transformer models. We compare the unsupervised classification performance of our approach against a wide range of baselines built on ViT models. As illustrated in Figure (1), KALE consistently outperforms the state-of-the-art unsupervised method TURTLE in both of its variants, and even surpasses supervised full fine-tuning—all without relying on any labeled data. Our main contributions are summarized as follows:

- We propose KALE, a fully label-free method that improves model performance by combining knowledge aggregation through model fusion and adaptive representation alignment.
- We design a new unsupervised optimization function tailored to fusion optimization, incorporating both task-specific and layer-specific granularity, and introduce a class cardinality-aware filtering technique to stabilize and accelerate the fusion process.

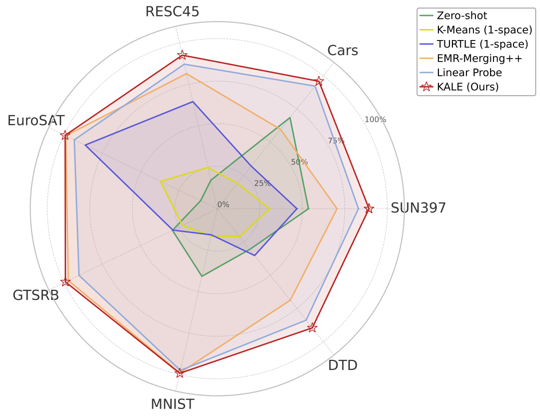


Figure 1: Comparison of unsupervised classification performance across multiple image classification tasks using CLIP ViT-B/32.

- We conduct extensive experiments on diverse image classification benchmarks. Results show that KALE outperforms full fine-tuning, the state-of-the-art unsupervised methods, demonstrating strong generalization and efficiency.

2 Related Work

2.1 (Weakly) Unsupervised Learning

Unsupervised and weakly supervised learning methods focus on extracting effective representations from data with limited or no annotations. TURTLE [8] enables state-of-the-art unsupervised transfer by optimizing margin-based classifiers in foundation model representations. Self-supervised learning (SSL) [14] learns generalizable features via pretext tasks, such as contrastive learning [35] that distinguishes similar and dissimilar samples, and masked autoencoding [11] that reconstructs masked inputs. Furthermore, deep clustering techniques [7] jointly optimize feature representations and cluster assignments to group data based on semantic similarity. The effectiveness of learned representations is typically assessed via linear probing [1], which trains simple classifiers with minimal supervision on downstream tasks. While these approaches avoid labeled data, they often underperform compared to supervised fine-tuning on individual tasks. Our work addresses this critical gap by leveraging model fusion for robust knowledge transfer and aggregation, enabling improved performance.

2.2 Label-Free Model Enhancement

Label-free methods aim to enhance model performance without relying on annotated data. Classic approaches include self-supervised learning [10], which learns representations via pretext tasks like contrastive or masked modeling, and unsupervised domain adaptation [24], which aligns distributions across domains using adversarial or discrepancy-based techniques. Pseudo-labeling [2] and consistency-based training [17] generate synthetic labels or enforce prediction stability to leverage unlabeled data. While effective, these methods often rely on carefully designed augmentations, or access to source data. More recent trends explore extracting knowledge

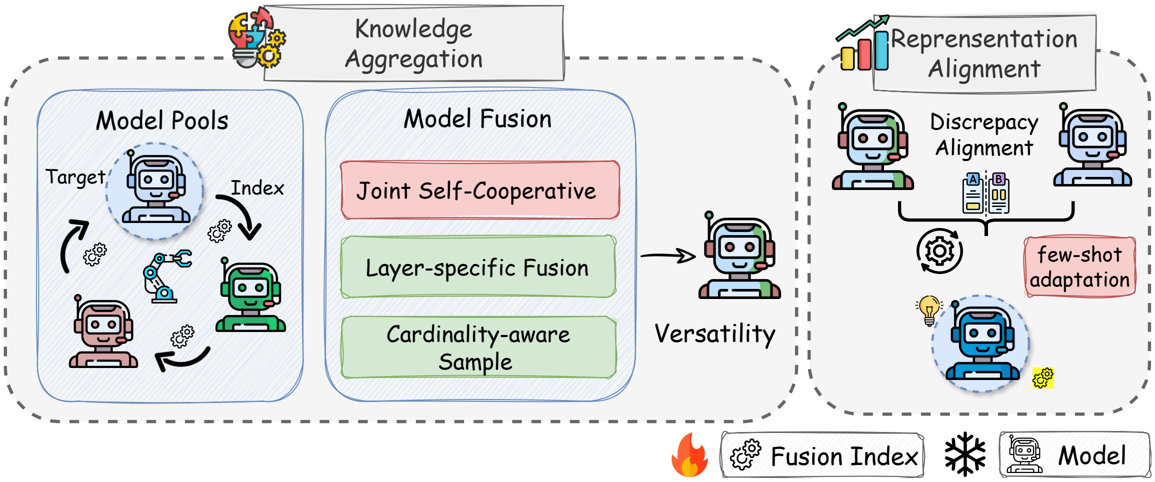


Figure 2: Overview of KALE. Two stages are included. (1) Knowledge aggregation via model fusion; (2) Adaptive representation alignment with few-shots adaptation.

directly from pretrained models [16, 28, 43], bypassing the need for additional training. In contrast, our method, KALE, adopts a model-centric view by fusing task-relevant knowledge from multiple models and adapting it to the target task via unsupervised representation alignment. This approach requires no labels, no source domain data, and no auxiliary pretraining, offering a scalable alternative for label-free model enhancement.

2.3 Model Fusion for Knowledge Aggregation

Model fusion is an emerging approach that aims to integrate multiple pretrained models into a single unified model [16, 26, 28, 40]. Techniques such as weight averaging [36], Fisher-weighted merging [28], task-specific editing [16], TIES-Merging [40], and AdaMerging [41] have been proposed to aggregate knowledge across models without requiring labeled data. EMR-Merging [15] represents the state-of-the-art fusion method, leveraging an elect-mask-rescale strategy to effectively resolve weight conflicts. While these methods show promise for multi-task learning, they often suffer from performance degradation on individual tasks due to interference among incompatible parameters. KALE builds on these advances by introducing a principled fusion optimization objective and a second-stage adaptation process to mitigate knowledge interference and boost task-specific accuracy.

