From Adaptation to Deployment: Principle-Driven Techniques for Large-scale Multimodal Models

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Massimo Bini
aus
Latisana/Italien

Tübingen
2025

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlinchen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	16.12.2025
Dekan:	Prof. Dr. Thilo Stehle
Berichterstatterin:	Prof. Dr. Zeynep Akata
Berichterstatter:	Prof. Dr. Niki Kilbertus

UNIVERSITY OF TÜBINGEN

Department of Computer Science

From Adaptation to Deployment: Principle-Driven Techniques for Large-scale Multimodal Models

Massimo Bini

Master of Science in Mathematical Engineering

Advisors: Zeynep Akata
Full Professor, Technical University of Munich
Anna Khoreva
Doctor Researcher

Examination Committee

Chair: Peter Gehler
Full Professor, University of Tübingen

Members: Zeynep Akata
Full Professor, Technical University of Munich
Niki Kilbertus
Full Professor, Technical University of Munich
Hilde Kuehne
Full Professor, University of Tübingen

Doctorate in Computer Science

University of Tübingen
November, 2025

Abstract

The rise of massive foundation models pretrained on large-scale datasets has enabled unprecedented generalization, but their immense size presents critical deployment challenges, demanding substantial memory and computational resources that make specialized adaptation and deployment in resource-constrained environments prohibitively expensive. This thesis addresses these challenges through three interconnected contributions that progressively enable efficient model adaptation and deployment.

Our first contribution, *ETHER*, challenges prevailing assumptions about parameter-efficient fine-tuning (PEFT). While LoRA's simplicity has democratized finetuning, parallel research has investigated sophisticated geometric constraints, such as preserving hyperspherical energy (HE) via orthogonal transformations. *ETHER*, initially motivated by HE preservation through hyperplane reflections, reveals through empirical analysis that adaptation robustness actually derives from bounding the Frobenius norm of weight updates rather than maintaining HE. This insight suggests a paradigm shift: effective finetuning depends on constraining weight deviation, not on geometric energy preservation. However, *ETHER*'s fixed rank, predetermined boundaries, and computational overhead from matrix multiplications limit its practical applicability.

Building directly on this insight, our second contribution, DeLoRA, translates the Frobenius-bounded robustness principle into a practical, LoRA-like method. DeLoRA operates with arbitrary rank, replaces matrix multiplications with additions, and implicitly maintains a Frobenius boundary by decoupling magnitude and angular learning. This decoupling enables stable training at larger learning rates, and effectively prevents catastrophic overwriting, while its flexible design supports diverse applications.

Our third contribution addresses the remaining bottleneck: the foundation model itself. While parameter-efficient adapters dramatically reduce specialization costs, the base model's size remains the primary deployment constraint. Motivated by the observation that massive generalist models contain substantial redundancy when deployed for specific operations, we demonstrate how combining parameter-efficient adaptation with knowledge distillation enables practical deployment of specialist models. Through MemLoRA, deployed in an LLM-powered memory system, we show how lightweight adapters can

function as specialist experts—one for each memory operation—when enhanced through knowledge distillation and paired with a single compact student model. This paradigm achieves equivalent performance with dramatically reduced memory footprint, faster inference, and minimal deployment cost.

Together, these contributions establish a cohesive framework for making foundation models accessible in resource-constrained environments—from discovering robust adaptation principles, to designing efficient methods that embody these principles, to ultimately demonstrating practical deployment through PEFT-enhanced distillation.

Contents

List of Figures

List of Tables

Listings

1

Introduction

Pretrained models have fundamentally transformed the landscape of artificial intelligence, establishing themselves as the cornerstone of modern machine learning systems [42]. By leveraging large-scale unsupervised or self-supervised learning on massive datasets, these models capture rich, transferable representations that can be effectively adapted to a wide variety of downstream tasks with minimal task-specific training. This paradigm shift—from training models from scratch to adapting pretrained foundations—has yielded unprecedented improvements across domains, from natural language processing [4, 57] to computer vision [28, 146], and has become the de facto standard in the development of contemporary state-of-the-art AI systems.

The success of pretrained models is rooted in two fundamental principles that have long been central to machine learning: *transfer learning* and *representation learning*. Transfer learning enables models to leverage knowledge acquired from one or more source tasks to improve performance on target tasks, even when labeled data is scarce [112, 131]. Representation learning focuses on discovering meaningful features from raw data that can generalize across different contexts [8, 154]. Together, these principles empower pretrained models to overcome the traditional bottleneck of data hunger in deep learning, enabling effective learning with limited labeled examples by capitalizing on the vast amounts of unlabeled data available in the real world [47].

In this thesis, we address critical challenges in the deployment of large-scale foundation models, with particular emphasis on robustness and efficiency in resource-constrained environments. Driven by considerations on adaptations that preserve models' intrinsic properties [93, 115], we identify and provide empirical evidence on novel key factors for adaptation robustness. Building upon these insights, we develop novel adaptation methods guided by robustness and deployability objectives. Finally, motivated by observations of redundancies in large-scale models [2, 73] and of specialization through adaptation [43], we demonstrate how these same tools can enable the replacement of massive models with compact task-specific experts via knowledge distillation. This demonstrates the powerfulness of these methods, beyond simple adaptation tools.

The remainder of this chapter first provides a historical overview, tracing the evolution

from early transfer learning approaches to modern foundation models. Subsequently, we analyze the theoretical foundations of pretraining and transfer learning, elucidating how these concepts inform current methodologies. Following this, we introduce the challenges of adapting and deploying these large-scale foundation models, outlining current solutions and opportunities, with special focus on parameter-efficient finetuning and knowledge distillation. We conclude the chapter by clearly outlining the primary contributions made in this thesis, setting the stage for detailed discussions presented in subsequent chapters.

1.1 From Transfer Learning to Pretrained Models

The concept of transfer learning has deep roots in machine learning [135], motivated by the observation that humans can leverage previously learned knowledge to solve new problems with remarkable efficiency [112]. Rather than learning each new task in isolation from scratch, intelligent systems can transfer relevant knowledge across related tasks, dramatically reducing the amount of task-specific training required. This capability is particularly valuable in scenarios where labeled data is expensive or difficult to obtain [112], which encompasses the majority of real-world applications.

Early explorations of transfer learning focused on two primary approaches: *feature transfer* and *parameter transfer*. Feature transfer methods aimed to learn effective representations from source tasks that could be reused in target tasks, while parameter transfer approaches sought to share model parameters or prior distributions across tasks [112]. These foundational ideas laid the groundwork for modern pretrained models, establishing the key insight that knowledge captured during training on one task can be beneficial for learning another.

The deep learning revolution brought new urgency and opportunities to transfer learning research. As neural networks grew deeper and more powerful, they also became increasingly data-hungry, requiring massive labeled datasets to achieve good performance [47]. This created a fundamental tension: while deep networks had the capacity to learn rich representations [47, 72], obtaining sufficient labeled data for every task of interest was often infeasible. Pretraining emerged as an elegant solution to this challenge, enabling the separation of representation learning (performed once on large-scale unlabeled or weakly-labeled data) from task-specific adaptation (performed with limited labeled examples) [16, 117, 154].

The Rise of Self-Supervised Pretraining. A crucial turning point in the evolution of pretrained models was the adoption of *self-supervised learning* as the primary pretraining paradigm [25, 117]. Unlike supervised learning, which requires expensive manual annotations, self-supervised learning leverages the inherent structure within data itself to create supervision signals. This breakthrough enabled the utilization of virtually unlimited

unlabeled data, as the model could generate its own training targets from the input data without human intervention.

In natural language processing, this manifested through objectives such as predicting masked words in sentences (masked language modeling) [25] or predicting the next word in a sequence (autoregressive language modeling) [117]. These seemingly simple tasks, when performed at scale on massive text corpora, enabled models to capture rich linguistic knowledge including syntax, semantics, and even factual information about the world. The introduction of the Transformer architecture [139] further accelerated this progress, providing an efficient and scalable foundation that could effectively process long-range dependencies in sequential data.

The success of models like GPT (Generative Pretrained Transformer) [117] and BERT (Bidirectional Encoder Representations from Transformers) [25] in 2018 marked a pivotal moment [42]. These models demonstrated that self-supervised pretraining on large text corpora, followed by finetuning on specific tasks, could achieve state-of-the-art results across a wide range of NLP benchmarks [140]. This paradigm quickly spread beyond language to other domains, with pretrained models showing impressive results in computer vision [16, 28], speech recognition [5], and multimodal learning [90, 101, 116, 123].

The Scaling Era and Foundation Models. Following the initial success of self-supervised pretrained models, the field entered an era characterized by aggressive scaling of model size and training data. Models grew from millions of parameters to billions [13, 118, 136] and eventually trillions [6, 23, 30, 110], while training datasets expanded from gigabytes to terabytes of data [13, 30, 61]. This scaling revealed emergent capabilities [144]: as models grew larger, they began to exhibit qualitatively improved behaviors, including few-shot and even zero-shot learning abilities—the capacity to perform tasks with minimal or no task-specific examples [13].

GPT-3 [13], with 175 billion parameters, exemplified this trend, demonstrating that sufficiently large language models could solve novel tasks simply through careful prompting, without any parameter updates. This gave rise to the concept of *foundation models*: large-scale models trained on broad data that can be adapted to a wide range of downstream applications [12]. These models have become the foundation upon which numerous applications are built, fundamentally changing how AI systems are developed and deployed.

However, this scaling trend has introduced significant challenges. Training and deploying models with billions of parameters requires substantial computational resources, specialized hardware, and extensive engineering effort [12, 13]. Even finetuning such models on specific tasks can be prohibitively expensive in terms of computation, memory, and time [51, 53]. Furthermore, maintaining separate fine-tuned copies of large models for different tasks is impractical due to storage constraints. These challenges have created an urgent need for efficient adaptation methods.

1.2 Efficient Adaptation of Foundation models

As mentioned above, despite their impressive capabilities, pretrained models present significant practical challenges when adapting them to downstream tasks. Traditional *full finetuning*—updating all model parameters on task-specific data—has several critical limitations: (i) *Computational Cost* - finetuning billion-parameter models requires substantial computational resources, often necessitating multiple high-end GPUs and extended training times; (ii) *Memory Requirements* - storing optimizer states and gradients for all parameters during training can require several times more memory than the model itself [65], making finetuning infeasible on resource-constrained devices; (iii) *Storage Overhead* - maintaining separate fine-tuned copies of large models for different tasks leads to enormous storage requirements, particularly in multi-task or multi-tenant scenarios; (iv) *Catastrophic Forgetting* - aggressive finetuning can cause models to forget valuable knowledge acquired during pretraining, potentially degrading performance on related tasks [64, 67, 75, 142]; and (v) *Deployment Complexity* - serving multiple task-specific model copies introduces operational complexity and increases inference costs.

These limitations have motivated intensive research into *Parameter-Efficient Fine-Tuning (PEFT)* methods—techniques that adapt pretrained models to new tasks by updating only a small subset of parameters or introducing a small number of additional trainable parameters [53, 78, 81]. The key insight is that the full capacity of large pretrained models may not be necessary for adaptation; instead, small, targeted modifications can often achieve comparable or even superior performance to full finetuning [9, 92, 94] while dramatically reducing computational and memory costs.

PEFT approaches offer several compelling advantages: (i) *Reduced Computational Cost* – training time and compute requirements are substantially reduced, making adaptation feasible even with limited resources; (ii) *Lower Memory Footprint* – storing optimizer states for only a subset of parameters significantly reduces memory consumption during training, enabling adaptation on consumer-grade hardware [24]; (iii) *Efficient Multi-Task Serving* – by sharing a single frozen base model across tasks and maintaining only small task-specific modules, storage and serving costs can be dramatically reduced [55]; and (iv) *Refined Adaptation* – different PEFT methods bring different forms of regularization and properties, explicitly or implicitly, from preserving pretraining knowledge, to potentially improving performance on small task-specific datasets [9, 115].

Parameter-Efficient Finetuning Methods. Motivated by different efficiency considerations and architectural insights, the field of PEFT has evolved along several distinct paradigms. *Adapter-based methods* [51, 113] insert small trainable modules between frozen transformer layers, with subsequent work exploring bottleneck architectures [125]. *Prompt-tuning approaches* prepend learnable soft prompts to the input [78], with prefix-tuning [81] extending this to key-value pairs across layers. *Low-rank adaptation methods*, pioneered by LoRA [53], inject trainable low-rank decomposition matrices into weight updates,

spawning numerous variants including AdaLoRA [160] with adaptive rank allocation, QLoRA [24] combining quantization with LoRA, and DoRA [92] decomposing weights into magnitude and direction components. *Orthogonal fine-tuning methods* employ orthogonal transformation not to alter the hyperspherical energy among column vectors of the weight matrices [93, 115]. Beyond these families, *selective fine-tuning* approaches identify and update specific parameter subsets [41], while *unified frameworks* [45] demonstrate that many PEFT methods can be viewed through a common lens. Recent work has also explored composition and scaling: combining multiple PEFT modules [20, 121], dynamic selection mechanisms [34, 157], and analysis of how PEFT methods scale with model size and task complexity [84].

Among these approaches, methods that introduce lightweight trainable modules are of particular interest, as these modules can be merged with pretrained weights at inference time, thereby eliminating additional latency—a critical advantage for massive models requiring multiple forward passes, such as LLM-based systems. Within this PEFT family, trained adapters can be seamlessly merged and unmerged from the base weights, enabling straightforward deployment in multi-task systems that swap adapters based on task requirements.

This thesis focuses specifically on this family of PEFT methods, as they offer the greatest versatility for effective deployment. In particular, we introduce novel PEFT methods that: (i) originate from hyperspherical energy considerations, (ii) identify an alternative optimization driver that effectively prevents pretrained weight overwriting, introducing a new family of methods, and (iii) culminate in a comprehensive LoRA variant that integrates these characteristics with LoRA's flexible rank structure and enhanced properties, ultimately combining the advantages of two PEFT families.

1.3 From Adaptation to Deployment

Beyond adaptation efficiency, the deployment of large foundation models presents another critical challenge: even with parameter-efficient adapters, the size of the underlying model can be the limiting factor for effective deployment. This issue becomes increasingly critical as models continue to scale, imposing substantial memory requirements and computational costs during inference that make deployment impractical in resource-constrained environments or when serving multiple specialized models concurrently.

At the same time, recent analysis of foundation models' structural properties reveal a striking inefficiency: these models appear vastly underutilized, with their intrinsic dimensionality often orders of magnitude lower than their full parameter space [2]. This phenomenon stems from foundation models being trained as generalists on diverse, large-scale, corpora spanning multiple domains, tasks, and often languages [12]. While this overparametrization contributes to their broad generalization capabilities [2], it also implies that when deployed for specific applications, much of their learned capacity remains unnecessary for the task at hand [73].

Deployment via Knowledge Distillation. To address these deployment constraints while maintaining model capability, Knowledge Distillation (KD) has emerged as a powerful complementary technique [32, 50, 91, 97, 158]. By training small "student" models to mimic the behavior of large "teacher" models, knowledge distillation can produce compact models that retain much of the teacher's performance while dramatically reducing computational and memory requirements [50, 128]. This approach has become standard practice in modern LLM deployments, where model providers routinely publish multiple size variants of their models, with the smaller variants typically distilled from their larger counterparts [30]. This enables users to select the appropriate model size based on their specific deployment constraints and performance requirements.

In this thesis we synthesize these insights by demonstrating how parameter-efficient fine-tuning can be powered by knowledge distillation to build task-specific expert models. Specifically, we test this approach in LLM-powered memory systems, where large generalist models are asked to perform multiple specific tasks. Rather than fine-tuning entire models, we train parameter-efficient adapter "experts"—one per task—where each adapter is enhanced through knowledge distillation from a larger teacher model to capture task-specific expertise. This approach enables a practical deployment paradigm: the large foundation model can be entirely replaced by a single compact student model paired with a library of lightweight adapter experts, one for each specialized task. This dramatically reduces deployment costs while maintaining task-specific performance, as only the small student model and the relevant adapter need to be loaded for any given task.

1.4 Research Questions and Thesis Contributions

The development of this thesis was guided by several research questions that shaped its core contributions. OFT [115] approached model finetuning as an adaptation process that preserves hyperspherical energy, operating under the assumption that such preservation is critical for preventing knowledge deterioration during adaptation.

> *But, is this hyperspherical energy retention really key for finetuning in a knowledge-preserving way?*

In our first contribution, we empirically test this assumption by conducting finetuning experiments with a non-orthogonal variant of the proposed transformation—thereby explicitly violating hyperspherical energy preservation. Our results shows that this violation does not significantly impact performance, challenging the original hypothesis. Building on this insight, and motivated by the broader goal of developing finetuning methods that preserve costly pretraining knowledge, we introduce *ETHER* and its enhanced variant

ETHER+. These methods, based on hyperplane reflections, achieve comparable performance to existing approaches while requiring only a fraction of the computational cost and exhibiting superior non-deteriorating properties. Critically, we demonstrate that these desirable properties stem not from hyperspherical energy preservation, but rather from the transformation's implicit bound on Frobenius norm distance, which constrains the deviation from the pretrained model. We conduct a thorough investigation of this property, with particular emphasis on Stable Diffusion, enabling both quantitative analysis and qualitative visualization of the non-deteriorating effect relative to hyperspherical-preserving methods. Finally, we introduce a custom block-parallelization algorithm that enables more efficient matrix multiplication for our proposed methods.

