Self-Distillation as a Performance Recovery Mechanism for LLMs: Counteracting Compression and Catastrophic Forgetting

Anonymous authors
Paper under double-blind review

Abstract

Large Language Models (LLMs) have achieved remarkable success, underpinning diverse AI applications. However, they often suffer from performance degradation due to factors such as catastrophic forgetting during Supervised Fine-Tuning (SFT), quantization, and pruning. In this work, we introduce a performance recovery framework based on Self-Distillation Fine-Tuning (SDFT) that effectively restores model capabilities. Complementing this practical contribution, we provide a rigorous theoretical explanation for the underlying recovery mechanism. We posit that an LLM's generative capability fundamentally relies on the high-dimensional manifold constructed by its hidden layers. To investigate this, we employ Centered Kernel Alignment (CKA) to quantify the alignment between student and teacher activation trajectories, leveraging its invariance to orthogonal transformations and scaling. Our experiments demonstrate a strong correlation between performance recovery and manifold alignment, substantiating the claim that self-distillation effectively aligns the student's high-dimensional manifold with the optimal structure represented by the teacher. This study bridges the gap between practical recovery frameworks and geometric representation theory, offering new insights into the internal mechanisms of self-distillation.

1 Introduction

Large Language Models (LLMs) have revolutionized natural language understanding, reasoning, and generation. However, deploying generic base models into real-world applications necessitates further adaptation. To align with specific downstream tasks, models typically undergo Supervised Fine-Tuning (SFT); simultaneously, to meet resource constraints, techniques such as pruning and quantization become indispensable.

However, these operations often incur significant performance degradation. In continuous learning, multi-round SFT frequently triggers Catastrophic Forgetting, where models lose original general knowledge and skills while acquiring new domain-specific knowledge and task capabilities. Similarly, aggressive compression disrupts internal parameter distributions, leading to declines in accuracy and logical consistency. This "capability trade-off" forces difficult choices between specialization and generalization. Once a model degrades, traditional repair methods are often computationally prohibitive, sometimes requiring retraining from scratch, a very inefficient solution in the context of scarce computational resources.

In this paper, we propose an effective "Recovery Mechanism" for model degradation, leveraging Self-Distillation Fine-Tuning (SDFT) (Shenfeld et al., 2026), a specialized paradigm of Self-Distillation (SD) (Hinton et al., 2015). While traditional SD focuses on improving generalization bounds through self-imitation, we argue that when a model suffers from distribution shift due to SFT or compression, the regularization effect of SDFT acts as an "anchor." This mechanism pulls degraded parameters back toward the original high-performance manifold. Crucially, our approach relies solely on the model's own historical states without relying on an external teacher, thereby facilitating efficient performance recovery.

Building on this insight, we establish a unified recovery framework and validate it across diverse degradation scenarios. With a primary emphasis on catastrophic forgetting in multi-round SFT, we further demonstrate

the framework's efficacy against compression artifacts. Empirical results demonstrate that SDFT effectively restores model performance across multiple evaluation benchmarks, validating both its practical efficacy and theoretical foundation.

2 Related Work

Catastrophic Forgetting in LLMs. Catastrophic Forgetting (CF) refers to the phenomenon wherein neural networks to suddenly and significantly lose previously learned knowledge when trained on new data (De Lange et al., 2021). In the context of LLMs, this phenomenon manifests when multi-round Supervised Fine-Tuning (SFT) overwrites the knowledge and skills acquired in previous trainings (Li & Hoiem, 2017). Existing mitigation strategies generally fall into three categories: (1) Replay-based methods, which store a subset of old data to interleave with new training (De Lange et al., 2021); (2) Regularization-based methods, such as Elastic Weight Consolidation (EWC), which penalize changes to important parameters (Kirkpatrick et al., 2017); and (3) Parameter-isolation methods, which allocate separate parameters for different tasks (Rusu et al., 2016). While effective to some extent, these approaches often incur high computational costs, require access to historical data, or complicate model architecture. Crucially, most existing work focuses on preventing forgetting during new training, rather than recovering performance after degradation has occurred.

Model Compression and Performance Degradation. To deploy LLMs efficiently, techniques such as pruning (Ma et al., 2024) and quantization (Dettmers et al., 2023) are widely adopted. However, these operations inevitably introduce performance degradation. Aggressive pruning removes redundant neurons but may disrupt critical knowledge pathways, while low-bit quantization introduces noise that affects logical consistency and factual accuracy (Frantar et al., 2023). Traditional remedies often rely on Knowledge Distillation (KD), where a compressed student model is trained to mimic a larger teacher (Hinton et al., 2015). While external strong teachers (e.g., larger LLMs or API-based models) are theoretically applicable, they often introduce distribution shifts, high computational overhead, or privacy constraints that limit their practicality for post-degradation recovery. In contrast, Self-Distillation offers a self-contained alternative that leverages the model's own historical states, avoiding external dependencies while preserving task alignment. This makes SD particularly suitable for lightweight, privacy-sensitive, or distribution-consistent recovery scenarios.

Self-Distillation Fine-Tuning. Self-Distillation (SD) has emerged as a powerful technique for enhancing model generalization without relying on an external teacher. Early works demonstrated that training a model to mimic its own deeper layers or earlier checkpoints acts as an effective regularizer, reducing overfitting and improving accuracy (Furlanello et al., 2018). More recently, studies have extended SD to Self-Distillation Fine-Tuning (SDFT), enabling on-policy learning directly from demonstrations. By leveraging in-context learning, SDFT uses the model itself as a teacher to generate training signals that preserve prior capabilities while acquiring new skills. Across various tasks, SDFT consistently outperforms conventional SFT, achieving higher new-task accuracy while mitigating catastrophic forgetting. However, existing SDFT approaches primarily focus on preventing forgetting during the training process, often assuming the teacher and student are synchronized. In this paper, we extend SDFT to a more general framework where the teacher can be any historical state of the model, not just the current iteration. Crucially, we reposition this generalized SDFT as a post-hoc recovery mechanism, designed to restore performance after degradation has occurred, rather than merely preventing it during training.