3 Methodology

This section presents the overall framework of KALE, which consists of two key stages: (1) Knowledge aggregation via model fusion, and (2) Unsupervised representation alignment for performance enhancement. The entire process is designed to operate without any human-annotated labels, enabling truly label-free model enhancement.

3.1 Overview of KALE

KALE (Knowledge Aggregation for Label-free model Enhancement) is a two-stage framework designed to improve model performance without relying on any labeled data. Figure 2 shows the overview framework of KALE. It consists of the following components:

- **Stage 1: Knowledge Aggregation via Model Fusion.** KALE begins by aggregating knowledge from a set of fine-tuned models, all initialized from the same pretrained checkpoint. Instead of merging model weights directly, it identifies and combines the task-specific transformations (i.e., the changes made by each model relative to the shared initialization). These transformations are typically geometrically aligned in parameter space due to their shared training origin, which allows for a meaningful combination. Fusion coefficients are introduced to control the contribution of each source model, enabling the construction of an initial fused model that integrates diverse knowledge without any labeled supervision.
- **Stage 2: Adaptive Representation Alignment.** After initializing the fused model, KALE performs unsupervised adaptation using unlabeled target data. This stage aligns the internal representations of the fused model with those of a reference model, which could be a previous version of the fused model or another teacher model. The alignment is performed by minimizing a discrepancy measure (e.g., feature-level divergence), allowing the fused model to specialize toward the target domain. Importantly, this stage also refines the fusion coefficients introduced in the first stage, adapting the model's knowledge composition to better suit the target distribution.

Through this two-stage process, KALE enables effective knowledge transfer and adaptation in the absence of labeled data. It not only leverages the strengths of multiple fine-tuned models but also adapts to new domains or tasks via unsupervised representation alignment, making it a practical and efficient solution for zero-label generalization.

3.2 Stage 1: Knowledge Aggregation via Model Fusion

3.2.1 Fusion Objective. The first stage of KALE focuses on aggregating knowledge from multiple pretrained and fine-tuned models into a single unified model without relying on any labeled data.

Task Vector Representation. Given a set of N fine-tuned models $\{\theta_k\}_{k=1}^{N}$, where each θ_k denotes the parameters of the k-th model fine-tuned on a distinct task, and all share a common pretrained initialization θ_{pre}, we define the task-specific residuals (or *task vectors*) as:

$$\tau_k = \theta_k - \theta_{\text{pre}}, \quad k = 1, \ldots, N, \tag{1}$$

where τ_k encodes the learned adaptation from the pretrained model to task k. These task vectors reflect the semantic shifts introduced by each task and form the basis for constructing a fused model.

Fusion Formulation. The goal is to produce a fused model θ_M that incorporates complementary knowledge from all source tasks. To this end, we perform a weighted combination of task vectors:

$$\theta_M = \theta_{\text{pre}} + \sum_{k=1}^{N} \lambda_k \cdot \Phi(\tau_k), \tag{2}$$

where $\lambda_k \in \mathbb{R}$ is the fusion coefficient for task k, and $\Phi(\cdot)$ is a transformation function applied to task vectors (e.g., identity mapping, normalization, or attention-based reweighting). This formulation flexibly adjusts each task's influence in the fusion process.

Fusion Loss Objective. To balance the performance across all tasks, the fusion coefficients $\lambda = (\lambda_1, \ldots, \lambda_N)$ are optimized to minimize the average fusion loss over the tasks:

$$\lambda^* = \arg\min_{\lambda} \frac{1}{N} \sum_{k=1}^{N} \mathcal{L}_{\text{fuse}}^{(k)} \left(\theta_M(\lambda)\right), \tag{3}$$

where $\mathcal{L}_{\text{fuse}}^{(k)}$ denotes the unsupervised fusion loss measuring the fused model's performance on task k.

3.2.2 Limitations of Existing Methods. Existing model fusion strategies for multi-task learning can be broadly categorized into source-free and source-based approaches. (i) Source-free methods, such as Task Arithmetic [16] and TIES-Merging [40], linearly combine task vectors under the assumption of linear mode connectivity. While efficient and lightweight, these methods lack the flexibility to adapt to heterogeneous task behaviors, often resulting in performance degradation when tasks are incompatible or vary in scale. (ii) Source-based methods, including entropy minimization (EM)-based fusion [41], leverage unlabeled data and the output distributions of source models to guide optimization. Despite achieving better average performance, these methods are highly sensitive to noise in the early fusion stages. Randomly initialized fusion weights yield unreliable pseudo-labels, which, when iteratively reinforced, lead to *pseudo-label drift* and *error accumulation* [23]. This biased trajectory favors certain tasks while suppressing others, impairing generalization. A detailed analysis of error accumulation will be presented in the subsequent experimental results section. Furthermore, as the number of fused models increases, entropy becomes increasingly unreliable as a surrogate loss. Specifically, its correlation with the true optimization objective weakens, as evidenced by the decreasing Spearman's ρ between entropy and the final loss (Figure 3). This observation highlights the diminishing effectiveness of EM-based strategies in scalable multi-model settings.

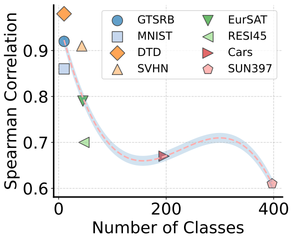


Figure 3: Spearman's ρ.

3.2.3 Joint Self-Cooperative Optimization. To address the aforementioned limitations, we propose a robust, label-free optimization strategy that jointly leverages **self-supervision** and **cooperative supervision** from task-specific teacher models. This dual-guided design mitigates the adverse effects of noisy early predictions and promotes balanced optimization across tasks.

Concretely, the fused model is trained to produce confident predictions via entropy minimization (self-supervision) while simultaneously aligning with the soft outputs of task-specific expert models (cooperative supervision). The self-supervised component reduces uncertainty and stabilizes early-stage optimization, whereas the cooperative signal provides external correction to mitigate misleading predictions and prevent convergence to biased task subsets. This joint objective effectively addresses pseudo-label drift and error accumulation, enhancing both the expressiveness and robustness of the fusion process and enabling more stable, generalizable performance across heterogeneous tasks.