Despite demonstrating strong robustness and desirable non-deteriorating properties, *ETHER* and *ETHER*+ exhibit several inherent limitations. In our second contribution, we begin by rigorously analyzing and characterizing these shortcomings: (i) although the transformations themselves are full-rank, we mathematically derive that the resulting weight updates are necessarily low-rank—a constraint we empirically show limits model expressivity; (ii) the fixed low-Frobenius-distance bound, while beneficial for stability, is potentially suboptimal as it cannot adapt to task-specific requirements; and (iii) despite substantial performance gains achieved through our block-parallelization algorithm, matrix multiplications remain computationally more expensive than LoRA-based approaches. These observations naturally led to a fundamental question:

> *Is it possible to combine the robustness of ETHER with the flexibility and efficiency of LoRA?*

Motivated by the insight that Frobenius norm constraints are essential for preventing pretrained knowledge deterioration, our second contribution establishes a principled framework for bridging *ETHER* and LoRA. We demonstrate two complementary pathways: first, how to incorporate *ETHER*'s robustness properties into LoRA; and second, how to integrate LoRA's flexibility into *ETHER*'s formulation. Through these approaches, we derive *DeLoRA*, a novel PEFT method that enhances the robustness properties of *ETHER* and *ETHER*+ while supporting arbitrary Frobenius boundaries and offering LoRA's flexibility: arbitrary rank selection and efficient computation through matrix additions rather than multiplications. Remarkably, when the boundary term is treated as a learnable parameter, DeLoRA maintains its non-deteriorating properties. This behavior reveals that DeLoRA effectively decouples angular learning from magnitude learning—a separation that preserves pretrained knowledge even under aggressive learning rate regimes. The method's flexibility positions it as a promising candidate for diverse applications, while its inherent robustness to model deterioration makes it particularly well-suited for deployment in systems requiring continual, on-the-fly adaptation.

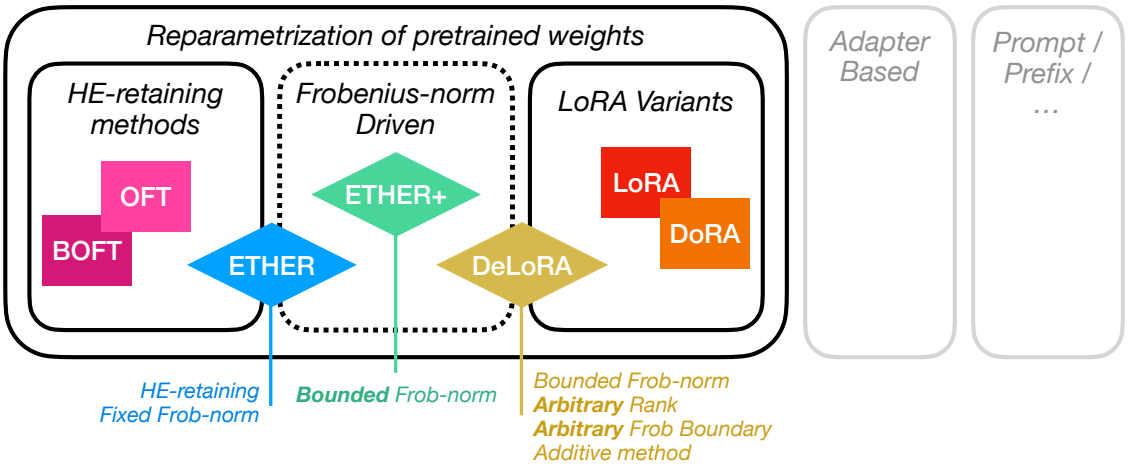

Figure 1.1: **Overview of PEFT innovations.** In this thesis, we introduce several reparametrization-based PEFT methods (rhombuses in figure). In particular, we begin our study on Hyperspherical Energy (HE) developments, driven by knowledge preservation hypothesis [*ETHER*]. From here, we challenge HE claims, and find as key driver Frobenius distance bounds [*ETHER+*] - introducing a new family of methods. Finally, we tackle all limitations of previous methods, and bridge the gap between Frobenius-norm-driven and LoRA-based methods [DeLoRA] - introducing knowledge preservation properties to LoRA, or, alternatively, introducing new deployment-friendly properties to *ETHER+*.

While the preceding contributions address the efficiency and robustness of model adaptation, they operate under the assumption that the large foundation model itself remains deployed. However, large foundation models impose significant deployment costs—requiring substantial memory, computational resources, and inference latency—costs that become particularly inefficient when deploying models for specialized tasks that don't require the full breadth of the foundation model's capabilities. Given the proven effectiveness of parameter-efficient adapters for specialization and knowledge distillation for model compression, a natural question emerges:

> *Can we power parameter-efficient adaptation with knowledge distillation to enable practical deployment of task-specific expert models?*

In our third contribution, we demonstrate such a synthesis. Rather than fine-tuning parameter-efficient adapters on top of large foundation models, we show how adapters can be enhanced through knowledge distillation to function as specialist experts. Specifically, we train parameter-efficient adapter "experts"—one per task—where each adapter captures task-specific knowledge distilled from a larger teacher model. This architecture enables a compelling deployment paradigm: the large foundation model can be entirely replaced by a single compact student model paired with a library of lightweight adapter experts. For any given task, only the small student model and its corresponding adapter need to be loaded, dramatically reducing deployment costs while preserving task-specific performance.

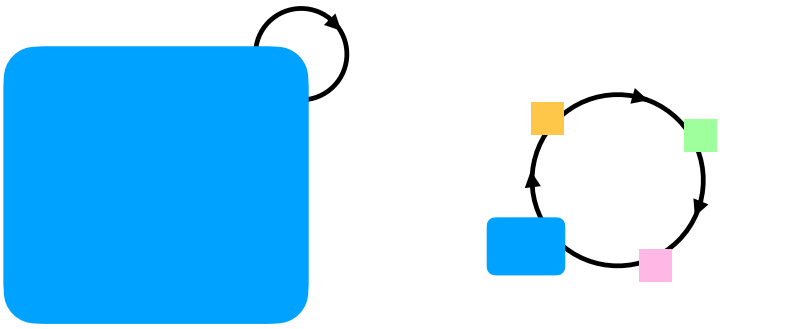

Figure 1.2: **Overview of Task-specific Efficient Deployment.** In this thesis, we do not limit PEFT methods to adaptation tools for large-scale models, but further demonstrate how they can drive more efficient deployments in multi-task systems. With MemLoRA, we demonstrate the powerfulness of this framework in LLM-powered memory systems. By equipping small language models (SLM) with task-specific adapter-experts trained via knowledge distillation (KD) we achieve performance comparable to 60x larger LLMs, enabling on-device deployment with minimal compromises.

Collectively, these contributions address a fundamental challenge: enabling efficient deployment and adaptation of large-scale foundation models. Our investigation consistently proceeds toward this central objective. First, with the development of *ETHER* and *ETHER+*, we establish the foundations for robust adaptation by identifying Frobenius norm constraints as a key principle for non-deteriorating fine-tuning—a critical insight for deployment scenarios where model reliability cannot be compromised. Second, with DeLoRA, we enhance these methods and bridge the gap with LoRA-based methods, enhancing LoRA with knowledge-preserving properties. Finally, we address the remaining deployment bottleneck: the foundation model itself. Motivated by the observation that massive generalist models contain substantial redundancy when deployed for specific tasks, we demonstrate how parameter-efficient adaptation can be powered by knowledge distillation to enable a transformative deployment paradigm. Rather than deploying large foundation models with task-specific adapters, we show how lightweight adapters can function as specialist experts when paired with a single compact student model—replacing the entire large model while preserving task-specific performance.

1.5 Thesis Outline

The structure of the remainder of the thesis is briefly described below. Each chapter corresponds to an individual research paper. All works in the thesis are solo-first-authored.

- **Chapter 2: *ETHER* - Efficient Finetuning of Large-Scale Models with Hyperplane Reflections** presents our work on bounded parameter-efficient finetuning through a relaxation of Householder transformations, published as a conference paper in ICML 2024. By framing adaptation as hyperplane reflections with constant distance

to the identity, *ETHER* achieves remarkable learning rate robustness and parameter efficiency across image generation and language understanding tasks.

- **Chapter 3: Decoupling Angles and Strength in Low-Rank Adaptation** details our method for decoupled low-rank adaptation that normalizes and scales LoRA matrices, published as a conference paper in ICLR 2025, and soon-to-be-released in HuggingFace's peft package [104] which will allow broad and easy deployment. By separating angular learning from adaptation strength, DeLoRA achieves both the robustness of bounded methods and the expressiveness of high-rank adaptation, demonstrating superior stability and performance across diverse benchmarks.
- **Chapter 4: MemLoRA - Distilling Expert Adapters for On-Device Memory Systems** explores MemLoRA, our efficient deployment of KD-trained expert adapters in LLM and VLM -powered memory systems, currently under review. This work bridges the gap between efficient adaptation and practical deployment, enabling on-device deployment with significant improvements in inference speed and resource utilization.
- **Chapter 5: Conclusions and Discussion** summarizes the contributions of this thesis, and outlines promising directions for future research in parameter-efficient adaptation and efficient deployment of foundation models.

Together, this thesis provides a comprehensive treatment of critical challenges in foundation model deployment, from achieving robust finetuning through bounded transformations, to enabling flexible high-rank updates with principled decoupling, to deploying adapted models efficiently on resource-constrained devices. By addressing the complete pipeline from adaptation to deployment, this work advances both our theoretical understanding and provides practical solutions for making large-scale AI systems accessible, robust, and deployable at scale.

2

ETHER: Efficient Finetuning of Large-Scale Models with Hyperplane Reflections

Parameter-efficient finetuning (PEFT) has become ubiquitous to adapt foundation models to downstream task requirements while retaining their generalization ability. However, the amount of additionally introduced parameters and compute for successful adaptation and hyperparameter searches can explode quickly, especially when deployed at scale to serve numerous individual requests. To ensure effective, parameter-efficient, and hyperparameter-robust adaptation, we propose the *ETHER* transformation family, which performs *E*fficient fine*T*uning via *H*yp*E*rplane *R*eflections. By design, *ETHER* transformations require *a minimal number of parameters*, are *less likely to deteriorate model performance*, and exhibit *robustness to hyperparameter and learning rate choices*. In particular, we introduce *ETHER* and its relaxation *ETHER+*, which match or outperform existing PEFT methods with significantly fewer parameters (~10-100 times lower than LoRA or OFT) across multiple image synthesis and natural language tasks without *exhaustive hyperparameter tuning*. Finally, we investigate the recent emphasis on Hyperspherical Energy retention for adaptation and raise questions on its practical utility. The code is available at `https://github.com/mwbini/ether`.

2.1 Introduction

Recently, large-scale foundation models [12] have demonstrated impressive general-purpose capabilities across both generative and discriminative tasks [66, 110, 123, 136], showing extensive flexibility and strong performance when further adapted to different, more specialized tasks such as instruction following or controlled image synthesis [19, 126, 132, 159].

While impressive, these capabilities come with parameter counts increasing into the billions [110, 114, 137]. To allow for affordable and scalable model adaptation that can serve large and diverse client bases, various techniques have been introduced in the literature. They range from full finetuning [130, 156, 169] to just a few layers of the pretrained model [71], concatenating additional learning modules [51, 108, 113], and more recently to adapters on the network weights with lightweight learnable transformations [54, 70, 115, 138]. The latter have proven particularly effective, introducing no inference latency, fewer adaptation parameters, and strong performance.

Conceptually, these methods finetune on smaller datasets to adapt to downstream task and data requirements, without (1) compromising too much on the costly pretraining and (2) incurring concept and semantic drifts by catastrophically overwriting pretrained weights [38, 56, 64, 67, 75, 102, 105, 124, 126]. Treading the line for a suitable trade-off between adaptation and retention of the foundational model capabilities thus presents itself as a difficult task to tackle, often requiring costly tuning of hyperparameters such as learning rates. This problem is acknowledged explicitly in [15, 39, 82] aiming to preserve Euclidean weight distances between pretrained and finetuned models, and implicitly with approaches opting for both lower learning rates (at the cost of more tuning iterations) and inclusion of tuning parameters via summation [115].

In particular, [115] hints that a Euclidean distance measure likely fails to fully capture the preservation of the network's ability, suggesting instead Hyperspherical Energy (HE) as an alternative measure. The resulting objective uses orthogonal transformations (OFT) for multiplicative weight changes that control HE. Still, even OFT requires specific and restricted hyperparameter choices such as small learning rates and initialization from identity matrices to ensure sufficient knowledge preservation. In addition, while more robust and stable for finetuning in controllable generation settings compared to LoRA [115], OFT comes with a high computational overhead due to matrix multiplication and a large number of tuning parameters.

In this work, we propose **E**fficient fine**T**uning via **H**yp**E**rplane **R**eflections (*ETHER*) - a new family of weight transformations, efficient in parameter count while preserving model abilities and being robust in convergence and learning rate choices. By default, *ETHER* transformations frame the tuning process as a search for suitable hyperplanes, along which weight vectors can be reflected based on the orthogonal Householder transformation [52]. This keeps the distance to the transformation neutral element - the identity

matrix - constant by construction and improves training stability while reducing the chance of deteriorating model performance. In addition, being built from single vectors, Householder transformations allow for efficient block-parallel matrix multiplication with minimal performance trade-offs.

However, situations may arise where the hard distance restriction of *ETHER* can prove suboptimal (such as for subject-driven image generation, where finegrained subject-specific semantics need to be retained). As such, we augment the *ETHER* family with *ETHER+* - a relaxation on the default *ETHER* method. More precisely, *ETHER+* derives from the Householder transformation, but breaks the orthogonality and constant distance constraints, introducing multiple hyperplanes that can interact with a weight vector. As a result, *ETHER+* allows for more controlled and finegrained adaptation, while still having a bounded distance to the transformation neutral element, and retaining the *ETHER* benefits of high parameter-efficiency, training stability, and hyperparameter robustness.

Indeed, across subject-driven image generation, controlled image synthesis, natural language understanding and instruction tuning tasks, we find that *ETHER* and especially *ETHER+* match and outperform existing methods using only a few additional tuning parameters (e.g. $100\times$ less than OFT when finetuning Stable Diffusion for controlled image synthesis) - all while presenting stronger learning rate robustness compared to other methods and consequently requiring minimal hyperparameter tuning to achieve strong performance (c.f. Sec. 2.4). Finally, we also utilize our experimental benchmark findings to further investigate and question the recent emphasis on transformation orthogonality and hyperspherical energy (HE) retention (e.g. [115]), showing how non-orthogonal *ETHER+* can achieve strong performance while displaying increased HE.

2.2 Related Work

Parameter-Efficient Finetuning (PEFT). PEFT of pretrained models has seen different strategies evolve in the past years - starting from finetuning protocols and concatenation of learnable modules [41, 51, 78, 81, 113] to more recently reparametrization of network weights with efficient transformations [54, 70, 115, 138, 160]. The latter have shown convincing trade-offs between adaptation quality, additional parameters, and inference latency. LoRA [54] transforms network weights by adding the result of a learnable, low-rank matrix product. On top of LoRA, multiple variations have been proposed, s.a. QLora [24] with quantized weights, AdaLoRA [160] with dynamic rank adjustment, and VeRA [70] with low-rank frozen random projections and trainable vectors to reduce parameter counts. OFT [115] instead learns matrix multiplier with orthogonality constraints to retain hyperspherical energy. In our work, we use the same paradigm but introduce hyperplane reflections for better parameter efficiency and learning rate robustness.

Controlling Diffusion Generative Models. Diffusion-based generative models show strong compositional generation [63, 109, 114, 123, 127]. Among these, [33, 126] popularized personalized generation - teaching models to generate variations of user-provided samples. Based on DreamBooth [126], other works [99, 122, 167] followed. ControlNet [159] shows model controllability through external signals s.a. semantic and depth maps or face landmarks via extra layers at the cost of higher inference latency. [115] show controllability through direct finetuning with learnable matrix-multiplication transformations. Our work suggests an alternative, more robust and parameter-efficient approach through hyperplane reflections.

Instruction Tuning Language Models. Large Language Models (LLMs) have shown striking generalization across a wide range of tasks [110, 136, 164, 170]. However, the default training objective often does not exactly match downstream task requirements and intentions. To address this mismatch, Instruction Tuning [100, 132, 143, 164] finetunes LLMs using additional `(Instruction, Output)` pairs to explicitly align the model with human preferences. This enhances capabilities and controllability while avoiding costly retraining [74]. Recently, methods based on LoRA [54] have been proposed to efficiently achieve this control [17, 24, 70, 138, 153]. This work proposes a strong alternative with further parameter-efficiency and high learning rate robustness.