3 Recovery Framework

3.1 Problem Formulation

Let LLM θ denote the original base model with parameters θ. After undergoing degradation processes such as multi-round SFT or compression, the model becomes LLM θ_1 with parameters θ_1, exhibiting performance drops in general knowledge and skills. Our goal is to obtain a recovered model LLM θ_2 with parameters θ_2 that maximizes performance on both original capabilities and new tasks.

3.2 Recovery Solutions

Figure 1 illustrates the overall architecture of our proposed Self-Distillation Recovery Framework in catastrophic forgetting scenario. Unlike traditional Fine-Tuning pipelines that solely optimize for new task performance, our framework introduces a dual-objective optimization process aimed at both capability recovery and task adaptation. The framework consists of three main components: (1) the Teacher LLM θ, constructed from the model's own historical checkpoints or earlier training states; (2) the Degraded Model θ_1, which serves as the initial student state suffering from performance loss due to prior multiple rounds of SFTs; and (3) the SDFT Recovery Process, where the student learns to mimic the teacher's output distribution while adapting to the datasets used in previous multiple rounds of SFTs. This self-contained process ensures that performance recovery is achieved without relying on external high-performance models or any external datasets.

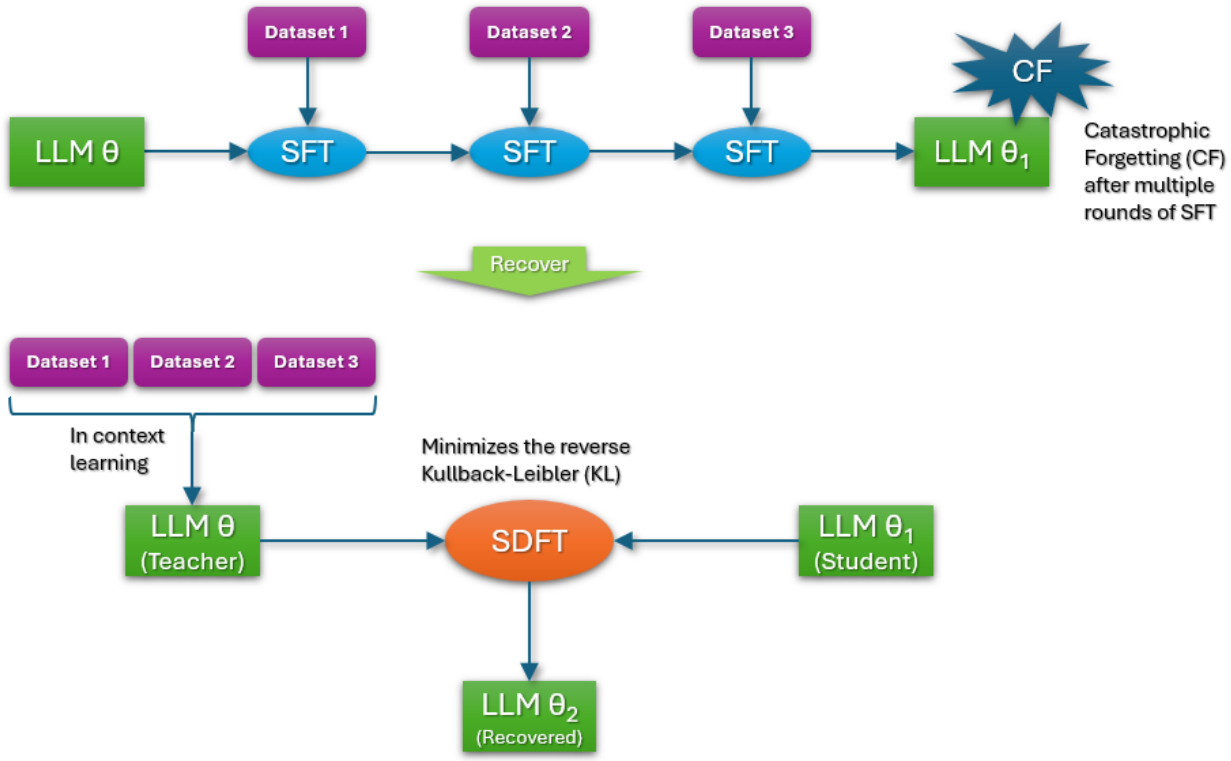

Figure 1: The Self-Distillation Recovery Framework for Catastrophic Forgetting

Figure 2 extends the proposed recovery framework to compression scenarios. When an LLM is subjected to pruning or quantization, it inevitably incurs varying degrees of performance degradation. To facilitate recovery, the framework necessitates the curation of expert demonstration datasets aligned with the degraded capabilities. For example, if the tool-calling task shows performance degradation, related datasets are needed for recovery; if general knowledge shows degradation, then SFT datasets used in post-training are required. Notably, apart from this data selection strategy, the underlying recovery mechanism remains identical to the catastrophic forgetting scenario, demonstrating the unified nature of our approach across different degradation types.

However, the original SDFT formulation exhibits a significant limitation at smaller scales (e.g., 3B variants), where insufficient in-context learning (ICL) capabilities fail to provide meaningful self-guidance, resulting in performance inferior to standard SFT. To address this, we propose an extended recovery strategy that introduces a single preliminary step while preserving the unified nature of our framework.