Formally, given an unlabeled instance x_k from task k, and the corresponding output logits from the fused model θ_M and the k-th model $y_k(x_k)$ (used as soft targets), the task-specific optimization loss is defined as:

$$\mathcal{L}(x_k;\lambda_k) = \alpha \cdot \left(-\sum_{k=1}^{K} y_k(x_k) \cdot \log \theta_M(x_k;\lambda_k)\right) + H(\theta_M(x_k;\lambda_k)), \tag{4}$$

where the first term is a cross-entropy loss encouraging consistency with teacher model predictions, and the second term is an entropy regularization that drives the fused model toward confident predictions. This joint loss helps stabilize early-stage optimization, mitigates error accumulation, and reduces task-level bias.

Layer-Specific Fine-Grained Fusion. To further enhance compatibility across model components, we observe that different layers exhibit distinct behaviors across tasks. Shallow layers often encode general low-level features, while deeper layers learn task-specific semantics. A uniform fusion coefficient across all layers ignores this heterogeneity and may underperform.

To address this, we introduce *layer-specific fusion coefficients* λ_k^l, optimized individually per layer l and per model k. The resulting objective is:

$$\mathcal{L}(x_k;\lambda_k^l) = \alpha \cdot \left(-\sum_{k=1}^{K} \sum_{l=1}^{L} y_k^l \cdot \log \theta_M^l(x_k;\lambda_k^l)\right) + H(\theta_M^l(x_k;\lambda_k^l)), \tag{5}$$

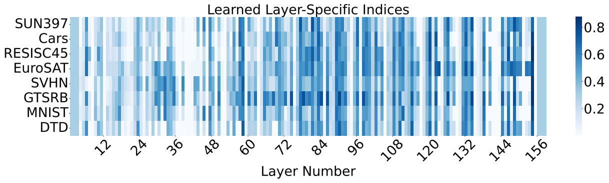


Figure 4: Learned layer-specific fusion indices across every pair of the eight models. Deeper colors represent a higher magnitude.

where θ_M^l is the output at layer l of the fused model. This fine-grained approach allows the fusion process to adapt flexibly to the representational diversity across layers.

Empirical Insights. Figure 4 illustrates the learned layer-specific indices across multiple models. The results show that shallow and middle layers tend to share similar fusion weights, while deeper layers display more divergent contributions, aligning with the intuition of hierarchical feature specialization. This confirms the effectiveness of our layer-specific optimization in aligning with model representation behavior. By integrating cooperative supervision, entropy regularization, and layer-wise fusion granularity, our joint optimization strategy enables robust knowledge aggregation across tasks, while mitigating common failure modes in existing fusion techniques.

3.2.4 *Class Cardinality-aware Sample Filtering.*

Despite the benefits of joint optimization, the quality of unlabeled samples remains critical. High-entropy predictions can degrade optimization, particularly for tasks with large class cardinality. To address this, we introduce a class-aware entropy-based filtering mechanism.

Entropy Thresholding. For each task k, we define a task-specific entropy threshold:

$$H_0(k) = \mu \cdot \log C_k, \tag{6}$$

where C_k is the number of classes and μ is a scaling factor. A sample x is retained only if:

$$F_{\text{ent}}(x; \theta_k) = \mathbb{I}\left[H(x; \theta_k) < H_0(k)\right], \tag{7}$$

where $\mathbb{I}[\cdot]$ is the indicator function.

Entropy-based Reweighting. To stabilize optimization when many samples are filtered, we apply a dynamic reweighting factor:

$$\gamma^{\theta_k} = \frac{1}{\exp\left[H(x; \theta_k) - H_0(k)\right]} \cdot \frac{B}{B - N_{F_{\text{ent}}}}, \tag{8}$$

where B is batch size and $N_{F_{\text{ent}}}$ is the number of retained samples. This ensures sufficient gradient contributions while preserving robustness.

Final Optimization Objective. The overall loss for optimizing the fusion coefficients integrates filtering and reweighting:

$$\min_{\lambda_k} \gamma^{\theta_k} \cdot F_{\text{ent}}(x; \theta_k) \cdot \mathcal{L}(x_k; \lambda_k). \tag{9}$$

This objective promotes reliable and token-efficient optimization, enabling robust knowledge fusion across tasks with diverse structures and label granularities.

3.3 Stage 2: Adaptive Representation Alignment

After obtaining the fused model from Stage 1, we further improve its task-specific generalization by aligning its intermediate representations with those of the target task model θ^T. This alignment step reduces representational mismatch and enhances cross-task transferability—without using labeled data.

Alignment Objective. Let $\mathcal{F}_T^l(\cdot)$ denote the representation at layer l of the target model θ^T, and let $\mathcal{F}_M^l(\cdot; \lambda)$ denote the corresponding representation of the fused model parameterized by fusion coefficients $\lambda = (\lambda_1, \ldots, \lambda_N)$. Given an unlabeled batch $X = \{x^{(i)}\}_{i=1}^B$ from the target task, we compute the Centered Kernel Alignment (CKA) similarity [20]:

$$\text{CKA}^{(l)} = \text{CKA}\left(\mathcal{F}_M^l(X; \lambda),\ \mathcal{F}_T^l(X)\right). \tag{10}$$

We define the alignment loss over the last two layers $\mathcal{S}$ as:

$$\mathcal{L}_{\text{align}} = \frac{1}{|\mathcal{S}|} \sum_{l \in \mathcal{S}} \left(1 - \text{CKA}^{(l)}\right). \tag{11}$$

Optimization. During this stage, the base models are frozen. We refine the fusion coefficients λ, which are embedded inside the fused model $\theta_M(\cdot; \lambda)$, by minimizing the alignment loss:

$$\lambda^* = \arg\min_{\lambda}\ \mathcal{L}_{\text{align}} + \eta \|\lambda\|_2^2, \tag{12}$$

where η is a regularization weight. This optimization can be completed using only a few forward passes over unlabeled examples, enabling an efficient few-shot adaptation.

Stage 2 performs lightweight representation alignment by adjusting internal fusion coefficients using only unlabeled data. By maximizing structural similarity to the target model's top-layer representations, the fused model achieves stronger capability on the singular task.