2.3 Method

We first discuss adapter-based PEFT in §2.3.1, before describing and motivating the use of hyperplane reflections in *ETHER* (§2.3.2). To encourage flexibility in trainable control and adaptation, we propose a simple, yet effective relaxation *ETHER+* in §2.3.3. Finally, §2.3.4 describes block-diagonal *ETHER* for improved computational efficiency.

2.3.1 Preliminaries

Parameter-Efficient Finetuning with Adapters. The most commonly deployed form of PEFT with an adapter is *Low-rank Adaptation* (*LoRA*, [54]). LoRA parametrizes a change of pretrained weights W as

$$(W + BA)^\intercal x + b$$

where BA is the matrix product of two low-rank matrices, i.e. for $W \in \mathbb{R}^{d\times f}$, $A \in \mathbb{R}^{d\times r}$ and $B \in \mathbb{R}^{r\times f}$. When rank $r << \min(d, f)$, this can bring down required tuning parameters significantly compared to full finetuning. In addition, BA can be absorbed into W during inference to avoid additional latency.

Orthogonal Finetuning (OFT). However, finetuning with LoRA can incur significant, potentially catastrophic weight changes. To ensure better preservation of pretrained model weights, [115] propose Orthogonal Finetuning (OFT). Based on the hypothesis that Hyperspherical Energy (HE) needs to be kept unaltered to preserve the original model abilities, OFT proposes the usage of multiplicative orthogonal transformations on the model weights. By retaining pairwise weight angles, HE can remain unaffected. However, to work in practice, [115] require the construction of the orthogonal matrix Q via a Cayley parametrization $Q = (I + S)(I - S)^{-1}$, where S is skew-symmetric. Notice that by using this parametrization, they limit the range of possible orthogonal matrices to those with determinant 1, missing orthogonal matrices with determinant equal to -1. As we show, this is relevant, as it excludes reflections, which motivate *ETHER*. To make OFT more parameter efficient, the orthogonal matrix $Q \in \mathbb{R}^{d\times d}$ is built in a block-diagonal fashion, made up of n smaller blocks Q^b of size $\frac{d}{n} \times \frac{d}{n}$. The final OFT transformation on the forward pass can then be described as

$$(Q^B W)^\intercal x + b$$

with block-diagonal Q^B. The trainable parameters are the n matrices $Q^b \in \mathbb{R}^{\frac{d}{n}\times\frac{d}{n}}$ that compose Q^B - more specifically the matrices R^b that build the skew-symmetric matrices $S^b = \frac{1}{2}(R^b - (R^b)^\intercal)$ for Q^b. For finetuning, the R^b are initialized as zero, such that $Q^B|_0 = I$ and consequently $Q^B|_0 W = W$ at the beginning of finetuning.

2.3.2 *ETHER*: Finetuning with Hyperplane Reflections

Fundamentally, *ETHER* (**E**fficient fine**T**uning via **H**yp**E**rplane **R**eflections) sets up weight transformations as hyperplane reflections. These reflections can be obtained via the

Householder transformation matrix $H \in \mathbb{R}^{d\times d}$ with

$$H = I - 2uu^\top \tag{2.1}$$

with $u \in \mathbb{R}^d$ the hyperplane unit normal vector and the corresponding outer product $uu^\top$. The reflection can be easily intuited when applied to a weight vector $w \in \mathbb{R}^d$:

$$Hw = (I - 2uu^\top)w = w - 2u(u^\top w).$$

Transformation H effectively subtracts twice the component of w projected on u, thereby reflecting it with respect to the hyperplane defined by u (see Fig. 2.1). By construction, hyperplane reflections are well-suited for the efficient finetuning of pretrained models, as they keep the distance to the transformation neutral element - the identity matrix - constant, which minimizes the risk of divergence from the pretrained model and deterioration of model performance (c.f. Fig. 2.4). This can be easily shown by computing the Frobenius norm of the difference between the Householder matrix H and the identity matrix I:

$$\|H - I\|_F = \|I - 2uu^\top - I\|_F = 2 \cdot \|uu^\top\|_F = 2 \tag{2.2}$$

The above equation leverages the fact that for any matrix M

$$\|M\|_F = \sqrt{\mathrm{Tr}(MM^\top)}$$

and that with $M = uu^\top$ and u having unit length $u_1^2 + u_2^2 + ... + u_d^2 = 1$, one can simply write (with $(uu^\top)^\top = uu^\top$)

$$\|uu^\top\|_F = \sqrt{\textstyle\sum_{i=1}^{d} u_i^2} = 1.$$

Since the finetuning process simply consists of finding the optimal directions of the reflection hyperplanes with bounded deviations from the transformation neutral element, it allows for (i) a very *low number of extra parameters* corresponding to the unit vectors u, and (ii) the usage of high learning rates, as *the risk of divergence is minimized*. This allows for general *learning rate robustness* and encourages fast convergence by default, as consistently high learning rates can be selected; reducing computational resources required to achieve good performance (e.g. Fig. 2.6).

Interestingly, as this transformation is orthogonal ($HH^\top = I$), it falls under the umbrella of orthogonal transformations motivated in OFT [115] from the perspective of Hyperspherical Energy control to better preserve model pretraining. However, OFT leverages the Cayley parametrization of orthogonal matrices, which only produces determinant 1 matrices. By construction, this excludes Householder matrices from OFT, which have determinant -1! However, as noted above, it is indeed in this particular setting and through the use of Householder transformations that high parameter efficiency, strong pretraining retention, and learning rate robustness arise.

On top of that, we further investigate the importance of Hyperspherical Energy retention by conducting a control study comparing OFT against its non-orthogonal variant

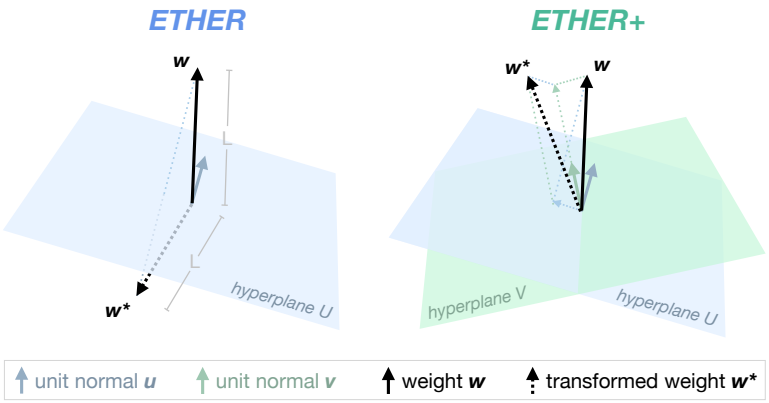

Figure 2.1: ***ETHER* and *ETHER+* sketches.** We visualize either a single hyperplane reflection for *ETHER* or two interacting hyperplanes for *ETHER+*, parametrized unit normals u (and v). Unlike *ETHER*, the final result of *ETHER+* does not have to retain the original length L, as the need for hard reflections is softened, and orthogonality is no longer guaranteed.

(*Naive*)[1]. Our experiments do not show significant differences in terms of control and training stability, suggesting that such properties stem from the multiplicative finetuning approach rather than the underlying HE retention, contrasting insights in [115] (c.f. Sec. 2.5.3). These findings partly motivate the exploration of a relaxed variant of the Householder reflection in the next section 2.3.3, which demonstrates that loosening the orthogonality constraint not only maintains good performance but can even lead to enhanced results.

2.3.3 Relaxing Orthogonality in ETHER

While finetuning via hyperplane reflections has several promising qualities as highlighted above, there is no free lunch. In particular, situations may arise where the strength of the transformation and inherent deviation from the identity may be too large by default, such as for potentially more nuanced tasks like subject-driven generation [126]. To allow for more nuanced transformations while retaining beneficial properties of *ETHER* - parameter efficiency and learning rate robustness through bounded deviations from the transformation neutral element - we propose the *ETHER+* relaxation

$$H^+ = I - uu^\top + vv^\top$$

with unit vectors $u, v \in \mathbb{R}^d$. This is a simple variation of the Householder transformation that now allows for interaction between two distinct hyperplanes (see Fig. 2.1). This helps to control the transformation strength as $uu^\top$ and $vv^\top$ can weaken or even cancel each other out to return the identity transformation in the limit where $u = v$. In addition,

[1] *Naive* employs an unconstrained block-diagonal transformation matrix N^B made up of n blocks and initialized as an identity matrix, i.e. having the same number of trainable parameters and initialization as OFT's transformation matrix Q^B.

the transformation distance remains bounded, as the relaxed variant H^+ always has $\|H^+ - I\|_F \leq 2$, i.e.

$$\max \|H^+ - I\|_F \leq \max \|H - I\|_F \,.$$

This follows immediately from the triangle inequality of norms, i.e. $\|vv^\intercal - uu^\intercal\|_F \leq \|vv^\intercal\|_F + \|uu^\intercal\|_F = 2$. Due to the weaker strength of this new transformation, we apply it both on the left (H^+) and right ($\tilde{H}^+$) of the weight matrix W, such that the forward pass becomes

$$\left(H^+ W \tilde{H}^+\right)^\intercal x + b.$$

Consequently, *ETHER+* effectively leverages a sequence of hyperplane interactions that no longer have to retain length to allow for more nuanced weight adjustment while still minimizing the risk of diverging from the pretrained model (as also shown e.g. in Figs. 2.3, 2.4, 2.5 and 2.6).

2.3.4 Efficient *ETHER* through Block-Parallelism

In multiplicative finetuning like OFT or *ETHER*, further computational load is introduced through additional matrix multiplications. To mitigate this issue, we introduce a block-diagonal formulation of *ETHER* similar to block-diagonal OFT described in §2.3.1. For this, we break down the Householder transformation H (eq. 2.1) into its corresponding block-diagonal variant H^B:

$$\operatorname{diag}(H^1 \cdots H^n) = I - 2 \begin{pmatrix} \hat{u}_1 \hat{u}_1^\intercal & & \\ & \ddots & \\ & & \hat{u}_n \hat{u}_n^\intercal \end{pmatrix}$$

with each i-th block-plane parameterized by $\hat{u}_i \in \mathbb{R}^{\frac{d}{n}}$. Of course, one can do the same for H^+. In both cases, such a block-diagonal formulation reduces the cost of computing H. More importantly, each i-th block now only affects the corresponding i-th block-row in the weight matrix W. This means we can split W into n sub-blocks $W^i \in \mathbb{R}^{\frac{d}{n} \times f}$, each of which is uniquely altered by its corresponding H^i counterpart. As a result, the full weight transformation can now be separated into smaller block-specific operations, reducing the overall number of computations. Furthermore, these operations can now be fully block-parallelized, significantly increasing training speed! In terms of computations, for each full-matrix-multiplication between H and W of sizes $d \times d$ and $d \times f$ respectively, $d(df)$ multiplications and $(d-1)df$ additions are necessary, accounting for $O(d^2 f)$ operations. With our block-parallel scheme, we reduce these to n block-specific $\frac{d}{n}(\frac{d}{n} f)$ multiplications and $\frac{d-1}{n}(\frac{d}{n} f)$ additions, resulting in $O(\frac{d^2 f}{n})$ operations (see Tab. 2.1).

Furthermore, with each block being built from a single vector of dimension $\frac{d}{n}$, *ETHER* transformations' construction ensures that the total number of trainable parameters remains constant for any n number of blocks. This stands in contrast to block-diagonal OFT, where the use of higher block counts was introduced to minimize the number of

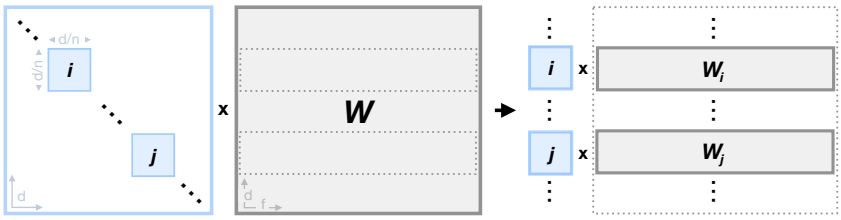

Figure 2.2: **Block-Parallel Computation scheme** between d-dimensional block-diagonal transformation with n blocks and a $d \times f$ -dimensional weight matrix W.

Table 2.1: **Better computational efficiency through block-diagonality** on **Phi1.5**-1.3B and **Llama-2**-7B, with internal dimensions of 2048 and 4096 respectively. As the number of blocks n increases, so does the computational efficiency, quantified by the decrease in TFLOPs required for a single backward pass (using a sample with longest sequence length). The larger the model's internal dimension, the larger the efficiency gain.

	Phi1.5-1.3B		**Llama-2-7B**	
	TFLOPs	rel. drop	TFLOPs	rel. drop
LoRA$_{r=8}$	6.04	-	6.85	-
OFT$_{n=256}$	9.13	-	25.26	-
ETHER$_{n=1}$	9.13	-	25.26	-
ETHER$_{n=4}$	7.07	-23%	12.07	-52%
ETHER$_{n=32}$	6.71	-27%	8.22	-68%
ETHER+$_{n=1}$	10.78	-	51.65	-
ETHER+$_{n=4}$	7.69	-29%	18.66	-64%
ETHER+$_{n=32}$	6.79	-37%	9.04	-83%

parameters while introducing noticeable decreases in adaptation performance! Instead, for block-diagonal *ETHER*, we find performance to be consistent over increasing block counts (see Supp. A.4), allowing for an improved computational footprint with negligible performance decrease.

2.4 Intriguing Properties of *ETHER*

This section investigates and highlights the bounded distance and non-deteriorating nature of *ETHER*/*ETHER*+ in more detail while providing insights into its favorable learning rate robustness and the reliable use of high learning rates for fast convergence. For completeness, we also report here comparisons with the unconstrained *Naive* method, to better show the impact of orthogonality as proposed by [115], and how our method provides much stronger robustness. Finally, we include a discussion on the parameter efficiency. For all experiments in this section, please see §2.5.1 for relevant implementation details.

Non-Deteriorating Nature. Because both *ETHER* and *ETHER*+ are upper-bounded in their possible perturbation over the pretrained weight matrices (as measured for example by the distance to the transformation neutral element, the identity matrix), finetuning with both methods will guarantee suitable results for most hyperparameter choices. This is easily visualized in Fig. 2.3 by looking at generation samples after perturbing Stable Diffusion with randomly sampled transformations for each approach - OFT, *ETHER* and *ETHER*+ - respectively. While *ETHER* uses a fixed-distance transformation (c.f. Eq. 2.2) that introduces a noticeable change (but still retaining semantics), *ETHER*+ can obtain both finegrained visual control as well as stronger semantic changes. Conversely, unbounded methods like OFT catastrophically deteriorate a model's generative abilities as the perturbation strength increases.

This results in a much more controlled generation setting for *ETHER* and *ETHER*+ finetuning. This is also depicted quantitatively in Fig. 2.4, which shows distances between the learned transformation and the transformed weights (at convergence) to the identity matrix and the pretrained weights, respectively, as a function of the learning rate. As can be seen, larger learning rate values for OFT and *Naive* finetuning (OFT without orthogonality constraints) result in distances that are orders of magnitude higher than those of *ETHER* and *ETHER*+, leading to catastrophic deterioration and model collapse (see Fig. A.1 in Supp.).

Learning Rate and Hyperparameter Robustness. Practically, the non-deteriorating nature of *ETHER* and *ETHER*+ manifests in learning rate robustness during finetuning. As the risks of divergence and collapse are minimized, training stability becomes much less dependent on the choice of learning rate. This is seen when evaluating performance (e.g. mIoU for controllable image synthesis in Fig. 2.5) and model convergence (Fig. 2.6) against learning rates. For non-*ETHER* methods, Fig. 2.5 shows significant performance drops for high learning rates, while Fig. 2.6 reveals fast convergence speeds for *ETHER*+ with learning rates covering multiple magnitudes, much more general than e.g. OFT.

This means that not only can good performance be guaranteed for most learning rate choices, but fast convergence as well, with competitive results already after the first epoch.

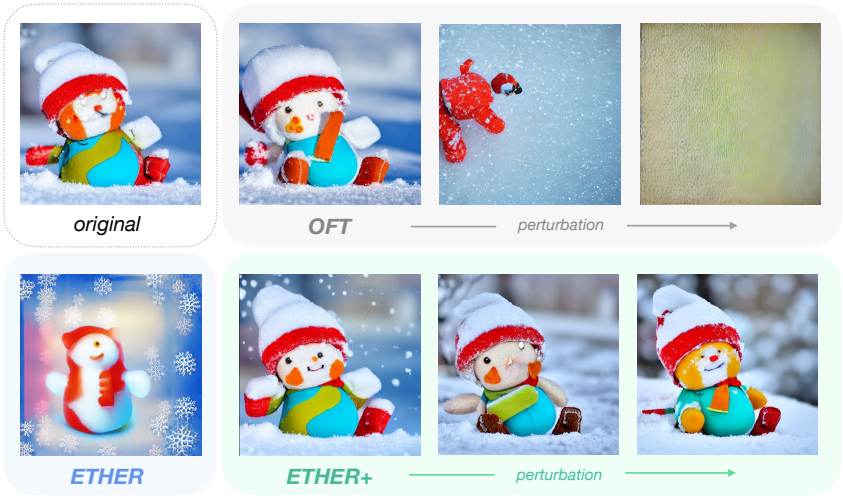

Figure 2.3: **Change in model behavior as a function of perturbation strength**, i.e. distance between weight transformation and identity matrix. As *ETHER* and *ETHER+* are upper-bounded in perturbation by construction, catastrophic deterioration of model performances is rarely encountered, and weight transformations remain controllable even for maximal deviations. For standard approaches, s.a. OFT, larger deviations from the identity matrix may occur during training and result in substantial divergence from the pretrained model. Notice also that by breaking orthogonality constraints in *ETHER+*, both smaller and stronger semantic variants can be learned.