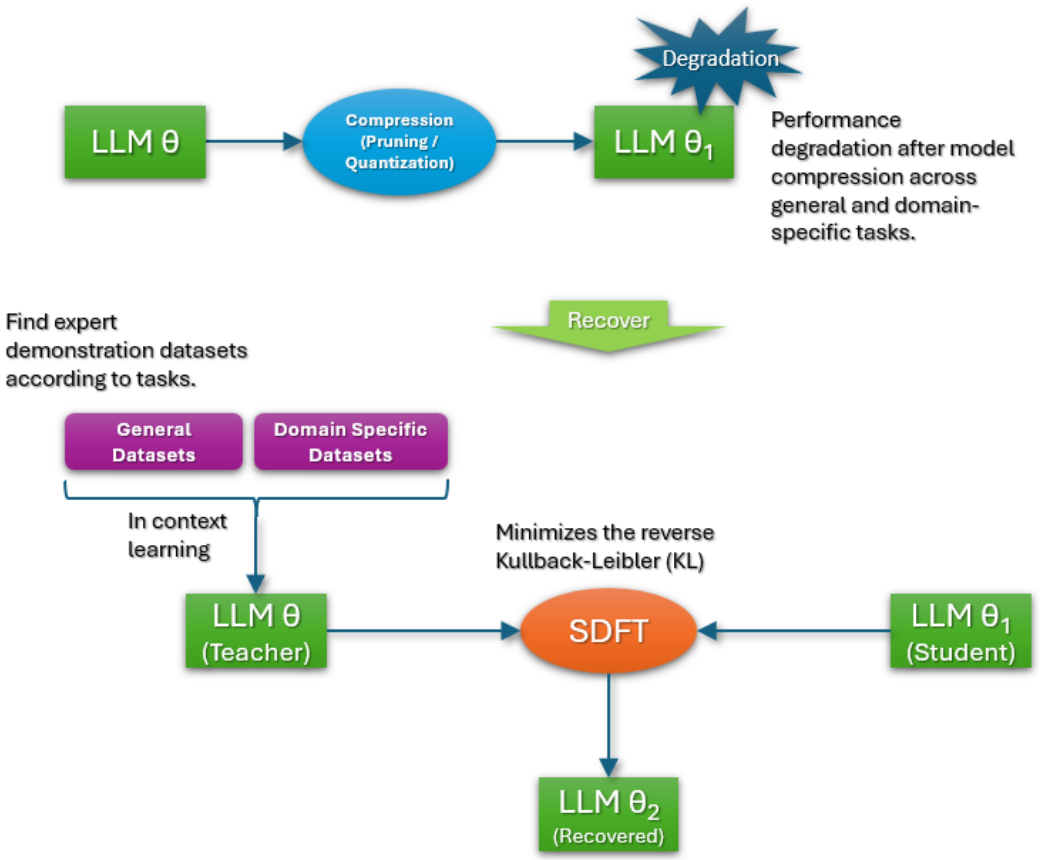

Figure 2: The Self-Distillation Recovery Framework for Compression

Figure 3 illustrates this enhanced workflow. The ineffectiveness of SDFT in small-scale models stems from its heavy reliance on robust ICL, which is typically underdeveloped in smaller architectures. Consequently, we first employ off-policy distillation using a large-scale LLM as the teacher to bootstrap the small model's ICL capabilities. While this step enhances ICL, it inevitably leads to degradation in general and domain-specific capabilities. Subsequently, we apply our SDFT recovery mechanism to restore these degraded capabilities. Ultimately, this two-stage process enables the small-scale model to retain its original capabilities while achieving improved ICL performance, effectively extending the applicability of our recovery framework to resource-constrained scenarios.

The external teacher is used only once to bootstrap ICL capabilities (enabling SDFT), whereas the core recovery process remains self-contained via SDFT. This hybrid approach balances practicality with the efficiency of self-distillation.

4 Theoretical Analysis of Self-Distillation via High-Dimensional Manifold Alignment

4.1 Introduction

Previous chapters have primarily focused on the empirical analysis of the recovery framework, leaving the underlying theoretical mechanisms unexplored. Why does self-distillation effectively recover model performance, and is there a geometric metric aligned with this phenomenon?

In this chapter, we answer these questions by shifting the focus from output distributions to internal representations. We posit that the generative capability of an LLM fundamentally relies on the high-dimensional manifold constructed by its hidden layers, and consequently the core function of self-distillation is not merely

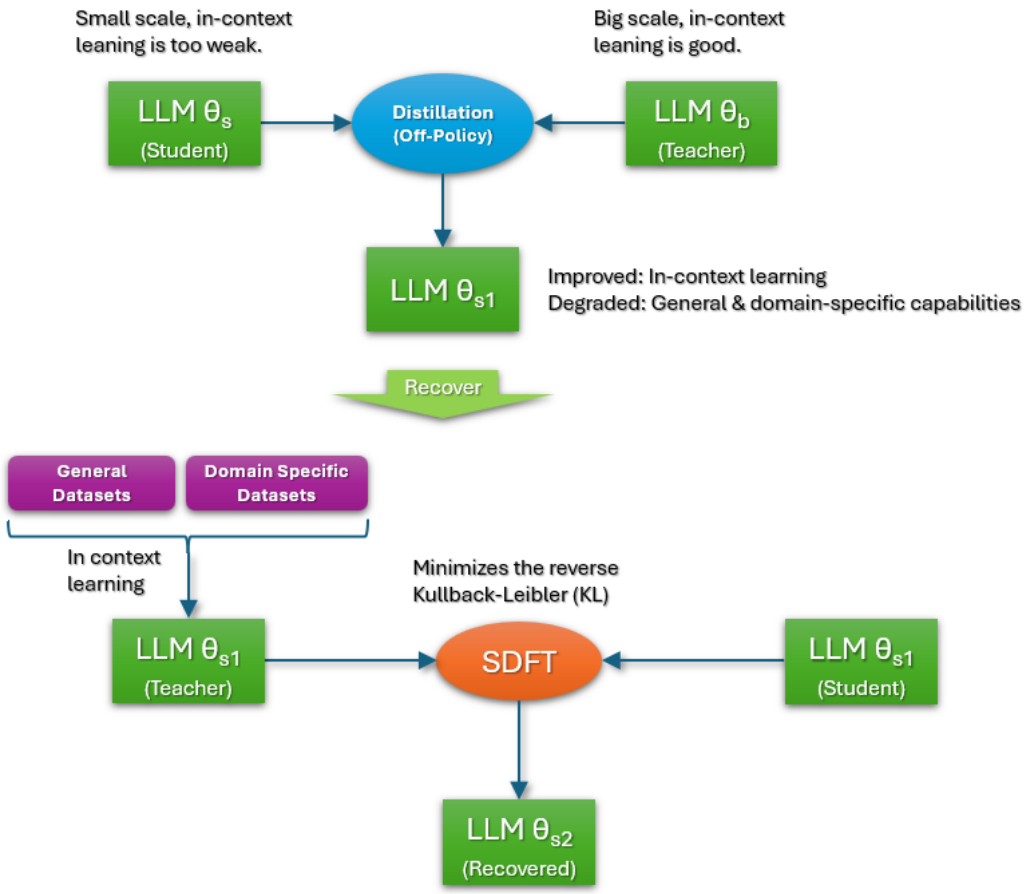

Figure 3: The Self-Distillation Recovery Framework for Small Scale LLM

optimizing output probabilities but regularizing the spatial structure of hidden states to align the student's manifold with the teacher's. Building on this premise, we propose a theoretical framework grounded in high-dimensional manifold geometry.