3.4 Overall Algorithm of KALE

Algorithm 1 outlines the proposed KALE framework, which enhances a pretrained model θ_{pre} by aggregating knowledge from N task-specific models $\{\theta_k\}_{k=1}^N$ using unlabeled data $\{\mathcal{D}_k\}$ and aligning with a target model θ^T. The method is guided by the entropy threshold μ, the loss weight α, and the regularization term η. In **Stage 1**, we compute task vectors $\tau_k = \theta_k - \theta_{\text{pre}}$ and initialize a fused model θ_M via weighted layer-wise fusion (Eqn. 2). Across iterations, we filter confident samples using entropy (Eqn. 7), reweight them via confidence scores (Eqn. 8), and update fusion coefficients $\{\lambda_k^l\}$ using a joint loss (Eqn. 4, 5). In **Stage 2**, we freeze all source models and adapt θ_M to the target task by aligning intermediate representations with θ^T using CKA similarity (Eqn. 10). The final coefficients λ^* are optimized to minimize alignment loss and ℓ_2 regularization, yielding the enhanced model $\theta_M(\lambda^*)$.

4 Experiment

To comprehensively evaluate both the effectiveness and efficiency of KALE in enhancing single-task model performance without the use of labeled data, we perform extensive experiments on a diverse range of image classification datasets. Additionally, we conduct thorough ablation studies to rigorously assess the contribution of each individual component within our method.

Algorithm 1: KALE

```
Input: $\theta_{\text{pre}}$, $\{\theta_k\}_{k=1}^N$, $\{\mathcal{D}_k\}_{k=1}^N$, $\mu$, $\alpha$, $\eta$
Output: Enhanced model $\theta_M(\lambda^*)$
// Stage 1
for $k = 1$ to $N$ do
    Extract task vector: $\tau_k \leftarrow \theta_k - \theta_{\text{pre}}$;
Initialize the fused model $\theta_M$ as defined in Eqn. 2;
for $i = 1$ to max epochs or convergence do
    for $k = 1$ to $N$ do
        Sample batch $x$ from $\mathcal{D}_k$;
        Filter and reweight samples based on Eqn. 7 and
          Eqn. 8;
    Update fusion coefficients $\{\lambda_k^l\}$ according to the joint
      loss in Eqn. 4 and Eqn. 5;
// Stage 2
Freeze $\theta_{\text{pre}}$, $\{\theta_k\}$, and $\theta^T$;
Sample a batch $X$ from the target task;
For selected layers $l \in \mathcal{S}$, compute fused and target
  representations;
Compute CKA similarity and alignment loss $\mathcal{L}_{\text{align}}$ as
  defined in Eqn. 10;
Update fusion coefficients $\lambda$ by minimizing $\mathcal{L}_{\text{align}} + \eta\|\lambda\|_2^2$;
return Enhanced model $\theta_M(\lambda^*)$
```

Tasks and Models. We evaluate KALE on seven diverse image classification datasets, each treated as a distinct target task: SUN397 [38], Cars [21], RESISC45 [3], EuroSAT [12], GTSRB [34], MNIST [22], and DTD [4]. These datasets span a wide range of visual domains, including natural scenes, remote sensing imagery, traffic signs, handwritten digits, textures, and fine-grained object categories. The number of classes ranges from 10 (e.g., MNIST) to 397 (SUN397). All experiments are based on vision encoders from the CLIP family [31]: ViT-B/32 and ViT-L/14, both pretrained on the Wikipedia-based image text dataset [33], and the finetuned models are obtained from the Task Vectors repository [16]. Accuracy is used as the evaluation metric for all tasks.

Baselines. Our baselines mainly follow prior work on unsupervised image classification, where TURTLE [8] represents the state-of-the-art. In addition to TURTLE, we include K-Means clustering [27] to provide a classical unsupervised baseline. To cover weakly-supervised approaches, we add Linear Probe [1], which trains a logistic regression classifier on frozen features with L2 regularization. For upper bounds, Full fine-tuning [37], training from scratch with full labels and multitask learning [15] jointly training on source and target tasks serve as our baselines. Furthermore, we consider label-free model aggregation baselines, including EMR-Merging [15], currently the SOTA in this category, alongside weight averaging [36] and Task arithmetic [16]. Zero-shot CLIP [31] is included as a vision-language pretrained baseline evaluated directly without adaptation. All methods are evaluated on single-task enhancement scenarios.

Implementation Details. KALE employs the standard Task Arithmetic method for weight transformation, as Eqn. 2, performing a linear combination with a fusion coefficient $\lambda_k = 0.3$. Fusion is applied exclusively to the Vision Transformer encoders, while classification heads are excluded due to the differing number of classes across tasks. Linear Probe uses CuML logistic regression with L2 regularization tuning; K-Means with 1000 iterations and 10 initializations, and TURTLE trained for 6000 steps with batch size 10,000 and entropy regularization $\gamma = 10$. TURTLE 2-space integrates DINOv2 features. Evaluation is via Hungarian-matched clustering accuracy on test sets.

4.1 Results

Performance Comparison. We present the overall performance in Table 1, where the highest values are highlighted in bold and the second-best results are underlined. As shown, our method KALE consistently achieves the best average accuracy across all benchmarks for both ViT-B/32 and ViT-L/14, reaching 90.2% and 94.0%, respectively. This outperforms all other approaches, including strong supervised baselines. Specifically, KALE surpasses the best supervised method (Full Fine-Tuning) by +0.7% on ViT-B/32 and +0.4% on ViT-L/14—remarkably, without using any labeled data.

Compared to recent unsupervised methods like TURTLE and K-Means, our approach shows substantial improvements, exceeding them by more than 20 percentage points on average. It also clearly outperforms weakly supervised baselines such as Linear Probe by +3.5% (ViT-B/32) and +2.7% (ViT-L/14). Furthermore, when compared to the strongest label-free model aggregation baseline, EMR-Merging++, KALE achieves a notable gain of +2.6% and +6.4% on ViT-B/32 and ViT-L/14, respectively. A minor exception is observed on the MNIST dataset, where our method slightly underperforms the best baseline by 0.2% (ViT-B/32) and 0.1% (ViT-L/14). We attribute this to MNIST's low visual complexity and narrow domain, which tend to benefit conventional fine-tuning or memorization-heavy strategies.