Since *ETHER* also only introduces a single hyperparameter, the number of diagonal blocks, which marginally impacts performance (c.f. §2.3.4), *ETHER* methods become very attractive for practical usage, as the need for grid-search and cautious low learning rate training for good performance (c.f. §4.1) is reduced.

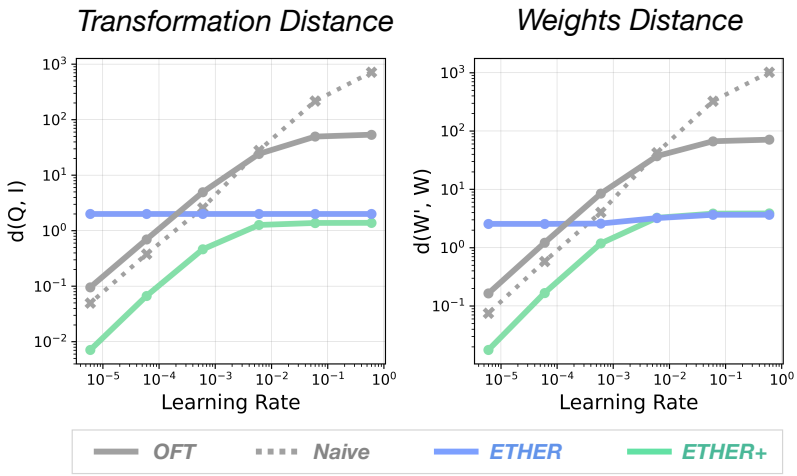

Figure 2.4: **Distances as a function of learning rates** between transformation and identity matrix (*Transformation Distance*), and finetuned and pretrained weights (*Weights Distance*). Distances obtained for subject-driven generation finetuning at convergence (1200 iterations). Results show distances magnitudes higher and unbounded for non-*ETHER* methods in both cases as learning rates increase.

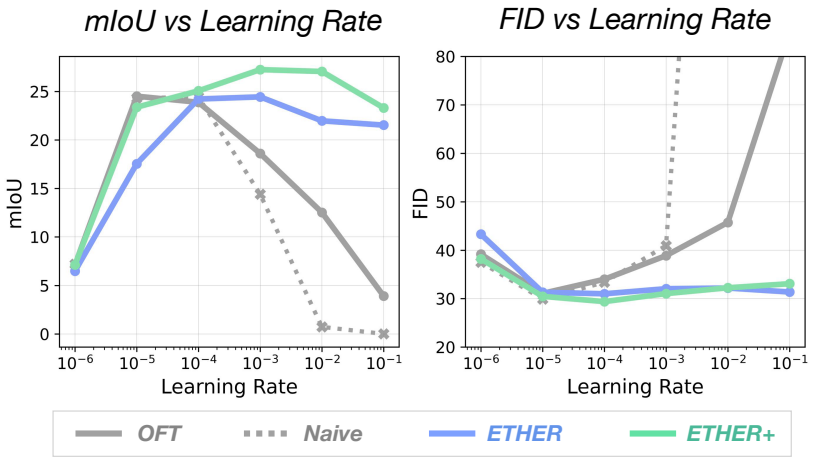

Figure 2.5: **mIoU and FID performances as a function of learning rates.** Results are obtained for controllable generation S2I finetuning on Stable Diffusion, and reveal a much stronger learning rate robustness of *ETHER*-based methods; retaining strong performance across entire learning rate magnitudes.

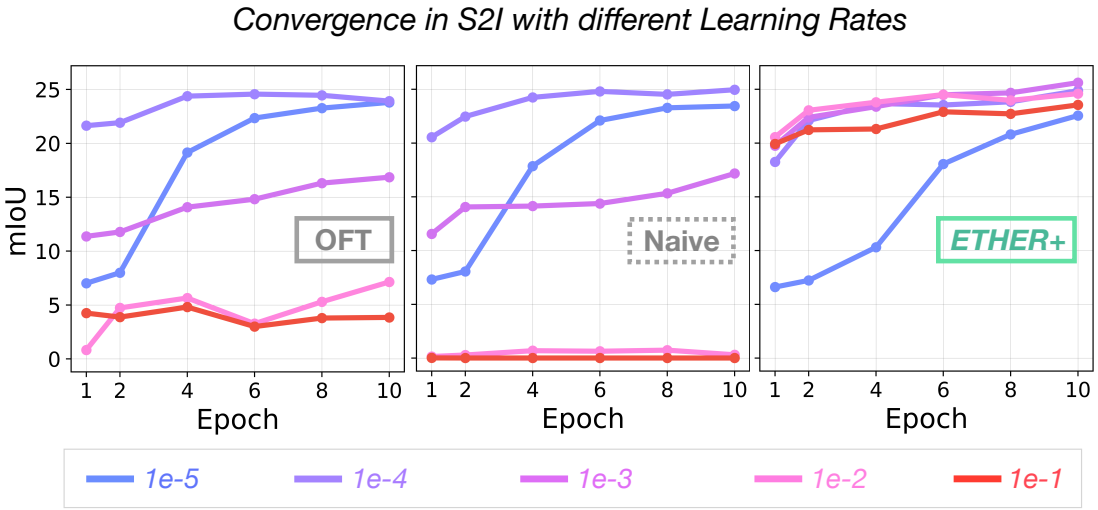

Figure 2.6: **Achieved controllability (mIoU) per epoch for different finetuning methods.** This figure extends Fig. 2.5 and highlights in detail how only a learning rate of 10^{-4} allows for optimal convergence in OFT and Naive, while for *ETHER*+ fastest convergence speeds are stably achieved across magnitudes.

Parameter Efficiency. Finally, we provide a more detailed exploration on the parameter efficiency of *ETHER*-based methods. Let L be the number of finetuned layers, d and f the respective weight dimensions for $W \in \mathbb{R}^{d \times f}$. Then the parameter complexity for OFT can be written as $O(\frac{Ld^2}{n})$ [115] with n number of diagonal blocks[2]. Similarly, for LoRA we get $O(Lr(d+f))$, while for *ETHER* and *ETHER*+ we only have $O(Ld)$ and $O(L(d+f))$

[2] [115] note a possible $O(Ld)$ if $n = \alpha d$. However, in practice, equally scaling n with d disproportionally reduces adaptation parameters for large weight matrices. As OFT is fairly dependent on the parameter count, we omit this estimate.

respectively. With respect to both LoRA and OFT, this omits at the very least the rank multiplier r, or a potentially quadratic scaling. As already motivated in Sec. 2.3, this results in incredibly efficient finetuning while achieving comparable or stronger performances. For example, when finetuning Stable Diffusion as done above, *ETHER* and *ETHER*+ use 120 times and 30 times fewer parameters than OFT respectively.

2.5 Benchmark Experiments

We first investigate generative model adaptation in Sec. 2.5.1, with a focus on subject-driven image synthesis (§2.5.1.1) and controllable image synthesis (§2.5.1.2) following recent works [93, 115]. Sec. 2.5.2 then correspondingly investigates language model adaptation, looking at both natural language understanding (§2.5.2.1) and instruction tuning (§2.5.2.2). Finally, we study the importance of orthogonality and hyperspherical energy on finetuning performance in Sec. 2.5.3.

2.5.1 *ETHER* for Image-generative Model Adaptation

For our experiments on diffusion-based generative models, we apply the finetuning methods on the pretrained Stable Diffusion-v1.5 [123], following the setting from OFT [115]. Our experiments follow best practices and hyperparameter choices for each method. For implementation details, please refer to Supp. A.3.

2.5.1.1 Subject-driven Generation

We first deploy *ETHER* and *ETHER*+ on subject-driven generation following [115, 126]; finetuning the generative model for each of the 30 subjects and 25 prompts. For each combination, we generate four images, and measure image quality via a DINO [14] and a CLIP image encoder [116], text-prompt fidelity via a CLIP text encoder, and image diversity using LPIPS [161].

Quantitative Results. Results are shown in Tab. 2.2. On subject-driven generation, we find competitive performance for both image quality, text-prompt fidelity and image diversity, particularly for *ETHER*+ (e.g. DINO and CLIP-I scores of 0.666 vs 0.652 and 0.8 vs 0.794 for OFT, respectively). Most importantly, we achieve this performance while only utilizing a fraction of tuning parameters; with *ETHER*+ only introducing $0.4M$ as compared to $11.6M$ by OFT. As hypothesized in Sec. 2.3, for nuanced finetuning, *ETHER*'s transformation strength seems to be too high to retain key semantic concepts in subject-driven generation, falling short in image quality with respect to other methods (e.g. also qualitatively depicted in Fig 2.3), despite achieving strong image diversity and text-prompt fidelity.

Table 2.2: **Subject-driven Generation Results.** We use r to denote rank, and n the number of diagonal blocks. We measure image quality (DINO, CLIP-I), text-prompt fidelity (CLIP-T) and image diversity (LPIPS). *ETHER*+ addresses finegrained adaptation shortcomings of *ETHER* (c.f. Sec. 2.3.3) and achieves strong performance with only few adaptation parameters.

	#params	DINO ↑	CLIP-I ↑	CLIP-T ↑	LPIPS ↑
Real Images	-	0.703	0.864	-	0.695
DreamBooth	859.5M	0.644	0.793	0.236	0.709
LoRA$_{r=4}$	0.8M	0.660	0.796	0.231	0.714
OFT$_{n=4}$	11.6M	0.652	0.794	0.241	0.725
ETHER	0.1M	0.567	0.746	**0.256**	**0.766**
ETHER+	0.4M	**0.666**	**0.800**	0.240	0.729

Table 2.3: **Semantic Map to Image Results.** We use n to denote the number of diagonal blocks. *ETHER* and particularly *ETHER*+ achieve strong synthesis control (mIoU, Acc) with few parameters while retaining good image alignment (FID). We indicate with (+ magn. r.f.) the OFT version with magnitude re-fitting.

	#params	mIoU ↑	Acc ↑	FID ↓
Encoder-only	0	8.2	38.0	41.2
OFT$_{n=4}$	13.2M	24.5	62.8	31.1
OFT$_{n=4}$ (+ magn. r.f.)	13.4M	24.6	63.3	**30.8**
ETHER	0.1M	24.6	63.3	32.0
ETHER+	0.4M	**27.3**	**68.1**	31.0

2.5.1.2 Controllable Image Generation

This section applies *ETHER* for controllability of Stable Diffusion following [115] for the Semantic Map to Image (S2I) task on ADE20K [172]. We use the trainable encoder from ControlNet [159] for the control signal and perform finetuning on the Stable Diffusion weights only. We report a baseline with just the control signal encoder to highlight relative gains through finetuning. Evaluations are performed on 2000 images generated from the validation set using mean Intersection-over-Union (mIoU) and accuracy of semantic maps over generated images using UperNet-101 [150] pretrained on ADE20K. Finally, we measure the similarity between generated and original images via FID [49]. For OFT, we also test magnitude re-fitting [115] for an additional epoch.

Quantitative Results. Results are depicted in Tab. 2.3, and clearly demonstrate competitive control with both *ETHER* and *ETHER*+. Unlike subject-driven image generation, we find that *ETHER* performs on the same level as OFT multiplicative finetuning while using over 100× fewer parameters (e.g. 24.6 versus 24.5 mIoU of OFT with $0.1M$ versus $13.2M$ parameters). Introducing magnitude re-fitting to OFT yields only limited gains while adding $0.2M$ parameters. Similar to Tab. 2.2 for subject-driven image generation, we find that for controllable image synthesis, the *ETHER*+ relaxation provides additional performance gains (e.g. 27.3 vs 24.5 mIoU and 68.1 vs 62.8 Acc against OFT). Taking into

account the more robust (Fig. 2.5) and faster convergence (Fig. 2.6), this presents *ETHER*+ as a practically attractive finetuning alternative.

2.5.2 *ETHER* for Language Models Adaptation

To understand the applicability of the *ETHER* transformation family in the language domain, we follow [93]'s and [54]'s experimental setup. For fair comparisons, we run grid searches over the most relevant hyperparameters in common value ranges. For additional implementation details, please refer to Supp. A.3.

2.5.2.1 Natural Language Understanding

We begin by deploying *ETHER* and *ETHER*+ on the widely utilized [25, 46, 70, 96] GLUE benchmark [140], finetuning a pretrained DeBERTaV3-base model [46] following [93], from which we report the baselines' results. GLUE comprises various English sentence understanding tasks, such as inference tasks (MNLI, QNLI, RTE), classification of sentiment (SST-2) or correct English grammatical structures (CoLA), and semantic similarity and equivalence prediction (MRPC, QQP, STS-B). CoLA scores report the Matthews correlation coefficient, MNLI matched accuracy, and STS-B average correlation. All other tasks are evaluated on accuracy.

Quantitative Results. Results in Tab. 3.4 show that *ETHER* and *ETHER*+ match and even outperform previous methods with significantly fewer parameters. For example, *ETHER* outperforms the second-best BOFT on the RTE inference task (89.53 vs 88.81) or equivalence prediction on MRPC (93.68 vs 92.40) while using just one-ninth of the parameters ($0.085M$ compared to $0.75M$). *ETHER*+ sets both the best performance on STS-B and particularly the highest overall score (90.10) using less than half of the parameters of BOFT. These results provide additional support for the practical viability of *ETHER* transformations, now for natural language adaptation - being a strong, but much more parameter-efficient competitor.

2.5.2.2 Instruction Tuning

Our instruction tuning experiments make use of Llama-2-7B [137] as pretrained model, finetuning it on the Alpaca dataset [132] for one epoch. To operate on a consumer GPU, we truncate the maximum sequence length to 256 and use bfloat16 precision [60]. We evaluate 0-shot performance of our instruction-tuned model on (i) Massive Multitask Language Understanding (MMLU) [48] with 57 different tasks in four different subjects (STEM, Humanities, Social Sciences, Others); (ii) the AI2 Reasoning Challenge (ARC) [21], a common-sense reasoning dataset of questions from science grade exams; (iii) TruthfulQA [87] comprising 817 questions spanning 38 categories testing how much the model (wrongly) relies on imitation of human text to answer.

Quantitative Results. Results in Tab. 2.5 show that both *ETHER* and *ETHER*+ outperform comparable finetuning approaches while utilizing fewer parameters. Across all metrics,

Table 2.4: **GLUE benchmark.** Comparisons of different methods finetuning DeBERTaV3-base. Results of all baselines are taken from [93]. We use r to denote rank, and n the number of diagonal blocks. As can be seen, *ETHER* and *ETHER+* achieve competitive performances across metrics while utilizing fewer parameters (up to a magnitude in the case of *ETHER*) while also retaining all practical benefits such as learning rate robustness depicted e.g. in Sec. 2.4.

	#params	MNLI↑	SST-2↑	CoLA↑	QQP↑	QNLI↑	RTE↑	MRPC↑	STS-B↑	**Avg↑**
Full Finet.	184M	89.90	95.63	69.19	**92.40**	94.03	83.75	89.46	91.60	88.25
BitFit	0.10M	89.37	94.84	66.96	88.41	92.24	78.70	87.75	91.35	86.20
H-Adapter	1.22M	90.13	95.53	68.64	91.91	94.11	84.48	89.95	91.48	88.28
P-Adapter	1.18M	90.33	95.61	68.77	92.04	94.29	85.20	89.46	91.54	88.41
LoRA$_{r=8}$	1.33M	90.65	94.95	69.82	91.99	93.87	85.20	89.95	91.60	88.50
AdaLoRA	1.27M	**90.76**	96.10	71.45	92.23	**94.55**	88.09	90.69	91.84	89.46
OFT$_{n=16}$	0.79M	90.33	96.33	**73.91**	92.10	94.07	87.36	92.16	91.91	89.77
BOFT$^{m=2}_{n=8}$	0.75M	90.25	**96.44**	72.95	92.10	94.23	88.81	92.40	91.92	89.89
ETHER	0.09M	90.23	96.10	71.31	91.42	94.31	**89.53**	**93.68**	92.30	89.86
ETHER+	0.33M	90.52	96.33	72.64	92.22	94.33	**89.53**	92.89	**92.35**	**90.10**

Table 2.5: **Instruction Tuning.** We use r to denote rank, and n the number of diagonal blocks. Both *ETHER* and *ETHER+* outperform LoRA/OFT which use up to a magnitude more parameters, and beat VeRA with similar parameter counts.