To validate this theoretical framework, we employ Centered Kernel Alignment (CKA) (Kornblith et al., 2019) as a metric to quantify the alignment of manifold structures between the student and the teacher, leveraging its critical advantage over metrics like Mean Squared Error (MSE) — namely, invariance to orthogonal transformations and scaling.

4.2 Problem Formulation and Manifold Definition

Given an input sequence $X = (x_1, x_2, \ldots, x_L)$, where L denotes the sequence length. For a certain hidden layer of an LLM (e.g., the last hidden layer), each token x_t corresponds to a d-dimensional activation vector $h_t \in \mathbb{R}^d$. We stack the activation vectors of all tokens from a complete forward pass to form the Activation Matrix $H \in \mathbb{R}^{L \times d}$:

$$H = \begin{bmatrix} h_1^T \\ h_2^T \\ \vdots \\ h_L^T \end{bmatrix} \tag{1}$$

From the perspective of manifold learning, each row in H represents a sample point on the high-dimensional semantic manifold $\mathcal{M}$, and the entire matrix H constitutes a discrete trajectory of the sequence on this

manifold. The student model S and the teacher model T generate activation matrices H_S and H_T, respectively. Our objective is to measure the geometric alignment between these two trajectories. It is important to clarify that we do not compare the complete underlying manifolds of the student and teacher models directly. Instead, we utilize activation trajectories, which serve as discrete samples from these manifolds. This approach is both theoretically representative and computationally feasible.

Directly comparing the element values of activation matrices H_S and H_T (e.g., using MSE) is inappropriate because neural network representations possess rotation invariance. Semantically identical features may exist along different coordinate axes in the hidden space. To capture the intrinsic structure of the manifold, we must measure the relative relationships between tokens rather than their absolute coordinates.

We compute the Linear Kernel Matrix $K \in \mathbb{R}^{L \times L}$:

$$K = HH^T \tag{2}$$

Here, the element $K_{ij} = h_i \cdot h_j$ represents the similarity between the i-th and j-th tokens in the hidden space. The matrix K encodes the semantic dependency structure within the sequence and serves as a representation of the geometric properties of the manifold.

4.3 Calculation Procedure

We follow the six steps below to calculate the manifold alignment degree between H_S and H_T:

1. **Input Consistency:** Input the identical sequence (Prompt + Ground Truth) into both the student and teacher models to ensure one-to-one correspondence of token positions.
2. **Activation Extraction:** Extract the same layer activation matrices $H_S, H_T \in \mathbb{R}^{L \times d}$.
3. **Kernel Matrix Computation:** Compute the linear kernel matrices $K_S = H_S H_S^T$ and $K_T = H_T H_T^T$.
4. **Centering Operation:** Construct the centering matrix $C = I_L - \frac{1}{L}\mathbf{1}\mathbf{1}^T$. Compute the centered kernel matrices $K_{SC} = CK_SC$ and $K_{TC} = CK_TC$. This step eliminates global biases in activation values, ensuring the metric focuses solely on relative structure.
5. **Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al., 2005) Computation:** Compute the Frobenius inner product of the two centered kernel matrices, which is the Trace of their product:
$$\text{HSIC}(H_S, H_T) = \text{tr}(K_{SC}K_{TC}) = \text{tr}(K_SCK_TC) \tag{3}$$
The original HSIC definition includes a scaling factor $\frac{1}{(L-1)^2}$ for unbiased estimation, as the factor cancels out in the normalized CKA ratio and is thus omitted for simplicity.
6. **CKA Normalization:** The final alignment score is calculated as:
$$\text{CKA}(H_S, H_T) = \frac{\text{HSIC}(H_S, H_T)}{\sqrt{\text{HSIC}(H_S, H_S) \cdot \text{HSIC}(H_T, H_T)}} \tag{4}$$

The value of CKA ranges from $[0, 1]$. A score closer to 1 indicates that the geometric structure of the student's activation trajectory highly coincides with that of the teacher, implying the student has successfully recovered the high-dimensional manifold constructed by the teacher.

4.4 Theoretical Explanation

In summary, our theoretical analysis posits that self-distillation can recover LLM performance because LLM generative capability fundamentally relies on the high-dimensional manifold constructed by the hidden layers, and self-distillation can align the student's manifold with the teacher's optimal manifold structure. Furthermore, we identify CKA as a robust metric to quantify this degree of manifold alignment.

Table 1: Three-stage forgetting–recovery pipeline on Qwen2.5-3B-Instruct. SFT on Tooluse induces catastrophic forgetting of Science (37.87%, down from 59.96%). Recovery SDFT restores Science while preserving Tooluse.

Pipeline Stage	Tooluse	Science
Science expert	19.59%	59.96%
+ SFT Tooluse	64.95%	37.87%
+ Recovery (t=base)	65.98%	61.54%
+ Recovery (t=expert)	57.73%	65.48%

Based on this theory, we have constructed a comprehensive analysis framework that mathematically formalizes activation trajectories as manifold samples and derived a CKA-based alignment scoring method.

In the following chapter, we will show empirical results that validate our theoretical analysis.

5 Experiments

This section empirically validates the recovery framework (Section 3) and the manifold alignment theory (Section 4) across the three degradation scenarios proposed in Figures 1–3: catastrophic forgetting (Section 5.1), compression (Section 5.2), and small-model bootstrapping (Section 5.3). Experiments are conducted on Qwen2.5-3B-Instruct and Qwen2.5-7B-Instruct (Qwen et al., 2024) across two task domains — Tooluse (structured tool-use) and Science (scientific QA) — with general capability preservation assessed on MMLU and Winogrande (5-shot). Compression is applied via NF4 quantization and 10% structured FFN pruning, with recovery via standard SDFT as described in Section 3.