The superiority of KALE can be attributed to three key factors: (i) *Knowledge integration across multiple models.* By aggregating pre-trained models with related but diverse knowledge, KALE effectively captures shared capabilities—such as generalizable feature extraction—in tasks like image classification, yielding stronger generalization. (ii) *Conflict-aware and fine-grained model fusion.* Unlike previous aggregation-based methods that often suffer from destructive task interference, our approach employs a layer-specific fusion strategy that mitigates such conflicts. The notable performance margin over EMR-Merging++ and other merging baselines highlights the stability and effectiveness of this design. (iii) *Adaptive representation alignment for performance recovery.* Our second-stage feature-level alignment addresses distribution mismatches introduced during fusion, restoring task-specific model capacity and improving downstream performance without label supervision.

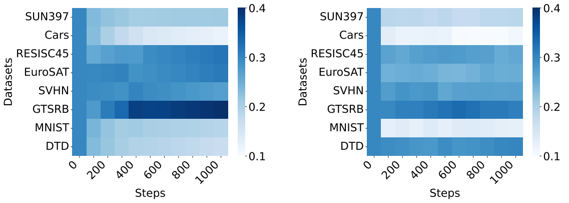
These advantages hold consistently across both model scales, demonstrating the scalability of KALE. Traditional clustering approaches and linear probes, although computationally efficient, fail to leverage the rich cross-model knowledge, falling short by over 20 percentage points on average. Similarly, zero-shot CLIP, despite its large-scale pretraining, underperforms compared to our method, reinforcing the necessity of explicit downstream adaptation.

Table 1: Performance comparison on image classification benchmarks using ViT-B/32 and ViT-L/14. The highest values are highlighted in bold, and the second-best results are underlined.

	Method	SUN397	Cars	RESC45	EuroSAT	GTSRB	MNIST	DTD	Avg.
ViT-B/32	Zero-shot CLIP [31]	63.2	59.6	60.2	45.0	32.6	48.3	44.4	50.5
	Multi-task Learning [15]	73.9	74.4	93.9	98.2	**98.9**	99.5	77.9	88.1
	Full Fine-Tuning [37]	75.3	77.7	96.1	99.7	98.7	**99.7**	79.4	89.5
	Linear Probe [1]	76.1	**80.9**	92.5	95.2	86.7	98.9	76.5	86.7
	K-Means (1-space) [27]	50.4	43.7	66.0	63.3	32.7	57.5	43.7	51.0
	K-Means (2-space) [27]	57.5	50.1	65.3	64.8	23.4	46.9	50.1	51.2
	TURTLE (1-space) [8]	58.1	49.1	82.2	69.8	39.6	80.9	49.1	61.3
	TURTLE (2-space) [8]	65.2	46.0	88.0	95.6	38.3	97.2	56.0	69.5
	Task Arithmetic [16, 18]	63.8	62.1	72.0	77.6	65.1	94.0	52.2	69.5
	Weight Averaging [36]	65.3	63.4	71.4	71.7	52.8	87.5	50.1	66.0
	EMR-Merging++ [15]	75.2	72.8	93.5	99.5	98.1	99.6	74.4	87.6
	KALE (Ours)	**79.3**	78.4	**96.1**	**99.8**	98.8	99.5	**79.6**	**90.2**
ViT-L/14	Zero-shot CLIP [31]	66.8	77.7	71.0	59.9	50.5	76.3	55.3	65.7
	Multi-task Learning [15]	80.8	90.6	96.3	96.3	99.1	99.6	84.4	92.4
	Full Fine-Tuning [37]	82.3	92.4	**97.4**	**100.0**	99.2	**99.7**	84.1	93.6
	Linear Probe [1]	81.5	90.7	95.5	97.0	93.2	99.0	81.9	91.3
	K-Means (1-space) [27]	55.4	50.5	73.7	71.7	45.9	66.6	50.5	59.2
	K-Means (2-space) [27]	60.6	49.0	73.9	84.7	31.3	69.0	49.0	59.6
	TURTLE (1-space) [8]	63.4	57.6	87.6	93.8	50.3	66.3	57.6	68.1
	TURTLE (2-space) [8]	67.9	64.6	89.6	96.0	48.4	97.8	57.3	74.5
	Task Arithmetic [16, 18]	63.8	62.1	72.0	77.6	65.1	94.0	52.2	69.5
	Weight Averaging [36]	65.3	63.4	71.4	71.7	52.8	87.5	50.1	66.0
	EMR-Merging++ [15]	83.2	90.7	96.8	99.6	99.1	99.7	82.7	93.1
	KALE (Ours)	**84.6**	**92.7**	**97.4**	99.7	**99.4**	**99.7**	**84.8**	**94.0**

Overall, these results validate KALE as a powerful label-free alternative to both supervised fine-tuning and existing unsupervised or weakly supervised baselines, offering a scalable and effective solution for performance enhancement via model aggregation.

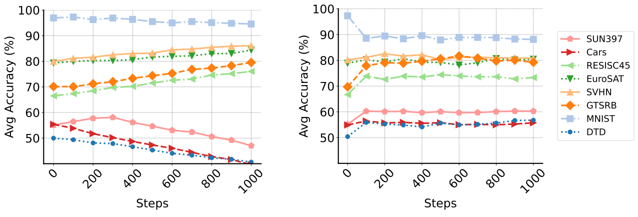
Effectiveness Analysis. To evaluate the effectiveness of our Joint Self-Cooperative Optimization strategy in addressing the error accumulation problem, we first visualize the variance of fusion coefficient magnitudes during the adaptation process of using Entropy Minimization and our proposed Joint-Cooperative optimization.



(a) Entropy Minimization **(b) Joint Optimization (Ours)**

Figure 5: Fusion coefficient variance during adaptation.