	#params	MMLU↑	ARC↑	Tru-1↑	Tru-2↑
Llama-2-7B	-	41.81	42.92	25.21	38.95
VeRA$_{r=64}$	0.27M	42.30	45.13	27.41	41.04
VeRA$_{r=256}$	1.05M	42.21	43.85	25.33	39.02
LoRA$_{r=1}$	0.52M	42.40	44.62	27.05	41.94
LoRA$_{r=8}$	4.19M	43.61	46.16	28.76	42.21
OFT$_{n=256}$	2.09M	42.92	44.88	27.42	41.11
ETHER$_{n=32}$	0.26M	44.57	45.14	27.91	41.83
ETHER+$_{n=32}$	1.04M	**44.87**	**46.50**	**29.38**	**43.51**

the Llama-2-7B baseline is consistently surpassed by significant margins (e.g. 44.87 MMLU for *ETHER+* vs the 41.81 baseline, or 46.50 vs 42.92 ARC score). Despite being the most parameter-efficient method, *ETHER* outperforms all baselines with comparable number of parameters, such as the recently introduced VeRA [70] with rank $r = 64$, and LoRA rank 1. Surprisingly, increasing the rank of VeRA to 256 leads to a decrease in performance, while LoRA rank 8 shows better results but is still outperformed on MMLU despite having 16× more parameters. On the other hand, *ETHER+* surpasses all other methods across all benchmarks, while having 4× fewer parameters than LoRA rank 8.

2.5.3 Hyperspherical Energy for Effective PEFT

[115] link finetuning stability and performance obtained by transforming the weights via matrix-multiplication to the orthogonality of the transformations, and a consequently unaltered hyperspherical energy (HE). To test this assumption, we have included an OFT

control baseline (*Naive*), which does not utilize orthogonality constraints, on the same finetuning settings in which OFT was proposed. Results at convergence, as reported in Tab. 2.6, do not show significant differences, while actually introducing the overhead of computing the Cayley parametrizations (which also involve computing the inverse of a matrix). We also included the *Naive* baseline in the learning rate robustness studies in Fig. 2.4 and Fig. 2.5, showcasing that while differences are present for high learning rates, the optimal working range remains unaltered. Finally, we validate that the HE indeed varies during training, as reported in Fig. 2.7.

In contrast, on these same evaluations, our newly proposed *ETHER* transformation family, by introducing a boundary on the Euclidean distance on the transformation side, achieves stronger performance and greater robustness. This is especially true for the non-orthogonal *ETHER+*, which alters the overall HE even more than *Naive* (Fig. 2.7). This evidence diminishes the role of the HE and instead emphasizes the greater importance of the Euclidean distance, establishing the *ETHER* family as a favorable option in multiplicative finetuning settings.

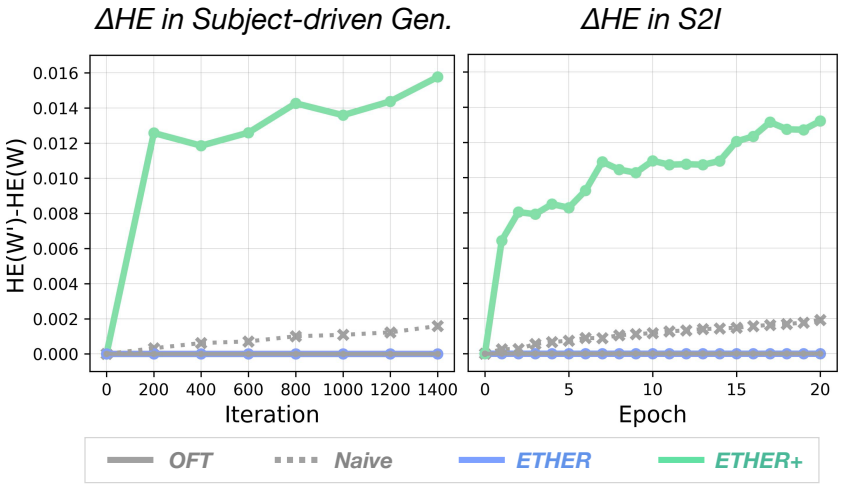

Figure 2.7: **Difference in HE** between finetuned/pretrained models for Subject-driven Generation and S2I. Notice that by removing the orthogonality constraint, both *ETHER+* and Naive alter the HE of the pretrained model, while OFT and *ETHER* do not.

Table 2.6: **OFT vs Naive.** OFT performance-test against its non-orthogonal counterpart Naive. We show that results don't differ significantly, questioning the relevance of HE retaining for finetuning performance.

	Subject-driven Generation				**S2I**		
	DINO	CLIP-I	CLIP-T	LPIPS	mIoU	Acc	FID
$\text{OFT}_{n=4}$	0.652	0.794	0.241	0.725	24.5	62.8	31.1
$\text{Naive}_{n=4}$	0.648	0.793	0.245	0.730	24.3	62.9	29.9

2.6 Conclusions

Our paper introduces the *ETHER* family of transformations for parameter-efficient finetuning. Based on the Householder formulation of hyperplane reflections, *ETHER* methods frame finetuning as a search for unit normal vectors that define hyperplanes along which weight vectors are reflected. In doing so, *ETHER* (and its relaxation *ETHER+* for more finegrained adaptation) fix (or upper bound) the distance of learned transformations from the identity matrix (the transformation neutral element), thereby minimizing the risk of finetuning divergence. Put together, *ETHER* methods operate more parameter-efficiently than other PEFT methods (e.g., around 10-100 times less than LoRA or OFT), have higher learning rate robustness and encourage fast convergence. Consequently, *ETHER* transformations require less expansive hyperparameter searches to achieve good performance, making them very attractive for practical deployment.

Limitations. Of course, there is no free lunch. While both *ETHER* and its relaxation *ETHER+* show strong results with few parameters across a broad range of tasks, increasing the expressive power of the transformation is not as straightforward as in other methods, such as LoRA, where one can adjust the rank parameter to more closely approximate full finetuning. Moreover, multiplicative methods introduce a computational overhead during training compared to additive methods. Thanks to our block-parallel scheme, we make significant progress towards closing the gap between multiplicative and additive approaches; however, multiplicative methods still lag behind. This introduces a trade-off between parameter efficiency and computational overhead when achieving similar performance levels.

3

Decoupling Angles and Strength in Low-rank Adaptation

Parameter-Efficient FineTuning (PEFT) methods have recently gained significant popularity thanks to the widespread availability of large-scale pretrained models. These methods allow for quick adaptation to downstream tasks with minimal computational cost. However, popular finetuning methods such as LoRA exhibit limited robustness when it comes to hyperparameter choices or extended training regimes, preventing optimal out-of-the-box performance. In contrast, bounded approaches, such as *ETHER*, provide greater robustness but are limited to extremely low-rank adaptations and fixed-strength transformations, reducing their adaptation expressive power. In this work, we propose DeLoRA, a novel finetuning method that normalizes and scales learnable low-rank matrices. By bounding the distance of the transformation, DeLoRA effectively decouples the angular learning from the adaptation strength, enhancing robustness without compromising performance. Through evaluations on subject-driven image generation, natural language understanding, and instruction tuning, we show that DeLoRA matches or surpasses performance of competing PEFT methods, while exhibiting stronger robustness. Code is available at `https://github.com/ExplainableML/DeLoRA`.

3.1 Introduction

The rapid advancement of deep learning has led to the development of large-scale pre-trained models in various domains, especially in computer vision and natural language processing [116, 123, 136, 137]. However, the enormous size of these models, reaching billions of parameters, presents significant challenges when adapting them to specific downstream tasks, particularly in terms of computational cost and efficiency. To address these challenges, Parameter Efficient FineTuning (PEFT) methods have emerged. PEFT methods are characterized by their introduction of a small set of learnable parameters, in contrast to the extensive parameter updates required in full finetuning. Notable examples include adapters [51] and prompt tuning [78]. In this work, we focus on enhancing LoRA [54], a widely adopted finetuning method known for its simplicity and effectiveness. However, despite its success, LoRA is sensitive to hyperparameter choices [9] and often exhibits performance degradation during extended finetuning [115]. While robust finetuning approaches such as *ETHER* and *ETHER+* [11] address some of these limitations, they are constrained to extremely low-rank adaptations and fixed-strength transformations.

Therefore, we propose DeLoRA, an enhanced version of LoRA that introduces a boundary on the weight updates through normalization, decoupling the angular learning from the adaptation strength. This enhances adaptability across diverse settings while preserving capabilities for personalization and merging at inference time. We motivate DeLoRA from two distinct perspectives: as an extension of LoRA through the introduction of additional normalization, and as an evolution of *ETHER* by enabling high-rank updates. We conduct ablation studies on the design choices and demonstrate improvements over both LoRA and *ETHER*. Additionally, we validate the advantages of DeLoRA by evaluating it across diverse tasks in image generation and LLM adaptation.

In summary, we make the following contributions in this work: (i) we thoroughly review the formulations of LoRA and *ETHER* and derive a novel PEFT method, DeLoRA; (ii) we demonstrate DeLoRA enhanced robustness and decoupling compared to alternatives; (iii) we extensively ablate the formulation of DeLoRA by deriving it from both LoRA and *ETHER*; (iv) we evaluate DeLoRA on both vision and language benchmarks, matching or surpassing the performance of competing PEFT methods.

3.2 Decoupled Low-rank Adaptation (DeLoRA)

Our decoupled low-rank adaptation approach, by introducing learnable boundaries on the weight updates, effectively combines the strengths of LoRA and *ETHER* methods, allowing for high expressivity and finetuning robustness. In the following sections, we will (i) present an overview of the PEFT methods LoRA and *ETHER*, focusing on their respective limitations (section 3.2.1) and (ii) describe how we derive our proposed DeLoRA method from both perspectives (section 3.2.2), along with a comparison with DoRA [92], a method that also targets decoupling angular and magnitude components.

3.2.1 Preliminaries: LoRA & *ETHER*, and Their Limitations

Here, we provide a detailed review of LoRA [54] and *ETHER* [11], with a particular focus on their limitations.

Low-rank Adaptation (LoRA). *Low-rank Adaptation* (*LoRA*) [54] parametrizes the update of pretrained weights $W \in \mathbb{R}^{d \times f}$ during finetuning as

$$\left(W + \frac{\alpha}{r} BA\right)^\intercal x + b \tag{3.1}$$

where $A \in \mathbb{R}^{r \times d}$ and $B \in \mathbb{R}^{f \times r}$ are the learnable matrices, α is a scaling factor, and r is the rank of the final BA matrix. When $r \ll \min(d, f)$, LoRA substantially reduces the number of parameters required for finetuning compared to full finetuning. Furthermore, BA matrices can be integrated into W at inference time, eliminating additional latency.

However, LoRA is known to be highly sensitive to hyperparameter choices [9], and it is prone to deterioration with over-training [115], thus requiring careful tuning and experimentation to achieve an optimal balance between a sufficiently high learning rate and avoiding catastrophic overwriting of the pretrained weights. In our proposed DeLoRA, we mitigate this behavior by introducing a boundary to the weight updates, which results in robust performance across a broad range of learning rates.

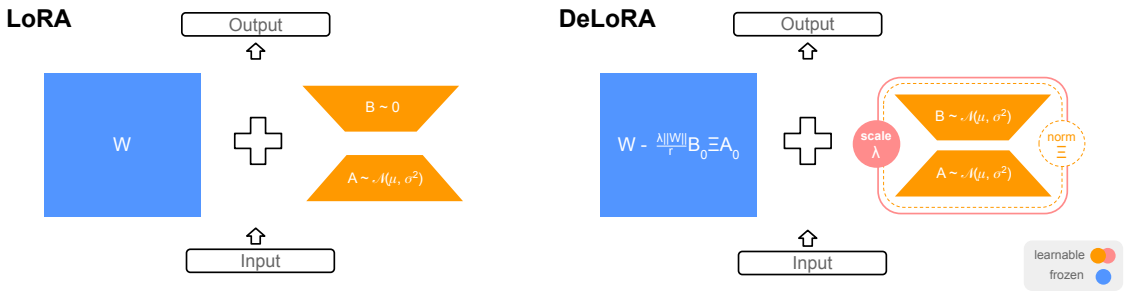

Figure 3.1: Visualizations (Left) of the original LoRA [54] and (Right) of our proposed method DeLoRA. In addition to the low-rank matrices B, A, we introduce a normalization Ξ and a scaling factor λ, which effectively decouple the angular learning from the adaptation strength.

Finetuning with Hyperplane Reflections (*ETHER*). Following efficiency and robustness arguments, [11] propose to employ bounded transformations for finetuning, namely *ETHER* and *ETHER+*. *ETHER* (left side in eq. (3.2)) and *ETHER+* (right side) introduce multiplicative transformations H or H^+ respectively, which act on the pretrained weights as follows:

$$(HW)^\intercal x + b \quad , \quad \left(H^+ W \tilde{H}^+\right)^\intercal x + b. \tag{3.2}$$

Here, $H = I - 2uu^\intercal$, $H^+ = I - uu^\intercal + vv^\intercal$, $\tilde{H}^+ = I - \tilde{u}\tilde{u}^\intercal + \tilde{v}\tilde{v}^\intercal$ (where u, v, $\tilde{u}$, $\tilde{v}$ are unit vectors) are bounded in terms of their distance to the identity transformation, as per

$$\|H - I\|_F = 2 \quad , \quad \|H^+ - I\|_F \leq 2, \tag{3.3}$$

where the subscript $_F$ denotes the Frobenius norm. This upper bound on the transformation distance prevents weight changes that cause catastrophic overwriting, as shown by [11].

However, enforcing a constant boundary on the transformation distance can limit the finetuning performance, as the boundary may be too strict to adapt the layer or pretrained model at hand to the respective task. Furthermore, by rewriting the formulations in eq. (3.2) in a residual form, we can show that the weight updates are intrinsically limited to be low-rank (see section B.1), which limits the finetuning capacity of such methods. In DeLoRA, by introducing a normalization and a scaling factor to LoRA matrices, we show how to achieve robustness comparable to *ETHER* while enabling control over both boundary and rank, ultimately enhancing model expressivity and performance.

3.2.2 DeLoRA

While both LoRA and *ETHER* demonstrate valuable properties, namely parameter efficiency and robustness, they also exhibit notable limitations. Our proposed PEFT method, DeLoRA, addresses these shortcomings by synthesizing the strengths of both approaches. In this regard, DeLoRA can be thought of as an extension of LoRA that incorporates *ETHER*'s robustness properties or, alternatively, as an enhancement of *ETHER* that adopts LoRA's more expressive paradigm. In the following, we will present both derivations and finally summarize in a concise way our proposed DeLoRA formulation.

Deriving DeLoRA from LoRA. In order to achieve robustness to learning rates, we first observe that in LoRA's eq. (3.1) the norm of the weight updates ΔW is proportional to ΔBA, which in turn is proportional to the learning rate. This means that the update strength at each training step is directly driven by the learning rate, which can lead to catastrophic overwriting in high learning rate regimes. In order to mitigate this behavior, we want to introduce a normalization term. To do this, we start by decomposing the BA matrix into the sum of its rank-1 components, i.e.

$$BA = \sum_{i=1}^{r} b_i a_i^\intercal \tag{3.4}$$

■ *Controllable Boundary.* Similarly to *ETHER*, we normalize each rank-1 entry, making the Frobenius norm of each single rank-1 component equal to 1. This normalization can be introduced as in

$$\sum_{i=1}^{r} \frac{b_i a_i^\intercal}{\|b_i\|\|a_i\|} = B\Xi A \tag{3.5}$$

where Ξ is a diagonal matrix with entries $\Xi_{i,i} = \frac{1}{\|b_i\|\|a_i\|}$ for $i = 1, \ldots, r$, $\Xi_{i,j} = 0$ for $i, j = 1, \ldots, r, i \neq j$. The final update distance with respect to the pretrained weights thus is bounded as

$$\|B\Xi A\| = \left\| \sum_{i=1}^{r} b_i a_i^\intercal \right\| \leq \sum_{i=1}^{r} \|b_i a_i^\intercal\| = r \tag{3.6}$$

Most importantly, the boundary is independent of the used learning rate. Next, to control the boundary and remove its rank dependency, we scale $B\Xi A$ by a factor $\frac{\lambda}{r}$, as in

$$\left\| \frac{\lambda}{r} B\Xi A \right\| \leq \lambda. \tag{3.7}$$

Now, the boundary is equal to λ and can be chosen arbitrarily to better fit the pretrained network or task at hand. To enable greater flexibility and layer-specific boundaries, we make each distinct λ learnable, allowing finetuning to adapt their values accordingly. Hence, we effectively decouple the angular learning (the normalized $B\Xi A$ matrices) from the adaptation strength, as measured by the boundary λ. Furthermore, introducing a single additional learnable parameter λ to each finetuned matrix creates only negligible overhead in terms of overall trainable parameters and training speed.