5.1 Recovery from Catastrophic Forgetting

We first validate the recovery framework (Section 3) and the manifold alignment theory (Section 4) on catastrophic forgetting: a model trained on task A loses its capabilities after subsequent SFT on task B. We construct a three-stage pipeline on Qwen2.5-3B-Instruct — (1) train an SDFT expert on Science, (2) apply standard SFT on Tooluse (inducing forgetting), (3) apply recovery SDFT — and measure both task accuracy and last-layer CKA against the original Science expert at each stage.

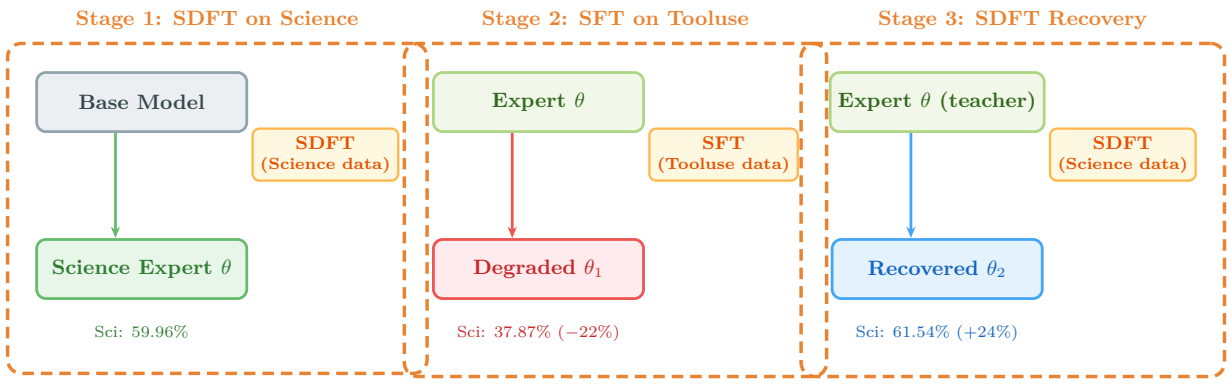

Figure 4: Three-stage forgetting–recovery pipeline. Stage 1 trains a Science expert via SDFT. Stage 2 applies standard SFT on Tooluse, inducing catastrophic forgetting of Science. Stage 3 applies recovery SDFT to restore Science capabilities.

Results. Table 1 demonstrates that recovery SDFT effectively reverses catastrophic forgetting. Recovery with t=base restores Science from 37.87% to 61.54% (+23.67%) while preserving Tooluse at 65.98%, demonstrating that both task capabilities can coexist after recovery. Recovery with t=expert further boosts

Table 2: Task-specific accuracy across the quantization recovery pipeline. SDFT not only recovers capabilities lost to quantization but actively enhances them beyond the original bf16 model.

Config	3B Tooluse	3B Science	7B Tooluse	7B Science
bf16 (θ)	29.20%	31.80%	42.40%	33.90%
NF4 (θ_1)	28.70%	30.40%	41.60%	33.00%
SDFT (θ_2)	**50.52%**	**45.36%**	**62.89%**	**52.30%**
SDFT gain ($\theta_2 - \theta_1$)	+21.82%	+14.96%	+21.29%	+19.30%
Net vs bf16 ($\theta_2 - \theta$)	+21.32%	+13.56%	+20.49%	+18.40%

Science to 65.48% at a moderate Tooluse trade-off (57.73%). The manifold alignment analysis underlying this recovery is presented in Section 5.4.

5.2 Recovery from Compression

We next validate the recovery framework on compression-induced degradation, testing two complementary methods — NF4 quantization and structured FFN pruning — which produce fundamentally different degradation patterns.

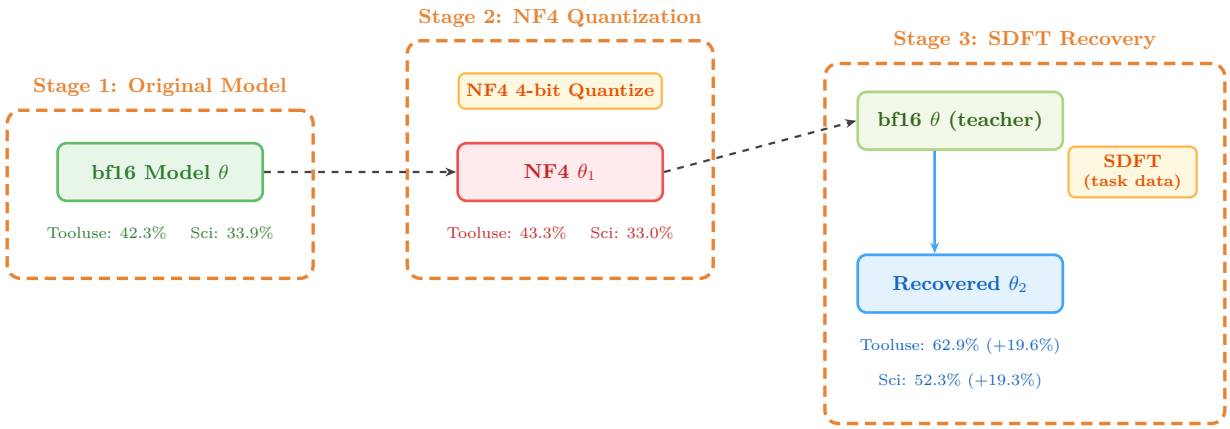

Figure 5: Compression recovery pipeline. Stage 1: original bf16 model θ. Stage 2: NF4 quantization produces θ_1. Stage 3: SDFT recovers θ_2 on task-specific data using θ as teacher.

Figure 5 illustrates the three-stage pipeline. Starting from the original bf16 model θ (Stage 1), NF4 quantization compresses θ to a 4-bit model θ_1 (Stage 2), which largely preserves task accuracy but introduces latent manifold misalignment. In Stage 3, SDFT uses θ as a static teacher to distill task-specific knowledge into θ_1 on target-domain data, producing the recovered model θ_2 — which substantially exceeds both θ and θ_1.

Quantization: task-specific recovery. Table 2 shows that SDFT yields +15–22% task-specific gains across all configurations, with recovered models substantially exceeding the original bf16 θ. This validates the core prediction of the recovery framework: on-policy distillation anchors parameters near the pre-compression manifold while simultaneously adapting to the task distribution.