As shown in Figure 5, the Entropy Minimization baseline exhibits increasing variance in fusion weights over adaptation steps (ever deeper color in Figure 5a), indicating an overemphasis on certain tasks and a lack of coordination among task-specific updates. Moreover, EM is highly sensitive to the number of adaptation steps, requiring careful manual tuning to avoid performance degradation. This instability reflects the absence of an effective error correction mechanism, leading to a failure to converge reliably. In contrast, our method maintains consistently stable and balanced fusion weights throughout the adaptation process (Figure 5a). This stability suggests that our Joint Self-Cooperative Optimization promotes more robust and coordinated task interaction. By jointly evolving shared representations with implicit mutual correction, our method enables effective cross-task generalization and ensures convergence without the need for sensitive step control. To further validate this, we track average accuracy across the adaptation trajectory.



(a) Entropy Minimization **(b) Joint Optimization (Ours)**

Figure 6: Average accuracy during adaptation.

Figure 6 reveals that EM suffers from accuracy degradation over time on challenging datasets such as SUN397, Cars, and DTD. This reflects a typical error accumulation effect due to uncoordinated updates and task interference. In contrast, our Joint Optimization method achieves steady performance gains, converges in fewer than 100 steps, and maintains stability without any collapse.

These results confirm that Joint Self-Cooperative Optimization effectively mitigates error accumulation by encouraging shared representations to evolve in a coordinated manner across tasks. This leads to more robust and generalizable adaptation, validating the stability and effectiveness of our two-stage framework.

Efficiency Analysis. We further examine the optimization dynamics in the early stages of training. In addition to overall performance, we focus on the adaptation efficiency of KALE. To this end, we track its accuracy progression over optimization steps during Stage 2 (adaptive representation alignment) across all datasets. Figure 7 reports the results for ViT-B/32, where each curve shows the average accuracy over five runs, and the blue dashed line denotes the performance of fully supervised fine-tuning.

Remarkably, KALE converges on all benchmarks within 100 steps, often requiring substantially fewer updates. For example, competitive accuracy is achieved in just 20 steps on MNIST and around 30 steps on EuroSAT. This rapid convergence highlights the efficiency and stability of our method, enabled by the well-initialized representations from Stage-1 and the lightweight nature of Stage-2 adaptation. By minimizing optimization overhead while maintaining strong accuracy, KALE significantly reduces the cost of deployment, making it a highly practical solution for label-free model enhancement.

4.2 Ablation Studies

To better understand the contribution of each component in KALE, we conduct ablation studies on representative datasets using the ViT-B/32 backbone. We isolate and evaluate the effectiveness of three core modules: (i) the knowledge aggregation stage (Stage-1), (ii) the sample filtering strategy, and (iii) the weight transformation functions.

Effectiveness of Knowledge Aggregation. Since our approach is grounded in the learn-from-model paradigm, it is essential to first verify whether our Stage-1 process indeed aggregates useful knowledge from diverse models. To this end, we isolate the fused model produced by Stage-1—before any downstream adaptation—and evaluate its multi-task generalization ability. This allows us to directly quantify the breadth and effectiveness of knowledge integration. We compare against a comprehensive set of state-of-the-art model fusion techniques, all of which are label-free, ensuring a fair comparison. Specifically, we compare our approach against a suite of state-of-the-art model fusion baselines, including Task Arithmetic (TA)[16], TIES-Merging (TM)[30], RegMean (RM)[18], Fisher-Merging (FM)[28], Model Soups (MS)[36], and AdaMerging (AD)[41].

As shown in Figure 8, our method—denoted in pink in the final column—consistently achieves the best accuracy on 7 out of 8 tasks, demonstrating robust and superior multi-task learning capabilities. In contrast, baseline fusion methods often suffer from performance drops on specific datasets due to parameter conflicts or insufficient integration of model-specific knowledge. Notably, KALE excels on complex benchmarks with large class spaces, such as SUN397, Cars, and DTD, where effective aggregation of complementary knowledge is crucial. These results confirm that our fusion strategy significantly broadens the scope of transferable knowledge, forming a solid foundation for the subsequent Stage-2 specialization.

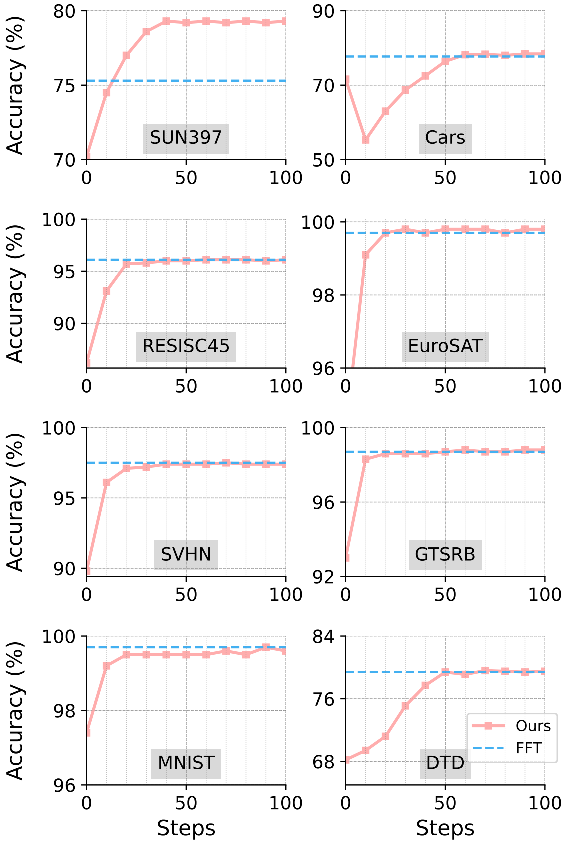


Figure 7: Adaptation efficiency of KALE using ViT-B/32.

Effectiveness of Class Cardinality-aware Sample Filtering. To assess the impact of our class-cardinality-based sample filtering strategy, we compare model performance with and without this filtering under identical experimental settings. The only distinction lies in the application of our filtering method.

As illustrated in Figure 9, our filtering method consistently leads to superior performance at every adaptation step across models of varying capacities. Notably, models employing the filtering converge significantly faster, demonstrating improved learning efficiency and reduced training time. This accelerated convergence suggests that the filtered training set better prioritizes informative and representative samples, thereby facilitating more effective knowledge integration during adaptation. These findings underscore the critical role of filtering out redundant, noisy, or low-value samples that could otherwise dilute the learning signal. By focusing on a carefully selected subset aligned with class cardinality, our method enables more focused gradient updates and alleviates potential interference during knowledge aggregation. Ultimately, this results in not only higher accuracy but also more stable and efficient model specialization. This component is thus essential for maximizing the benefits of our multi-stage framework.