■ *Weights-norm Scaling.* Previous works suggest that when finetuning image generative models such as Stable Diffusion, multiplicative finetuning methods exhibit stronger performance [93, 115] than additive counterparts. We argue this may arise because multiplicative methods induce weight updates relative to the pretrained weights W, meaning updates are inherently layer-specific. This might be especially relevant when adapting a diverse set of layers, which is the case for our Stable Diffusion adaptations (see fig. 3.3). To mimic this approach, in our additive proposed method DeLoRA, we introduce a scaling factor equal to the pretrained weights norm. This can be formally stated as

$$\Delta W = \frac{\lambda \|W\|}{r} B\Xi A. \tag{3.8}$$

Our ablation studies on Stable Diffusion finetuning tasks demonstrate such performance improvements empirically (see section 3.3.2).

■ *Initialization.* To initialize the finetuning process from the pretrained model, DeLoRA's normalization operation does not allow to simply follow LoRA's zero initialization of the B matrix. From preliminary experiments, we find that introducing a small epsilon to avoid division by 0, would sometimes lead to unstable results. Therefore, we instead

take inspiration from [11, 106] and subtract a copy of the kaiming-randomly initialized matrices to the frozen pretrained weights, as in

$$W = \bar{W} - \left(\frac{\lambda\|W\|}{r} B\Xi A\right)_0 \tag{3.9}$$

where $\bar{W}$ is the original pretrained matrix, and $(\frac{\lambda\|W\|}{r} B\Xi A)_0$ is the update matrix at time 0.

Deriving DeLoRA from *ETHER* So far, we showed how to derive DeLORA from LoRA. Alternatively, it is possible to derive DeLoRA by introducing properties of LoRA to *ETHER*. We find this to be insightful to understand the impact of each individual component from a theoretical perspective. In addition, we quantitatively ablate all innovations of DeLoRA in section 3.3.2.

■ *Controllable Boundary.* One of the primary limitations of *ETHER* and *ETHER+* is their fixed boundary (see section 3.2.1), which is fixed and thus cannot be adapted to the pretrained model in use. We address this limitation by introducing a scaling parameter λ as in

$$H = I - \lambda uu^\top \quad , \quad H^+ = I - \frac{\lambda}{2} uu^\top + \frac{\lambda}{2} vv^\top. \tag{3.10}$$

Consequently, the boundaries on the distances of H and H^+ from the identity matrix become $\|H - I\|_F = \lambda$, and $\|H^+ - I\|_F \leq \lambda$. In section 3.3.2, we show that this modification, i.e. introducing a controllable bound, leads to the largest increase in performance.

■ *Increasing the rank.* In section B.1, we demonstrate that *ETHER* and *ETHER+* are restricted to rank-1 and rank-4 weight updates respectively. In order to arbitrarily control the rank, we extend the H^+ parameter of *ETHER+* to $\hat{H}$, which allows for an arbitrary number of weight reflection operations:

$$\hat{H} = I - \sum_{i=1}^{r/2} u_i u_i^\top + \sum_{i=1}^{r/2} v_i v_i^\top. \tag{3.11}$$

We can rewrite $\hat{H}$ by gathering the u and v unit vectors into two rank-$\frac{r}{2}$ matrices, as in

$$\hat{H} = I - U\Sigma U^\top + V\Theta V^\top, \tag{3.12}$$

where Σ and Θ are diagonal normalization matrices with entries $\Sigma_{i,i} = \frac{1}{\|u_i\|^2}$, $\Theta_{i,i} = \frac{1}{\|v_i\|^2}$, The entries on the diagonals of Σ and Θ are constructed to normalize u and v to unit vectors. Thus, the distance from the identity matrix becomes

$$\|\hat{H} - I\| = \left\| \sum_{i=1}^{r/2} u_i u_i^\top - \sum_{i=1}^{r/2} v_i v_i^\top \right\| \leq \sum_{i=1}^{r/2} \|u_i u_i^\top\| + \sum_{i=1}^{r/2} \|v_i v_i^\top\| = r. \tag{3.13}$$

As above, we can control the boundary on the distance, and remove the rank dependency, by introducing a scaling factor $\frac{\lambda}{r}$ as in

$$\hat{H} = I - \frac{\lambda}{r} U\Sigma U^\top + \frac{\lambda}{r} V\Theta V^\top \tag{3.14}$$

■ *U,V Relaxation.* Finally, we relax $U\Sigma U^\intercal, V\Theta V^\intercal$ and replace them with distinct trainable matrices $B\Xi A$ and $D\Phi C$ respectively, which leads to $\hat{H} = I - \frac{\lambda}{r}(B\Xi A - D\Phi C)W$. We emphasize how this formulation resembles a multiplicative analog of our proposed DeLoRA method, and we include this variant in our ablation study. We ablate all alternatives in Section 3.3.2. There, we find that DeLoRA, combined with weights-norm scaled updates, as in multiplicative finetuning, achieves overall stronger performance.

DeLoRA formulation. Summarizing, our proposed DeLoRA finetuning method consists in learning a normalized low-rank matrix $B\Xi A$ and a scale λ, updating the pretrained weights as in

$$\left(W + \frac{\lambda\|\bar{W}\|}{r} B\Xi A\right)^\intercal x + b \tag{3.15}$$

This formulation inherently constrains the learnable finetuning updates in a $\lambda\|\bar{W}\|$-sized ball, where $\bar{W}$ is the norm of the pretrained weights, achieving a decoupling of the transformation strength from the angular learning.

In more detail, the key components are:

- *Normalization:* Ξ is a r-dimensional diagonal matrix that normalizes LoRA's inner low-dimensional bottleneck (eq. (3.5)), bounding the Frobenius norm of $B\Xi A$ to r (eq. (3.6)).
- *Scaling Factors:* (i) $1/r$ is used to remove the rank dependency on the boundary dimensionality, (ii) $\|\bar{W}\|$ to make the weight updates proportional to the pretrained weights, and (iii) λ to control the adaptation strength and allow for a layer-specific boundary adaptation (eq. (3.7))
- *Initialization:* Pretrained initialization follows by merging to the pretrained weights a frozen copy of the initialized finetuning adaptation matrices (eq. (3.9)).

DoRA vs DeLoRA discussion. DoRA [92], similarly to our work, addresses finetuning targeting the decoupling of angular and magnitude components, by using a formulation that leads to weight updates $W' = m\frac{W+\Delta W}{\|W+\Delta W\|}$. We can summarize the key differences between DoRA and our proposed method in two main aspects: (i) DoRA applies normalization and scaling operations on the fully finetuned weights, and (ii) these operations are performed on the column space of the weight matrices, which significantly differs from our approach. In contrast, we argue that DeLoRA finetuning has two key advantages: (i) by introducing the normalization and scaling operations directly on the weight updates ΔW, it more effectively prevents divergence from the pretrained model, and (ii) by normalizing the inner low-dimensional space (as opposed to the column space), it implicitly enforces a Frobenius-norm boundary, providing a mathematical guarantee against divergence. These ultimately result in (i) peculiar training dynamics (as depticted in fig. 3.3, whereas DoRA and LoRA exhibit similar behavior), and (ii) enhanced decoupling, supported by

the robustness performance in fig. 3.2 and in section B.3. In this regard, we notice that although DeLoRA's learnable boundary theoretically allows an unbounded Frobenius norm, divergence from the pretrained weights does not happen in practice, as also shown in section B.4. This demonstrates that during finetuning, DeLoRA's learnable boundary is able to effectively adjust and avoid divergence from the pretrained weights–behavior that is not observed with DoRA.

3.3 Experiments

In this section, we evaluate our proposed DeLoRA method for image generation, natural language understanding, and instruction tuning tasks. We begin by providing a detailed description of these tasks and their relevance. To justify our design choices, we present a comprehensive ablation study that highlights the key innovations of DeLoRA. Finally, we demonstrate that DeLoRA not only matches or exceeds the performance of LoRA and other state-of-the-art methods but also exhibits superior robustness. This enhanced stability is particularly evident in two aspects: reduced sensitivity to learning rate selection and improved performance retention during extended finetuning periods.

3.3.1 Tasks

Subject-driven Image Generation. Following [11, 115], we assess the effectiveness of our proposed methods in the DreamBooth setting [126], specifically by adapting Stable Diffusion [123] to recontextualize a subject shown in a set of images according to a given prompt. The dataset, sourced from [126], comprises 30 subjects, each paired with 25 prompts. The task is to finetune Stable Diffusion to generate images portraying the given subject in the context defined by the prompts. We report an example in in section B.5 (fig. B.3, left side). For each combination of image and prompt, after finetuning, we generate four images and measure the subject-fidelity by DINO [14] and CLIP [116], as proposed by [126]. Here, the score represents the similarity of generated and given images, measuring the faithfulness of generating images of the given subject to the provided real images. Among the two metrics, the DINO score is more significant since it is more sensitive to subject-unique features [126].

Semantic Map to Image Following [11, 115], we evaluate the ability of our proposed methods in finetuning Stable Diffusion to generate realistic images based on given segmentation maps. The image should follow the spatial structure laid out in the segmentation map as closely as possible. Examples of segmentation maps and their corresponding generated images are presented in section B.5 (fig. B.3, right side). For the control signal, we use the pretrained encoder from ControlNet [159]. For training and evaluation, we utilize semantic maps and images from the ADE20K dataset [172]. After training, we generate images for 2000 segmentation masks from the ADE20K validation set and report

the mean Intersection-over-Union (mIoU) and accuracy of semantic maps as predicted by UperNet-101 [150]. Note that we only use the Semantic Map to Image task to ablate our method design decisions.

Natural Language Understanding We evaluate DeLoRA's performance in adapting small-scale language models by finetuning and evaluating a pretrained RoBERTa-base model [96] on the General Language Understanding Evaluation (GLUE) benchmark [140]. GLUE tasks have been extensively used to measure natural language understanding performance, comprising inference tasks (MNLI, QNLI, RTE), sentiment classification (SST-2), and correct identification of English grammatical structures (CoLA). CoLA results refer to Matthews correlation coefficient, MNLI to matched accuracy, and STS-B to average correlation, while all other tasks are evaluated on accuracy. For a proper evaluation on the validation set, we adopt the setup proposed by [147], and split the validation set into two subsets, guarded by a pre-defined seed, that will be used for model selection and evaluation. We provide more details in section 3.3.3.

Instruction Tuning. We evaluate how effectively DeLoRA can adapt LLMs to follow user-given instructions, finetuning LLaMA-2-7B [137] on the Alpaca dataset [132]. Following bini2024ether, we evaluate the zero-shot performance of instruction-tuned models on four different tasks, namely (1) Massive Multitask Language Understanding (MMLU) [48], which features 57 tasks in different categories such as STEM, Humanities, and Social Sciences; (2) AI2 Reasoning Challenge (ARC) [21], which contains over 7000 grade-school science questions; (3) TruthfulQA [87], which contains 817 questions representing common misconceptions in 38 categories like health, law, finance and politics. TruthfulQA additionally features two separate sub-tasks, namely single-true and multi-true. In single-true, only one of the provided answers is correct, and the model has to select the unique correct answer. In multi-true, several of the provided answers may be correct, and the model has to assign a high probability to correct answers and a low probability to incorrect answers.

3.3.2 Ablation of DeLoRA design choices

In this section, we ablate the incremental design choices that transform LoRA and *ETHER+* into DeLoRA, evaluating these on the subject-driven generation and semantic map-to-image tasks. From the LoRA derivation (top-down in Tables 3.1,3.2), we show how incorporating normalization with a controllable boundary and weight scaling into pre-trained matrices enhances performance. From the *ETHER+* derivation (bottom-up in Tables 3.1,3.2), we show how introducing a controllable scale, a higher-rank formulation, relaxed learnable matrices, and an additive finetuning tranformation, incrementally improves performance.

Method	ΔW formulation	DINO	CLIP-I
LoRA [rank-r]	BA	0.674	0.785
$\downarrow$ + normalize w/ controllable boundary	$\frac{\lambda}{r}B\Xi A$	0.682	0.809
$\cdot$ + normalize w/ controllable boundary + weights-scaling $\cdot$ + controllable boundary + high rank + relaxed + additive FT **(DeLoRA)**	$\frac{\|W\|\lambda}{r}B\Xi A$	**0.701**	0.825
$\uparrow$ + controllable scale + high rank + relaxed	$\frac{\lambda}{r}(B\Xi A - D\Phi C)W$	<u>0.696</u>	<u>0.833</u>
$\|$ + controllable boundary + high rank	$\frac{\lambda}{r}(U\Sigma U^\intercal - V\Theta V^\intercal)W$	0.685	**0.840**
$\|$ + controllable boundary	$\lambda(uu^\intercal - vv^\intercal)W$	0.678	0.810
***ETHER*+ (one-sided)** [rank-2, boundary equal to 2]	$(uu^\intercal - vv^\intercal)W$	0.624	0.746

Table 3.1: Ablation of DeLoRA innovations on the **Subject-driven Image Generation** task. We show how different components affect performance from both LoRA and *ETHER* derivation.

Method	ΔW Formulation	mIoU$\uparrow$	Acc.$\uparrow$	FID$\downarrow$
LoRA [rank-r]	BA	25.13	64.95	31.35
$\downarrow$ + normalize w/ controllable boundary	$\frac{\lambda}{r}B\Xi A$	<u>25.66</u>	**65.82**	31.01
$\cdot$ + normalize w/ controllable boundary + weights-scaling $\cdot$ + controllable boundary + high rank + relaxed + additive FT **(DeLoRA)**	$\frac{\|W\|\lambda}{r}B\Xi A$	**26.10**	65.08	<u>30.71</u>
$\uparrow$ + controllable boundary + high rank + relaxed	$\frac{\lambda}{r}(B\Xi A - D\Phi C)W$	25.55	<u>65.16</u>	**29.89**
$\|$ + controllable boundary	$\lambda(uu^\intercal - vv^\intercal)W$	24.56	62.70	31.28
***ETHER*+ (one-sided)** [rank-2, boundary equal to 2]	$(uu^\intercal - vv^\intercal)W$	23.46	62.26	31.18

Table 3.2: Ablation of DeLoRA innovations on the **Semantic Map to Image** task. We show how different components from both LoRA and *ETHER* derivations incrementally improve performance.

Results for subject-driven image generation are in table 3.1. For this ablation we use a small-scale version of the setting proposed by [126], finetuning 3 subjects over 25 prompts each (10% of the data). Among all modifications, we notice how the introduction of a controllable boundary in *ETHER*+ (one-sided) has the highest impact, raising the DINO score from 0.624 to 0.678 and the CLIP score from 0.746 to 0.810. This shows how the lack of strength is the hindering factor for *ETHER*+(one-sided), as already noted by [11]. Starting from LoRA, we notice how the weights-norm scaling has the largest impact on performance, raising the DINO score from 0.682 to 0.701 and the CLIP score from 0.809 to 0.825. Additionally, we note that DeLoRA's performance without the weights-norm scaling falls short compared to its multiplicative counterpart.

For the Semantic Map to Image ablation study, we run a small-scale grid search by finetuning Stable Diffusion for 10 epochs on ADE20K in bfloat16 precision. Results are reported in Table table 3.2. We note how DeLoRA achieves best controllability among different variations. In addition, we also note the increase in Accuracy when increasing the rank of *ETHER*+, hinting that it could have been a limiting factor.

3.3.3 Benchmark Results

Subject-Driven Image Generation Results are in table 3.3. For a comprehensive benchmark performance comparison, we report low-rank results from [11], while running and evaluating LoRA, DoRA, and DeLoRA methods at a consistent rank. For each method, we conduct a grid search to identify optimal hyperparameters using the same 3 subjects

as in the ablation studies, then evaluate the top-performing configurations on the full 30-subject benchmark, testing each across three distinct seeds. The best and average results are reported in table 3.3. We notice that LoRA, DoRA, and DeLoRA, all achieve comparable average performance in terms of DINO and CLIP-Image, all outperforming lower-rank baselines. This shows that DeLoRA is able to effectively combine *ETHER*+ robustness properties with superior performance.

Method		#param	DINO	CLIP-I
Real Images			0.703	0.864
DreamBooth	[126]	859.5M	0.644	0.793
$\text{OFT}_{n=4}$	[115]	11.6M	0.652	0.794
ETHER+	[11]	0.4M	0.666	0.800
$\text{LoRA}_{r=4}$	[54]	0.8M	0.660	0.796
$\text{LoRA}_{r=16}$	[54]	3.2M	<u>0.686</u>	0.818
$\text{DoRA}_{r=16}$	[92]	3.2M	**0.687**	<u>0.819</u>
$\text{DeLoRA}_{r=16}$	(ours)	3.2M	<u>0.686</u>	**0.820**
$\text{LoRA}^{\dagger}_{r=16}$	[54]	3.2M	0.688	0.818
$\text{DoRA}^{\dagger}_{r=16}$	[92]	3.2M	<u>0.689</u>	<u>0.819</u>
$\text{DeLoRA}^{\dagger}_{r=16}$	(ours)	3.2M	**0.693**	**0.820**

Table 3.3: Results for evaluating DeLoRA in **subject-driven image generation**. † indicates experiments with tuned hyperparameters.