Quantization: general capability preservation. Table 3 confirms that SDFT does not trade task gains for general degradation. NF4 quantization introduces modest compression loss (−1.16% for 3B, −1.03% for 7B), and SDFT actively recovers this gap rather than widening it. The best 7B variant restores 75% of the compression loss (+0.77%), validating the anchoring mechanism proposed in Section 1: on-policy distillation stabilizes the parameter distribution, a phenomenon we formalize as manifold realignment in Section 4.

Table 3: General capability preservation under quantization. SDFT achieves +15–22% task-specific gains (Table 2) while actively recovering general capabilities: the best 7B variant restores 75% of the compression loss (+0.77% of −1.03%).

	3B			7B		
Config	MMLU	Wino	Avg	MMLU	Wino	Avg
bf16 (θ)	65.47	68.75	67.11	71.80	70.48	71.14
NF4 (θ_1)	64.34	67.56	65.95	71.08	69.14	70.11
SDFT-tooluse (θ_2)	64.25	67.72	65.99	71.20	69.53	70.37
SDFT-science (θ_2)	64.55	67.96	66.26	71.20	70.56	70.88
Compression loss ($\theta_1-\theta$)		−1.16%			−1.03%	
Best SDFT recovery ($\theta_2-\theta_1$)		+0.31%			+0.77%	

Table 4: SDFT recovery on FFN-pruned Qwen2.5-7B-Instruct (10% pruning). SDFT-tooluse recovers 64% of MMLU degradation (+4.57%) while achieving +5.56% on Tooluse. SDFT-science trades MMLU (−1.43%) for stronger Science gains (+9.73%).

Config	Tooluse	Science	MMLU	MMLU Δ vs θ_1
Original (θ)	29.86%	38.78%	66.21%	—
Pruned (θ_1)	28.52%	33.66%	59.09%	—
SDFT-tooluse (θ_2)	**34.08%**	35.75%	63.66%	+4.57%
SDFT-science (θ_2)	32.99%	**43.39%**	57.66%	−1.43%
Pruning degradation ($\theta_1-\theta$)	−1.34%	−5.12%	−7.12%	
Best SDFT recovery ($\theta_2-\theta_1$)	+5.56%	+9.73%	+4.57%	
Best net vs θ ($\theta_2-\theta$)	+4.22%	+4.61%	−2.55%	

Pruning: a harder recovery problem. Unlike quantization, pruning physically removes neurons, producing more severe and asymmetric degradation (−7.12% MMLU). Table 4 shows that SDFT still recovers effectively: SDFT-tooluse restores 64% of the MMLU gap while exceeding θ on both target tasks. Notably, SDFT exhibits positive cross-domain transfer — SDFT-science improves Tool-use from 28.52% to 32.99% (+4.47%) without any tool-use training data, surpassing even θ (29.86%). This indicates that SDFT recovers general representational capacity rather than memorizing task-specific patterns.

Cross-compression comparison. Figure 6 summarizes the contrast: quantization recovery yields +15–22% task gains while actively recovering general capabilities, whereas pruning recovery demonstrates stronger cross-domain transfer but incomplete MMLU restoration (−2.55% net vs θ). This difference reflects the nature of each degradation — quantization introduces noise while preserving architecture; pruning permanently removes capacity. Together, these results confirm that SDFT operates as a general-purpose recovery mechanism across compression types, consistent with the framework proposed in Section 3.

5.3 Extending to Small Models

The recovery framework relies on the teacher's in-context learning quality to generate effective training signals (Section 3). At smaller scales such as 3B, ICL capabilities are insufficient for standard SDFT to reach its full potential. We validate the two-stage pipeline proposed in Figure 3: (1) bootstrap ICL capabilities via off-policy distillation from a larger teacher, then (2) apply standard SDFT to recover general capabilities while strengthening task performance.

Setup. In Stage 1, Qwen2.5-7B-Instruct serves as teacher: both teacher and student are conditioned on the task prompt and an expert demonstration, and the 3B student minimizes KL divergence against the 7B teacher's output distribution. This off-policy step activates the 3B model's ICL capabilities but degrades

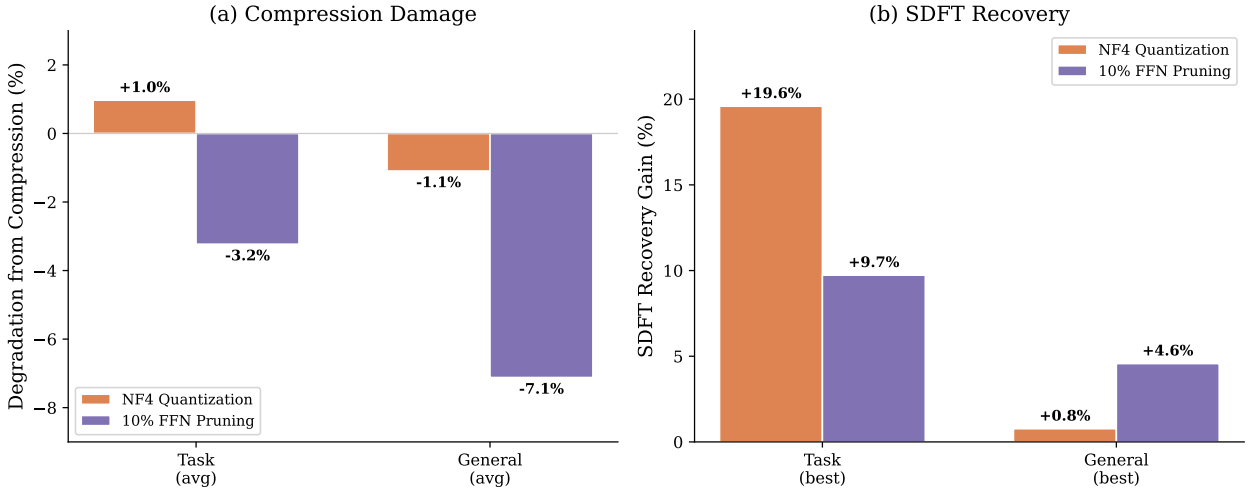

Figure 6: Cross-compression comparison. (a) Compression damage profile. (b) SDFT recovery gains. Quantization yields larger task improvements with full general preservation; pruning presents a harder recovery problem but SDFT still restores the majority of lost capabilities.