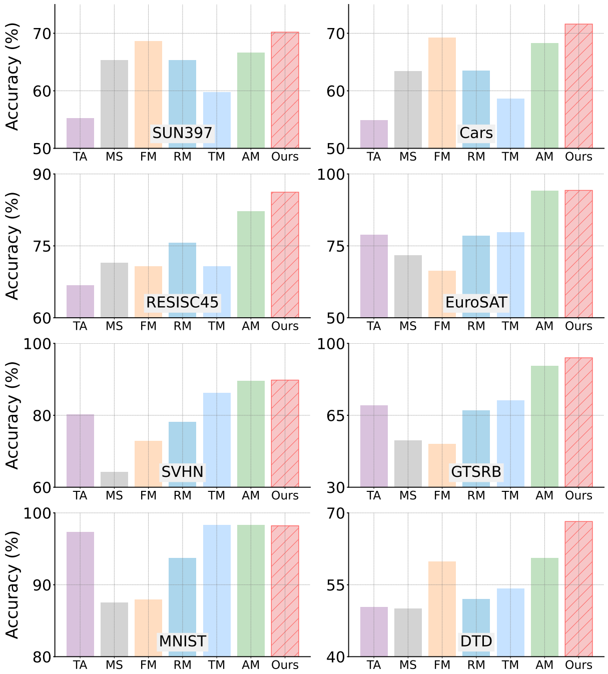


Figure 8: MTL performance comparison.

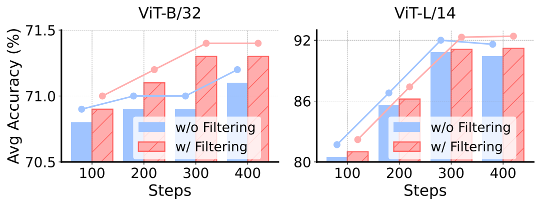
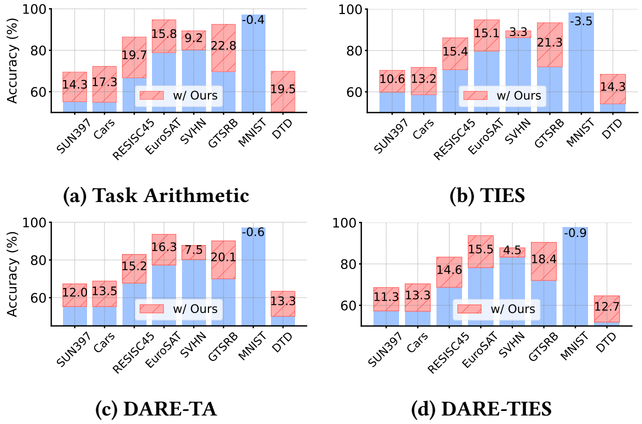


Figure 9: Effect of Class Cardinality-aware Filtering.

Effect of Weight Transformation Functions. Weight transformation functions $\Phi(\cdot)$ are key to model fusion, defining how parameters from multiple models combine into an initial fused model (Equation 2). As none guarantee lossless knowledge fusion, it is crucial to test whether our Joint Self-Cooperative Optimization consistently improves fusion quality across various initialization. To this end, we experiment with several representative weight transformation functions: standard Task Arithmetic (TA), TIES-Merging (TM), DARE-enhanced Task Arithmetic (DARE [42]+TA), and DARE-enhanced TIES-Merging (DARE+TIES). These methods reflect diverse approaches to parameter alignment, weighting, and fusion, encompassing both simple arithmetic and adaptation-based techniques.

As shown in Figure 10, when combined with our Joint Self-Cooperative Optimization, all these initializations exhibit substantial and consistent performance gains—achieving average accuracy improvements of 14.7%, 11.2%, 12.2%, and 11.2%, respectively. This consistent uplift demonstrates that, regardless of the initial fusion quality dictated by the choice of Φ, our optimization procedure robustly refines and enhances the fused model, effectively mitigating parameter conflicts and knowledge interference. These results validate the generality and effectiveness of our joint optimization framework, highlighting its critical role in unlocking the full potential of diverse weight transformation strategies. It ensures reliable knowledge aggregation and downstream task specialization, even when starting from imperfect initial fusions.



(a) Task Arithmetic

(b) TIES

(c) DARE-TA

(d) DARE-TIES

Figure 10: Effect of Weight Transformation Functions.

5 Conclusion

In this paper, we propose KALE, a label-free framework for model enhancement via knowledge aggregation and few-shot specialization. KALE eliminates the need for labeled data by fusing multiple models and introducing a joint self-cooperative optimization objective that aligns task- and layer-specific knowledge, integrates self and cooperative supervision, and mitigates error accumulation caused by entropy minimization. To ensure stable fusion, KALE also employs a class cardinality-aware sample filtering strategy. Experimental results show that our method outperforms full fine-tuning in the diverse image classification tasks on two of the Vision Transformer backbones (i.e., ViT-B/32 and ViT-L/14), with notable improvements on hard tasks SUN397 by over 4 points. It effectively tackles data scarcity, enabling fine-tuning in environments with limited or inaccessible labeled data. Additionally, our method achieves competitive performance in knowledge aggregation, surpassing SOTA model fusion methods by over 3 points on multi-task benchmarks. Moreover, the efficiency and effectiveness of each component are rigorously evaluated. These results highlight KALE's capability to enhance large models without relying on labeled data.

GenAI disclosure statement

Generative AI was used to fix grammatical issues in the paper.

Acknowledgments

This work is supported by the Guangzhou Industrial Information and Intelligent Key Laboratory Project (No. 2024A03J0628). It is also funded by the NSFC Project (No. 62306256) and the Natural Science Foundation of Guangdong Province (No. 2025A1515010261).

References

[1] Guillaume Alain and Yoshua Bengio. 2018. Understanding intermediate layers using linear classifier probes, 2017. In *URL https://openreview. net/forum*. ICLR, Vancouver, Canada, 1–13.