Natural Language Understanding Results are in table 3.4. For proper evaluation on the GLUE validation set, we follow [147, 148] and split the validation set into two subsets (determined by pre-defined seeds), and use the first subset to tune hyperparameters, and the second subset to evaluate method performance. For fair comparisons we use same seeds as [147, 148]. In addition, in order to compare with LoRA's implementation, we simply apply DeLoRA to Q,V attention layers with rank 8, which is likely sub-optimal with respect to applying lower-rank modules to a larger set of layers [54]. We notice how DeLoRA achieves better performance on CoLA, QNLI and STS-B, and an overall significantly better average score with respect to all baselines, demonstrating its efficacy in adapting language models for NLU tasks.

Instruction Tuning Results are in table 3.5. Results for all methods but DoRA and DeLoRA are reported from [11]. For these two, a proper grid search has been run following the same setup of [11]. Further details cab be found in B.2. We can see that DeLoRA achieves best results on three out of four tasks. This confirms the effectiveness of our improvements, which lead to optimal average performance in this setup. On the MMLU task, *ETHER* and *ETHER*+ outperform other methods, but fall short on other tasks, achieving lower average performance compared to DeLoRA. This might be due to the limited capacity of *ETHER* methods from their rank limitation.

3.3.4 Insights

In this section we analyze (i) the learning rate robustness properties, and (ii) the training dynamics, with a focus on prolonged training setting, of DeLoRA with respect to other finetuning methods. Then, we analyze (iii) how weights norms differ in a pretrained model, to better understand the weights-norm scaling effect in DeLoRA.

Learning Rate Robustness. We conducted a comprehensive learning rate robustness analysis in the setting of the Subject-driven Generation task of section 4.4. Evaluation is done reporting DINO scores (Fig.3.2, Left) and Euclidean distance between finetuned and pretrained weights of a projection layer in an attention module (Fig.3.2, Right) across multiple methods, using a range of learning rates derived from each method's base learning rate. Our analysis shows that DeLoRA is able to achieve the same robustness of *ETHER+*, while improving performance, whereas both LoRA and DoRA performance degrade at 4× the base learning rate. We also notice how LoRA updates' distance grows at higher learning rates, while interestingly DoRA, after 8×, does not diverge further, likely thanks to its magnitude control. However this does not lead to better performance in these regimes.

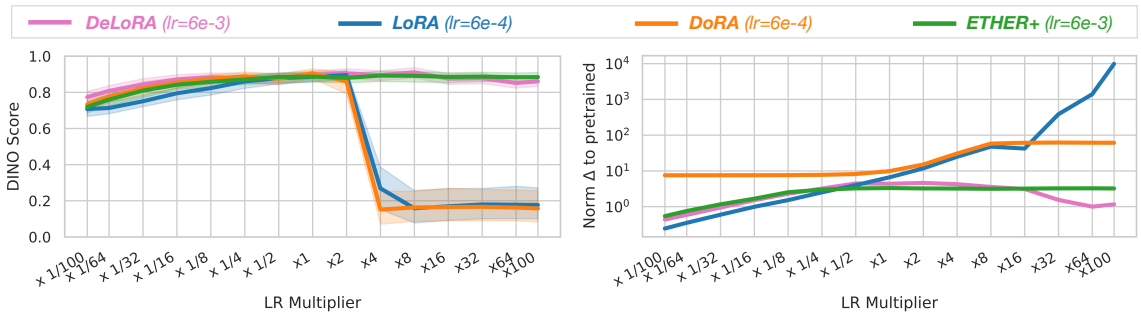

Figure 3.2: Learning rate robustness plots in Subject-driven generation task in terms of DINO scores (Left) and Euclidean distance between a finetuned vs pretrained projection layer weights (Right). Learning rates used for robustness evaluation were derived by multiplying the base learning rate in a range of factors.

Method		#param	MNLI	SST-2	MRPC	CoLA	QNLI	QQP	RTE	STS-B	Avg
Full Finet.		125M	87.3	94.4	87.9	62.4	92.5	91.7	78.3	90.6	85.6
BitFit	([155])	0.1M	84.7	**94.0**	88.1	54.0	91.0	87.3	69.8	89.5	82.3
IA3	([89])	0.06M	85.4	93.4	86.4	57.8	91.1	88.5	73.5	88.5	83.1
LoReFT	([148])	0.02M	83.1	93.4	**89.2**	60.4	91.2	87.4	**79.0**	90.0	84.2
RED	([147])	0.02M	83.9	93.9	**89.2**	61.0	90.7	87.2	78.0	90.4	84.3
LoRA	([54])	0.3M	86.6	93.9	88.7	59.7	**92.6**	90.4	75.3	90.3	84.7
Adapter^{FFN}	([113])	0.3M	**87.1**	93.0	88.8	58.5	92.0	90.2	77.7	90.4	84.7
Adapter	([51])	0.4M	87.0	93.3	88.4	60.9	92.5	**90.5**	76.5	90.5	85.0
DeLoRA	(ours)	0.3M	86.9	93.7	88.6	**64.7**	**92.6**	90.2	77.3	**90.6**	**85.6**

Table 3.4: Comparisons of different methods finetuning RoBERTa-base on **GLUE benchmark**. Results of all baselines are taken from [147] and [148].

Method		#param	MMLU	ARC	Tru-1	Tru-2	Avg
LLaMA-2-7B		-	41.81	42.92	25.21	38.95	37.22
$ETHER_{n=32}$	([11])	0.26M	<u>44.57</u>	45.14	27.91	41.83	39.86
$ETHER+_{n=32}$	([11])	1.04M	**44.87**	46.50	<u>29.38</u>	<u>43.51</u>	<u>41.07</u>
$LoRA_{r=8}$	([54])	4.19M	43.61	46.16	28.76	42.21	40.19
$DoRA_{r=8}$	([92])	4.19M	43.24	<u>47.18</u>	29.01	43.47	40.73
$DeLoRA_{r=8}$	(ours)	4.19M	44.21	**47.70**	**29.62**	**44.14**	**41.42**

Table 3.5: Results for **Instruction Tuning** on MMLU, ARC, and TruthfulQA benchmarks. Values represent accuracy scores achieved by different finetuning methods. Best scores are highlighted in bold, and second-best scores are underlined.

Finetuning Regime and Prolonged Training. We further investigate the behavior of weight updates across different methods by measuring the Euclidean distance between finetuned weight matrices (after merging) and the pretrained corresponding matrices during fine-tuning. This provides us a quantitative measure of the shift and rate at which fine-tuned weight matrices diverge from the pretrained weights. In fig. 3.3 (Left), we show this analysis for the out-projection matrix in one of StableDiffusion's Unet self-attention layers. We find that LoRA- and DoRA-trained weights continuously depart from the pretrained weights over the course of training, passing through an optimal regime but eventually overshooting and ending in a diverging regime (notice that best performance are typically found between 1000 and 1400 steps). In contrast, DeLoRA-trained weights exhibit a peculiar behavior, quickly moving away from the pretrained weights, until they reach the boundary, from which they cannot diverge further. We argue that this leads to prolonged training robustness, effectively avoiding catastrophic overwriting. Qualitative examples are provided in fig. 3.3 (Right) and in section B.5. Additionally, we highlight that by adjusting the boundary parameter λ, one can easily control the maximum allowable shift and, therefore, the level of finetuning robustness.

Figure 3.3: (Left) Euclidean Distance of finetuned weights to pretrained weights as a function of the number of training steps. (Right) Qualitative examples show that LoRA exhibits significant artifacts earlier in the process compared to DeLoRA, which maintains better image quality.

Weights Norms Heterogeneity. In fig. 3.4, we show the mean of column norms for weight matrices in different attention blocks of the U-Net in Stable Diffusion v1.5. By doing so, we highlight the effect of weights-norm scaling as introduced in section 4.3. We find that different modules, as well as different positions in the U-Net, show systematic differences with respect to weights norms. This points at differences within the pretrained model

which finetuning methods should account for. Our proposed scaling is one possibility to accomplish this. Exploring more sophisticated methods to include layer-wise differences is an interesting direction for future research.

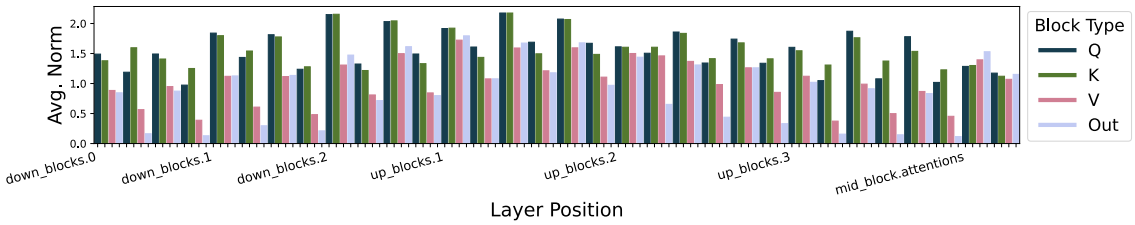

Figure 3.4: Average column norms of parameters in the attention modules of Stable Diffusion's Unet

3.4 Related Work

Parameter efficient finetuning (PEFT) is an active field of research, encompassing methods such as adapters [51], prompt- and prefix-tuning variations [78, 81, 95], and more specialized methods such as BitFit [155], FourierFT [37], and LayerNorm Tuning [168]. In this paper, we propose an improved PEFT method based on low-rank adapters (LoRA) first described by [54]. Therefore, we focus our review of previous work on LoRA variants and refer to recent surveys [43, 151] regarding PEFT methods in general. LoRA is a popular finetuning approach for large models, featuring advantages such as low-memory footprint and no additional inference cost [54]. Compared to full-finetuning, LoRA is also less prone to catastrophic forgetting [9].

However, beyond falling behind in performance on downstream tasks compared to full finetuning [9], previous work has identified and attempted to address different limitations of the original LoRA method. [85, 121, 149, 175] propose methods to overcome the low-rank limitation without sacrificing memory efficiency. Similarly, VeRA [70] keeps the original LoRA setup but reduces trainable parameters further by only scaling the randomly initialized matrices, which are shared across layers. To account for differences between layers, [26, 98, 160, 163] describe methods to dynamically adapt the rank of different LoRA adapters. Instead of changing the rank, in this work, we propose to dynamically change the scaling of LoRA matrices for different layers, highlighting the need for layer-adaptive methods. PiSSA [106] and MiLoRA [141] show how improved initialization of LoRA can lead to better performance and faster convergence. [173] and [44] show that LoRA matrices behave differently in terms of optimal initialization and learning rate. Our work is complementary to these findings, as we also argue for different treatments of LoRAs, but regarding different layers within a model, not within the same adapter. DoRA [92], similarly to our work, targets decoupling of angles and magnitudes, normalizing and scaling the full updated weight matrix $W + \Delta W$ on the column space, controlling each

singular column of the finetuned matrices, whereas we propose to normalize the inner r-dimensional space of each ΔW update matrix.

3.5 Conclusions

In this work, we proposed a novel parameter efficient finetuning method, DeLoRA, which combines the strengths of LoRA –controllable rank– and *ETHER* –bounded updates– to address their respective limitations. We showed that by normalizing and scaling low-rank updates, DeLoRA is able to effectively decouple the angular learning from the adaptation strength, leading to competitive performance and enhanced robustness. Beyond showing the advantages of DeLoRA, we provided detailed insights into its derivation, from both perspective of LoRA and *ETHER*, ablating the introduction of each incremental innovation. Finally, we investigated DeLoRA's robustness to learning rate variations and extended training, demonstrating that its decoupled update mechanism is critical for preventing divergence from the pretrained weights. These findings offer valuable perspectives for adapting pretrained models, by addressing key limitations of current PEFT approaches.

4

MemLoRA: Distilling Expert Adapters for On-Device Memory Systems

Memory-augmented Large Language Models (LLMs) have demonstrated remarkable consistency during prolonged dialogues by storing relevant memories and incorporating them as context. Such memory-based personalization is key also in on-device settings that allow users to keep their conversations and data private. However, memory-augmented systems typically rely on LLMs that are too costly for local on-device deployment. Even though small language models (SLMs) are more suitable for on-device inference than LLMs, they cannot achieve sufficient performance. Additionally, these LLM-based systems lack native visual capabilities, limiting their applicability in multimodal contexts. In this paper, we introduce *(i)* MemLoRA, a novel memory system that enables local deployment by equipping SLMs with specialized memory adapters, and *(ii)* its vision extension MemLoRA-V, which integrates small Vision-Language Models (SVLMs) to memory systems, enabling native visual understanding. Following knowledge distillation principles, each adapter is trained separately for specific memory operations—knowledge extraction, memory update, and memory-augmented generation. Equipped with memory adapters, small models enable accurate on-device memory operations without cloud dependency. On text-only operations, MemLoRA outperforms 10× larger baseline models (e.g., Gemma2-27b) and achieves performance comparable to 60× larger models (e.g., GPT-OSS-120b) on the LoCoMo benchmark. To evaluate visual understanding operations instead, we extend LoCoMo with challenging Visual Question Answering tasks that require direct visual reasoning. On this, our VLM-integrated MemLoRA-V shows massive improvements over caption-based approaches (81.3 vs 23.7 accuracy) while keeping strong performance in text-based tasks, demonstrating the efficacy of our method in multimodal contexts.

4.1 Introduction

Recent advancements in Large Language Models (LLMs) and Vision Language Models (VLMs) have led to their widespread use in conversational Artificial Intelligence (AI) systems, ranging from customer service chatbots to personal assistants and collaborative productivity tools [107, 170]. VLMs have demonstrated remarkable capabilities in multimodal understanding and generation, making them increasingly integral to human-computer interaction across diverse domains. However, the effectiveness of VLMs in real-world on-device conversational applications is fundamentally constrained by LLMs' limited context windows [18, 103]. While modern LLMs can process thousands of tokens in a single session, they cannot retain information across multiple conversations or maintain long-term user-specific knowledge. This limitation becomes particularly problematic in multi-session scenarios where users expect the system to remember previous interactions, preferences, and contextual details—a critical requirement for delivering truly personalized and coherent conversational experiences.

To address these challenges, researchers have proposed various memory systems that extend LLMs with persistent memory capabilities. Early approaches focused on integrating external memory through differentiable attention mechanisms [145] and retrieval-augmented generation from knowledge bases [79], establishing foundational paradigms for extending model knowledge beyond immediate context. Building on these foundations, recent works have explored sophisticated memory management strategies that mirror human cognitive processes, including temporal decay mechanisms for selective retention [171], hierarchical memory systems inspired by the design of operating systems [111], and knowledge graph representations that track evolving information over time [119]. Contemporary systems have further expanded the role of LLMs beyond generation, leveraging them as active agents within the memory pipeline itself. Examples include using LLMs to automatically extract knowledge and update the memory [18, 76], evaluate memory relevance and quality [103], and dynamically restructure knowledge networks according to emerging patterns [152], thereby transforming memory augmentation from a passive retrieval mechanism into an intelligent, self-improving system.

Despite these advances, current memory systems face significant practical limitations that restrict their deployment and effectiveness. Firstly, these systems fundamentally rely on large, often proprietary, LLMs for core memory operations—including extraction, organization, updating, and retrieval—necessitating continuous Application Programming Interface (API) calls to cloud-based services [18, 111]. This dependency not only introduces latency and cost concerns but also prevents on-device deployment, limiting their applicability in privacy-sensitive contexts, offline scenarios, or resource-constrained environments where cloud connectivity cannot be guaranteed. In our work, we tackle this challenge by replacing queries posed to a large-scale model through API, with a small on-device model, equipped with task-specific expert adapters. These adapters are trained via knowledge distillation through teacher answers or ground-truth data. We provide an

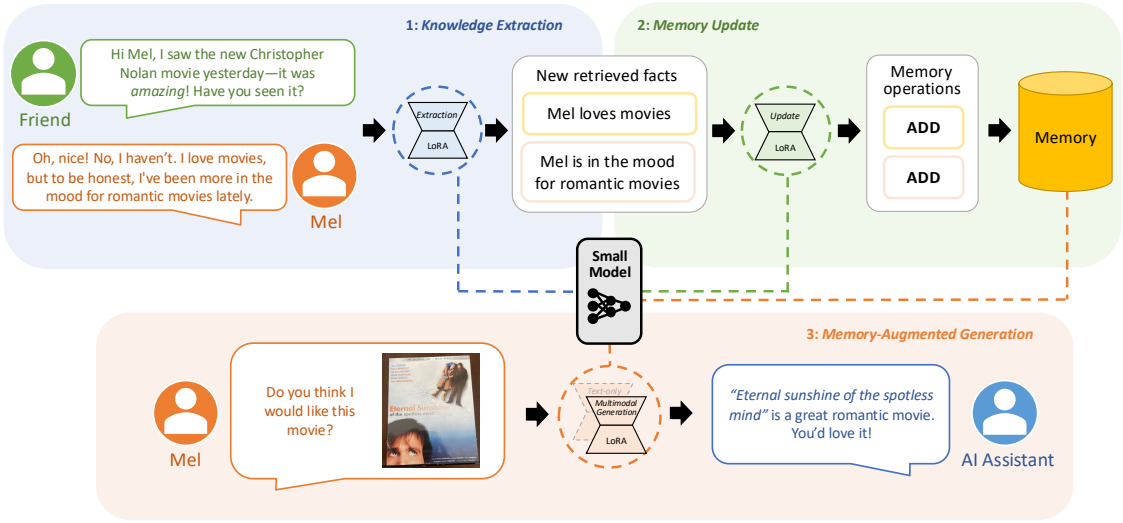

Figure 4.1: **Overview**. We employ specialized LoRA adapters to enable S(V)LMs to perform memory operations for on-device deployment. The base S(V)LM dynamically switches between expert adapters, each trained for a distinct stage: (1) *knowledge extraction*, (2) *memory update*, (3) *memory-augmented generation*. In the last stage, the model can switch between *text-only* and *multimodal* adapter, depending on the input. By specializing each adapter for its specific operation, MemLoRA(-V) achieves performance comparable to models 10-60x larger while enabling efficient local execution without cloud API dependencies.

overview of the approach and our considered setting in Figure 4.1.