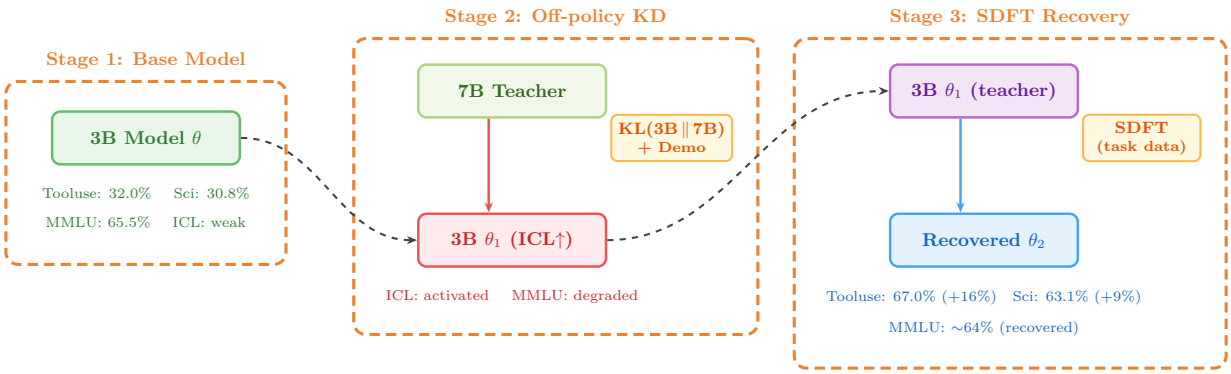

Figure 7: Two-stage small-model pipeline. Stage 1: base 3B model θ with weak ICL. Stage 2: off-policy distillation from 7B teacher activates ICL but degrades general capabilities. Stage 3: on-policy SDFT recovers general capabilities while strengthening task performance, completing the "degradation → recovery" loop.

general knowledge. In Stage 2, standard SDFT recovers general capabilities via on-policy self-distillation — the same "degradation → recovery" loop validated in Sections 5.1 and 5.2.

Results. Table 5 validates the two-stage pipeline. On Tooluse, the two-stage approach reaches 67.01% — a +16.49% improvement over direct SDFT (50.52%) and +35.05% over the base model. On Science, it reaches 63.12%, +8.88% above direct SDFT. These gains confirm that bootstrapping ICL via off-policy distillation unlocks the full potential of subsequent on-policy SDFT.

Crucially, the SDFT recovery step in Stage 2 fulfills its theoretical role: despite the off-policy distillation degrading MMLU, the final two-stage models preserve general capabilities within 1.5% of the base model (63.83% vs 65.47% on Tooluse; 64.02% vs 65.47% on Science). The MMLU gap between two-stage and direct SDFT is less than 1.2% on both tasks, confirming that on-policy self-distillation recovers the general capabilities lost during off-policy training. This validates the "degradation → SDFT recovery" loop proposed in Section 3 as a general mechanism that extends beyond compression to any form of capability loss.

Table 5: Two-stage distillation on Qwen2.5-3B-Instruct. Stage 1: off-policy distillation from 7B teacher bootstraps ICL. Stage 2: on-policy SDFT recovers general capabilities. The two-stage pipeline achieves +16.49% (Tooluse) and +8.88% (Science) over direct SDFT, with MMLU within 1.5% of the base model.

	Tooluse		Science	
Config	Task Acc	MMLU	Task Acc	MMLU
Base (θ)	31.96%	65.47%	30.77%	65.47%
Direct SDFT	50.52%	64.95%	54.24%	64.31%
Two-stage (ours)	**67.01%**	63.83%	**63.12%**	64.02%
Δ vs Direct SDFT	+16.49%	−1.12%	+8.88%	−0.29%

Table 6: Multi-stage pipeline: Science expert → SFT Tooluse (forgetting) → Recovery SDFT. CKA is computed at Layer 35 (Science eval, 507 samples) against the original Science expert. All four recovery configurations restore CKA toward the expert while recovering Science accuracy, confirming the manifold realignment mechanism proposed in Section 4.

Pipeline Stage	Tooluse	Science	CKA(Sci)	CKA(Tool)
Science expert	19.59%	59.96%	1.0000	1.0000
+ SFT Tooluse	64.95%	37.87%	0.9767	0.9388
+ Recovery (t=base)	65.98%	61.54%	0.9909	0.9425
+ Recovery (t=expert)	57.73%	65.48%	0.9877	0.9408

5.4 Manifold Alignment Validation

Section 4 posits that self-distillation recovers performance by realigning the student's high-dimensional manifold with the teacher's, and derives CKA as a rotation- and scale-invariant metric for quantifying this alignment. We now validate this theoretical framework empirically by testing two falsifiable predictions: (1) recovery SDFT should increase CKA between the recovered model and the pre-degradation expert, reversing the drift induced by intermediate fine-tuning; and (2) the magnitude of CKA misalignment should predict the severity of capability loss, establishing CKA as a diagnostic tool for forgetting.

Setup. We compute linear CKA following the procedure in Section 4. For each evaluation sample, we extract the d-dimensional activation vector from a given layer, forming the activation matrix $H \in \mathbb{R}^{n \times d}$ where n is the number of evaluation samples. We analyze the last hidden layer (Layer 35) of Qwen2.5-3B-Instruct (36 transformer layers). Activation matrices are centered and scaled (zero-mean, unit-variance per dimension) before computing kernel matrices.

CKA recovery in multi-stage pipelines. To test whether SDFT recovery restores manifold alignment, we construct the three-stage pipeline from Section 5.1: (1) train an SDFT expert on Science, (2) apply standard SFT on Tooluse (inducing forgetting of Science), and (3) apply recovery SDFT to restore Science capabilities.

Table 6 presents the central result. Across all four recovery configurations, SDFT increases CKA between the model and the original Science expert — without exception. This directly validates the theoretical prediction in Section 4 that self-distillation acts as a manifold realignment mechanism, not merely a behavioral correction at the output level.

The complete accuracy data reveals further insights. SFT on Tooluse drops Science by −22.09%, inducing severe forgetting. Yet recovery with t=base restores Science from 37.87% to 61.54% (+23.67%), nearly matching the original expert (59.96%), while preserving Tooluse at 65.98%.