[2] Paola Cascante-Bonilla, Fuwen Tan, Yanjun Qi, and Vicente Ordonez. 2021. Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. AAAI Press, Virtual Conference, 6912–6920.

[3] Gong Cheng, Junwei Han, and Xiaoqiang Lu. 2017. Remote sensing image scene classification: Benchmark and state of the art. *Proc. IEEE* 105, 10 (2017), 1865–1883.

[4] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE Computer Society, Columbus, Ohio, USA, 3606–3613.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* abs/1810.04805, – (2018), –. arXiv:1810.04805.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* –, – (2020), 1–22. https://arxiv.org/abs/2010.11929.

[7] Xun Fu, Wen-Bo Xie, Bin Chen, Tao Deng, Tian Zou, and Xin Wang. 2024. ACDM: An Effective and Scalable Active Clustering with Pairwise Constraint. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. ACM, Birmingham, United Kingdom, 643–652.

[8] Artyom Gadetsky, Yulun Jiang, and Maria Brbic. 2024. Let Go of Your Labels with Unsupervised Transfer. In *International Conference on Machine Learning*. PMLR, Proceedings of Machine Learning Research, Vienna, Austria, 14382–14407.

[9] Charles Goddard, Shamane Siriwardhana, Malikeh Ehghaghi, Luke Meyers, Vladimir Karpukhin, Brian Benedict, Mark McQuade, and Jacob Solawetz. 2024. Arcee's mergekit: A toolkit for merging large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*. Association for Computational Linguistics, Singapore, 477–485.

[10] Jie Gui, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao. 2024. A survey on self-supervised learning: Algorithms, applications, and future trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence* –, – (2024).

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. IEEE/CVF, New Orleans, LA, USA, 16000–16009.

[12] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 12, 7 (2019), 2217–2226.

[13] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR* 1, 2 (2022), 3.

[14] Chao Huang, Lianghao Xia, Xiang Wang, Xiangnan He, and Dawei Yin. 2022. Self-supervised learning for recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. ACM, Atlanta, GA, USA, 5136–5139.

[15] Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. 2024. EMR-Merging: Tuning-Free High-Performance Model Merging. *arXiv preprint arXiv:2405.17461* –, – (2024), –.

[16] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. *arXiv preprint arXiv:2211.02016* abs/2211.02016 (2023), –. arXiv:2211.02016.

[17] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. 2019. Consistency-based semi-supervised learning for object detection. *Advances in neural information processing systems* 32 (2019).

[18] Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2023. Dataless Knowledge Fusion by Merging Weights of Language Models. *The Eleventh International Conference on Learning Representations* (2023).

[19] Aghiles Kebaili, Jérôme Lapuyade-Lahorgue, and Su Ruan. 2023. Deep learning approaches for data augmentation in medical imaging: a review. *Journal of Imaging* 9, 4 (2023), 81.

[20] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*. PMLR, PMLR, New York, USA, 3519–3529.

[21] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*. IEEE Computer Society, Sydney, Australia, 554–561.

[22] Yann LeCun. 1998. The MNIST database of handwritten digits. *http://yann. lecun. com/exdb/mnist/* (1998).

[23] Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. 2024. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. *International Conference on Learning Representations* (2024).

[24] Jingjing Li, Zhiqi Yu, Zhekai Du, Lei Zhu, and Heng Tao Shen. 2024. A comprehensive survey on source-free domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* – (2024), –.

[25] Zhongzhou Liu, Hao Zhang, Kuicai Dong, and Yuan Fang. 2024. Collaborative Cross-modal Fusion with Large Language Model for Recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. ACM, Birmingham, United Kingdom, 1565–1574.

[26] Jinliang Lu, Ziliang Pang, Min Xiao, Yaochen Zhu, Rui Xia, and Jiajun Zhang. 2024. Merge, Ensemble, and Cooperate! A Survey on Collaborative Strategies in the Era of Large Language Models. *arXiv preprint arXiv:2407.06089* – (2024), –.

[27] James MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, Vol. 5. University of California press, University of California Press, Berkeley, CA, USA, 281–298.

[28] Michael S Matena and Colin A Raffel. 2022. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems* 35 (2022), 17703–17716.

[29] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. 2024. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems* 36 (2024).

[30] Ori Press, Ravid Shwartz-Ziv, Yann LeCun, and Matthias Bethge. 2024. The Entropy Enigma: Success and Failure of Entropy Minimization. *International Conference on Machine Learning* –, – (2024), –.

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Goh, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. 8748–8763.

[32] Surangika Ranathunga, En-Shiun Annie Lee, Marjana Prifti Skenduli, Ravi Shekhar, Mehreen Alam, and Rishemjit Kaur. 2023. Neural machine translation for low-resource languages: A survey. *Comput. Surveys* 55, 11 (2023), 1–37.

[33] Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. 2021. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. Association for Computing Machinery, Virtual Event, Canada, 2443–2449.

[34] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. 2011. The German traffic sign recognition benchmark: a multi-class classification competition. In *The 2011 international joint conference on neural networks*. IEEE, IEEE, San Jose, California, USA, 1453–1460.

[35] Lei Wang, Ee-Peng Lim, Zhiwei Liu, and Tianxiang Zhao. 2022. Explanation guided contrastive learning for sequential recommendation. In *Proceedings of the 31st ACM international conference on information & knowledge management*. ACM, Atlanta, GA, USA, 2017–2027.

[36] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*. PMLR, 23965–23998.

[37] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. 2022. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7959–7971.

[38] Jianxiong Xiao, Krista A Ehinger, James Hays, Antonio Torralba, and Aude Oliva. 2016. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision* 119 (2016), 3–22.

[39] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems* 33 (2020), 6256–6268.

[40] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2023. Resolving Interference When Merging Models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*. NeurIPS, NeurIPS, New Orleans, Louisiana, USA, 1234–1245.

[41] Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. 2024. AdaMerging: Adaptive Model Merging for Multi-Task Learning. *International Conference on Learning Representations* (2024).

[42] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*.

[43] Hongling Zheng, Li Shen, Anke Tang, Yong Luo, Han Hu, Bo Du, Yonggang Wen, and Dacheng Tao. 2025. Learning from models beyond fine-tuning. *Nature Machine Intelligence* 7, 1 (2025), 6–17.