Teacher choice introduces a diagnostic trade-off. The choice of teacher produces a trade-off visible in both accuracy and CKA: t=base produces higher CKA recovery (ΔCKA +0.014) with accuracy restored to original levels, while t=expert produces lower CKA recovery (ΔCKA +0.011) but pushes accuracy beyond

Table 7: CKA recovery magnitude (Layer 35, Science-first pipeline). In both recovery configurations, SDFT increases CKA toward the pre-degradation Science expert ($\Delta\text{CKA} > 0$), confirming manifold realignment.

Teacher	CKA(SFT)	CKA(Recovered)	ΔCKA
t=base	0.9767	0.9909	+0.0142
t=expert	0.9767	0.9877	+0.0110

the original expert (+5.52%). The expert teacher overshoots the original manifold geometry to achieve higher task accuracy, while the base teacher acts as a regularizer that faithfully restores the pre-degradation representation structure.

Table 7 quantifies this recovery magnitude: ΔCKA is positive without exception.

CKA misalignment predicts forgetting severity. A key prediction of our theoretical framework is that greater manifold misalignment should correspond to more severe performance degradation. Figure 8 confirms this: SFT Tooluse produces a CKA misalignment of 0.023 from the Science expert, corresponding to −22.09% Science accuracy drop, while recovery SDFT reduces this misalignment to 0.009 (t=base) and 0.012 (t=expert), restoring accuracy accordingly. This establishes last-layer CKA as a quantitative predictor of forgetting severity, providing empirical grounding for the CKA metric derived in Section 4.

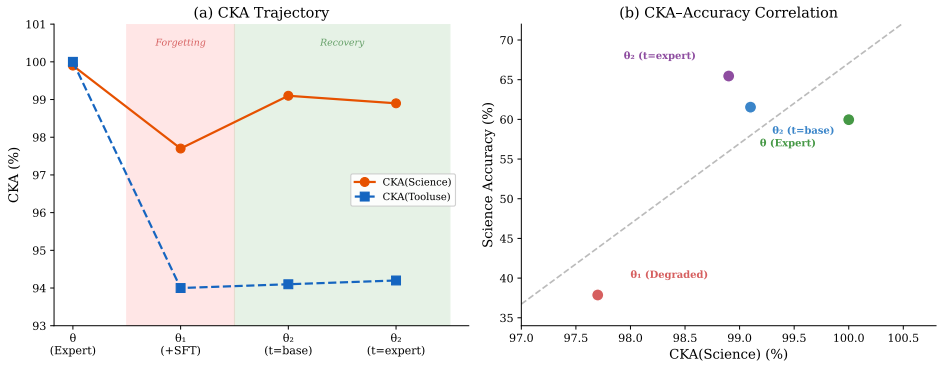

Figure 8: CKA misalignment at the last hidden layer (Layer 35) vs. accuracy change across all pipeline stages. Larger CKA misalignment from the expert corresponds to more severe capability loss. Recovery SDFT reduces misalignment while restoring accuracy, validating CKA as a diagnostic metric for forgetting severity as proposed in Section 4.

6 Discussion & Future Work

6.1 Discussion

Geometric Structure vs. Output Distribution. Our findings suggest that matching output distributions (logits) is effective, but manifold alignment provides a more fundamental explanation for performance recovery. The strong correlation between CKA scores and task performance indicates that aligning the internal geometric structure is a more fundamental mechanism. This supports the view that LLM capabilities are encoded in the topology of hidden representations rather than solely in output probabilities.

Dependency on Teacher Quality. While SDFT effectively recovers performance, it inherently relies on the quality of the teacher model. If the teacher's manifold itself is suboptimal, the student will align to this suboptimal structure. This highlights the importance of selecting a robust teacher or employing ensemble teachers to define a more reliable reference manifold for alignment.

Correlation vs. Causation. We observe a strong empirical correlation between manifold alignment and performance recovery. While our theoretical framework posits a causal link, we acknowledge that CKA measures structural similarity rather than direct functional capability. Future work should explore whether maximizing CKA directly as a loss function yields further improvements, which would provide interventional evidence to strengthen the causal relationship.

6.2 Future Work

Quantifying the Geometry-Performance Relationship. While our experiments establish a strong correlation between manifold alignment (CKA) and performance recovery, the precise quantitative mapping remains unexplored. For instance, a given percentage increase in CKA does not necessarily translate to a proportional gain in task accuracy, suggesting a non-linear or saturating relationship. Future work should aim to formulate a predictive theory that links geometric alignment metrics to functional performance bounds. Establishing such a relationship would allow CKA to serve as a proxy metric for early stopping or hyperparameter tuning, eliminating the need for expensive downstream evaluations during training.

7 Conclusion

In this work, we have addressed the critical challenge of performance degradation in LLMs caused by factors such as catastrophic forgetting during Supervised Fine-Tuning (SFT), quantization, and pruning. We have provided both a practical framework for LLM performance recovery and a rigorous theoretical explanation for its effectiveness. By shifting the focus from output distributions to internal geometric structures, we offer new insights into the internal mechanisms of self-distillation. We hope this research inspires further exploration of manifold-based analysis in deep learning, ultimately leading to more robust, interpretable, and efficient language models.

Broader Impact Statement

This work proposes a recovery mechanism for LLM performance degradation. While the framework itself is a general-purpose tool for improving model quality, we note that enhanced LLM capabilities could amplify both beneficial and harmful applications. We encourage practitioners to apply responsible deployment practices when using recovered models in production systems.

References

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385, 2021.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized language models. *Advances in Neural Information Processing Systems*, 36, 2023.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *International Conference on Learning Representations*, 2023.

Tommaso Furlanello, Zachary C. Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *Proceedings of the 35th International Conference on Machine Learning (ICML 2018)*, pp. 1607–1616, 2018.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. *Algorithmic Learning Theory*, pp. 63–77, 2005.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pp. 3519–3529. PMLR, 2019.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. LLM-Pruner: On the structural pruning of large language models. *Advances in Neural Information Processing Systems*, 36, 2024.

Team Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Sober, Koray Kavukcuoglu, and Raia Hadsell. Progressive neural networks. In *arXiv preprint arXiv:1606.04671*, 2016.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.