Secondly, while recent works have begun exploring multimodal capabilities, the handling of visual information remains predominantly text-centric: images are typically converted into textual descriptions through vision-language models before being stored and retrieved [18, 76], an approach that inevitably loses fine-grained visual details, spatial relationships, and numerical information embedded in charts or diagrams. This text-first paradigm, though computationally practical, fundamentally constrains the systems' ability to reason directly over visual content, limiting their effectiveness in domains where visual information plays a critical role, such as technical documentation, medical imaging, or design workflows. Notably, existing benchmarks for evaluating memory systems—such as LoCoMo [103], which focuses on text-based conversational question answering and event summarization—do not assess multimodal capabilities during inference. Although LoCoMo conversations contain images, the original evaluation relies solely on text-based captions, limiting assessment of native visual understanding. This evaluation gap means that a model's ability to process and reason over visual information directly, rather than through caption intermediaries, remains unmeasured.

In our work, we address both issues by integrating Vision Language Models (VLMs) in these memory-augmented systems, and by augmenting the LoCoMo benchmark with Visual Questions and Answers (VQA) on the conversational images. By doing this, not

only we are able to give native visual capabilities to memory-augmented systems, but we are also able to develop our MemLoRA memory system on small VLMs (SVLMs). For these, a novel expert adapter is further introduced to address the VQA task. Such an approach shows how having specialized adapters, one for each operation, can substitute the need for having massive models, and allow for on-device deployment effectively.

We summarize our contributions as follows. *(i)* We introduce the challenge of accurate on-device memory systems where small language models are used, eliminating reliance on cloud-based infrastructure to preserve privacy. *(ii)* We develop a highly-efficient yet well-performing solution that substantially improves over existing approaches and obtains performance close to that of significantly larger models. *(iii)* We extend memory systems to incorporate Vision Language Models with native visual capabilities and apply our MemLoRA framework to this multimodal setting through a specialized vision expert adapter. *(iv)* We augment the LoCoMo benchmark with challenging Visual Question Answering tasks that require direct image access, demonstrating that our approach achieves strong performance with superior efficiency in multimodal contexts.

4.2 Related Work

Memory-Augmented LLMs. Memory systems have improved LLMs' capabilities in several applications. Foundational approaches such as Memory Networks [145] and RAG [79] introduced external memory integration and document retrieval. More sophisticated systems have been inspired by human cognition and operating systems. Recent innovations like MemoryBank [171], MemGPT [111], Zep [119], and Mem0 [18] incorporate hierarchical memory tiers, session management, and self-improving capabilities, while specialized systems like ReadAgent [76] and A-Mem [152] implement human-inspired organizational principles such as gist memory compression and Zettelkasten-style knowledge networks [59]. Many of these approaches rely on agentic frameworks that orchestrate memory operations through iterative LLM queries for tasks such as memory extraction, updating, and retrieval. However, such methods require multiple queries to the LLM that are computationally expensive to run and do not prioritize on-device deployment scenarios. In our work, we employ specialized expert adapters on small models to perform memory operations locally, drastically reducing computational demands.

Knowledge Distillation with LLMs. Knowledge distillation has evolved into a diverse landscape of techniques aimed at transferring capabilities from powerful teacher models to more efficient student models [**xu2024survey**, **camuffo2025mocha**]. Generation-aware divergence methods address the limitations of traditional forward Kullback–Leibler Divergence (KLD) by introducing variants such as reverse KLD [40] or skew KLD [69], which better handle the challenges of auto-regressive generation while requiring access to the teacher model's internal logits or probability distributions. More recent methods have introduced preference-based frameworks that leverage implicit reward signals [83],

advantage functions [36], or pseudo-preference pairs [162] to guide the student toward generating outputs that align with the teacher's quality standards [68]. These approaches often necessitate either white-box access to the teacher model's internal states or involve multi-stage optimization procedures. Alternatively, output-only distillation methods that operate solely on generated text sequences represent another direction in the literature [22, 31, 132, 162]. Such approaches enable distillation from black-box models, proprietary APIs, or any teacher model regardless of architecture, and allow for direct modification of outputs into desired formats or structures before training. In our work, we found that this simpler solution worked well for distilling knowledge in memory systems, and given the practical advantages it offers, we adopted this approach.

Parameter Efficient Finetuning Methods. Parameter-efficient fine-tuning (PEFT) methods have emerged as a powerful approach for adapting large-scale models to specific tasks and domains, while drastically reducing computational requirements compared to full-model fine-tuning [27, 84]. Notable methods in this category are those that inject trainable modules into the model architecture, such as Low-Rank Adaptation (LoRA) [54] and its derivations [10, 70, 92]. A key advantage of these approaches is that the trained modules can be seamlessly merged and unmerged into the base model weights, eliminating any inference-time latency overhead—a crucial consideration for deployment of large-scale models in production environments—and enabling efficient multi-task setups where a single base model can be dynamically adapted to different tasks or domains simply by swapping the active PEFT module [55]. In our work, we leverage these properties and demonstrate how small (multimodal) language models with PEFT adapters are able to achieve performance on par with larger counterparts by swapping between expert memory adapters, while significantly increasing efficiency and enabling practical on-device deployment.

4.3 Method

In this section, we present the technical details of MemLoRA, our efficient memory system suitable for on-device deployment. We begin by introducing Mem0, the memory system we build upon, and describing its core operations (Section 4.3.1). We then detail our proposed MemLoRA solution, which replaces the LLM in Mem0 with an SLM and memory adapters through knowledge distillation (Section 4.3.2). Finally, we extend our approach to multimodal settings by incorporating vision understanding capabilities, enabling memory systems to process visual information natively (Section 4.3.3).

4.3.1 Preliminaries

Mem0. Mem0 is a memory system enhancing LLM applications with persistent, personalized memory across sessions. Mem0 operates through three main stages:
▪ *Knowledge Extraction.* Given a conversational exchange between a user and an AI assistant, Mem0 uses an extraction prompt to query an LLM f_{θ_L}, parametrized by θ_L. The extraction prompt guides the LLM to identify relevant knowledge, Ω, consisting of facts, preferences, and contextual information worth storing in memory from the dialogue.
▪ *Memory Update.* The extracted knowledge Ω is used to update the memory store M. Mem0 queries f_{θ_L} to determine how new information should be integrated with existing memory M—whether to add new entries (`ADD`), update existing ones (`UPDATE`), or delete outdated information (`DELETE`). This ensures the memory remains relevant and consistent over time.
▪ *Memory-Augmented Generation.* During inference, relevant memories Ω' are retrieved from the memory store M based on semantic similarity to the current query q: $\Omega' \leftarrow \text{FindRelatedKnowledge}(M, q)$. These memories Ω' are then provided in the prompt as additional context to f_{θ_L}, enabling it to generate responses that are consistent with past interactions and personalized to the user.

While effective, this approach requires multiple calls to the LLM f_{θ_L}, making it impractical for on-device deployment where computational efficiency, privacy, and offline functionality are critical.

Low-Rank Adaptation (LoRA). LoRA is a PEFT method to adapt pretrained models by injecting trainable low-rank matrices into specific layers while keeping the original model weights frozen. Given a pretrained weight matrix $W_0 \in \mathbb{R}^{d\times k}$, the LoRA adapter L represents the weight update as the product of two low-rank matrices (A, B):

$$W = W_0 + BA,$$

where $B \in \mathbb{R}^{d\times r}$ and $A \in \mathbb{R}^{r\times k}$ with rank $r \ll \min(d, k)$, and W being the updated weights. During training, only the matrices A and B are updated while W_0 remains frozen. LoRA's parameter efficiency and modularity make it ideal for resource-constrained environments.

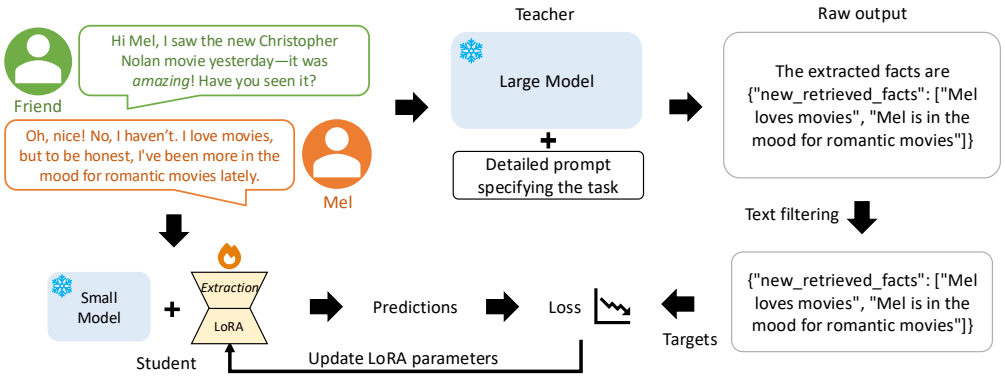

Figure 4.2: **Training Pipeline (*Extraction* LoRA)**. We first generate outputs for the specific memory-related task via a larger model (teacher). Raw output is further cleaned and used as target for training LoRA parameters of a small model (student).

Our proposed on-device memory system, MemLoRA, combines Mem0 and LoRA to support multiple task-specific adapters with minimal overhead.

4.3.2 Our Method: MemLoRA

MemLoRA addresses the deployment challenges of Mem0 systems by replacing the LLM f_{θ_L} with a smaller deployable-on-device model f_{θ_S} parametrized by $\theta_S; |\theta_S| \ll |\theta_L|$ and equipped with multiple specialized memory adapters. Our key insight is that each memory operations—extraction, update, and generation—can be treated as a distinct task amenable to specialized optimization through targeted fine-tuning.

Memory Adapters. Given a small language model f_{θ_S}, we employ LoRA to create lightweight expert memory adapters for each memory operation: L_e, L_u, L_g. The memory adapters are trained via distilling knowledge from the large model f_{θ_L}.

Knowledge Distillation Signal. Rather than distilling soft labels or logits from teacher models into memory adapters, we distill from teacher-generated text outputs $y_T \leftarrow f_{\theta_L}(q)$. We empirically find that training on textual outputs y_T achieves performance close to or exceeding that of teacher models. Such text-based distillation approach offers several practical advantages: *(i)* significant storage reduction compared to saving large logit tensors, *(ii)* flexibility to use student models with different tokenizers than the teacher, and *(iii)* the ability to apply data cleaning and filtering procedures to improve training data quality, and desired outputs, which might differ from base teacher outputs.

Data Preparation. We generate training data by using the teacher model f_{θ_L} on conversational samples from the LoCoMo dataset, then applying operation-specific processing:
▪ *Extraction Adapter.* We train on teacher-generated extractions. We do simple cleaning by removing the "thinking process" of the model, and keeping the minimal `json` form

output.

▪ *Update Adapter.* We observe the teacher model predicts unnecessary `NONE` (i.e., no action) operations for previously retrieved memories rather than focusing solely on newly extracted knowledge. In addition to standard cleaning as before, we filter the training data to process only updates related to new extractions, improving efficiency and focus.

▪ *Generation Adapter.* Leveraging the availability of ground-truth responses in the LoCoMo benchmark, we train directly on these high-quality references rather than teacher-generated outputs, which achieve only 40-50% accuracy on the benchmark. This ensures that the generation expert learns from optimal rather than suboptimal examples.

Training Pipeline. For each expert adapter L_e, L_u, L_g, we: *(i)* generate or prepare training data using the appropriate source (teacher outputs or ground truth), *(ii)* apply operation-specific cleaning and filtering procedures, *(iii)* train the expert adapters using standard next-token prediction with cross-entropy loss, and *(iv)* optimize each adapter independently, enabling specialization without interference.

This process yields three expert adapters: an extraction expert L_e for identifying relevant information from conversations, an update expert L_u for memory management decisions, and a generation expert L_g for producing memory-augmented responses. An illustration of the training pipeline is provided in Figure 4.2.

Inference Pipeline. During deployment, MemLoRA operates identically to Mem0 but dynamically loads the appropriate expert adapter at each stage. The base SLM, f_{θ_S}, switches between memory adapters as needed—extraction expert L_e for knowledge identification, update expert L_u for memory modifications, and generation expert L_g for response creation—maintaining the same three-stage pipeline while drastically reducing computational requirements and enabling fully-local execution.

4.3.3 Native Visual Understanding Capabilities

While language-based memory systems have proven effective for dialogue, real-world conversations often involve visual elements—shared images, screenshots, or visual references. Previous memory systems, including the original Mem0, processed images during the knowledge extraction phase, by using a BLIP captioning model [80] to extract general information about images in the conversation. However, this caption-based approach introduces two critical limitations: *(i)* once images are captioned during extraction, any information not captured in the caption is permanently lost, preventing later queries about visual details, and *(ii)* querying images on-the-fly is not natively supported, requiring a separate model to extract information from them.

Mem0-V. To address these limitations, we extend Mem0 to use Vision Language Models (VLMs). By replacing the earlier foundation model with a VLM, Mem0-V enables *(i)*

native knowledge extraction without requiring a separate image processor, and *(ii)* direct image processing in queries posed to the system, while keeping the remaining pipeline the same. This allows the system to access rich visual information throughout all memory operations rather than relying solely on pre-generated captions.

MemLoRA-V. We extend our efficient solution analogously by replacing the base SLM with a Small Vision Language Model (SVLM), yielding MemLoRA-V with native visual capabilities for on-device deployment. To support visual understanding, we introduce a fourth expert adapter specifically trained on Visual Question Answering (VQA) tasks using images from the LoCoMo dataset. Following our distillation approach for language experts, we train this vision expert L_g^V on output data generated by a larger vision-language teacher model. When MemLoRA-V receives a query about an image, it activates the vision expert adapter L_g^V rather than the language-based one L_g, leveraging specialized visual reasoning capabilities to process the image effectively.

LoCoMo VQA Augmentation. To evaluate these native image understanding capabilities, we recognize that the original LoCoMo questions are insufficient—they can often be answered using captions alone or do not require visual reasoning at all. Therefore, we create a novel VQA benchmark that augments LoCoMo with challenging visual questions about images already present in the dataset. To automate the creation of these challenging questions and ground-truth answers, we employ InternVL3-78B [174], one of the strongest open-source VLMs available at the time of development. We design questions to be "challenging" and "not ambiguous" by instructing the model to generate queries following three types: (a) counting object quantities, (b) identifying colors of specific image regions, and (c) asking about unusual objects in the scene, as illustrated in Figure 4.3. These question types were selected after evaluating eight alternatives, where a validator model (InternVL3-2B) attempted to answer each type. The three types that resulted in the highest error rates were chosen to construct our benchmark, ensuring the task requires genuine visual reasoning.

While most VQA benchmarks use open-ended questions [29, 77], this format requires resource-intensive LLM-as-a-judge approaches to reliably assess answer correctness [166]. To enable efficient evaluation, we design our questions to be easily assessable. Specifically, we instruct InternVL3-78B to generate questions answerable in one word and structure responses accordingly as: `"answer": "<one-word-answer>"` and `"reason": "<explanation>"`. The one-word answer enables evaluation using word similarity metrics, eliminating the ambiguity inherent in free-form responses. The reason field accommodates VLMs' natural tendency to explain their reasoning, making the format more aligned with how these models generate outputs. During supervised fine-tuning of the SVLM vision expert adapter, we leverage both fields to provide a richer training signal, which improves the expert's visual reasoning capabilities beyond what the answer alone would provide.

Figure 4.3: **Our augmentation of LoCoMo** includes challenging VQA tasks about (a) counting object quantities, (b) identifying colours, and (c) asking about unusual objects.

In summary, we introduce three key contributions for multimodal memory systems: Mem0-V, which extends the original Mem0 memory system with native VLM capabilities; MemLoRA-V, our efficient on-device variant with a specialized vision expert adapter; and a novel VQA benchmark augmentation for LoCoMo that enables rigorous evaluation of visual reasoning in memory-augmented systems.

4.4 Experiments

In this section, we evaluate our proposed method MemLoRA on memory-augmented dialogue and multimodal conversation understanding tasks, comparing its performance against Mem0 baselines of varying sizes. We demonstrate that MemLoRA achieves competitive performance with significantly larger models while providing massive improvements in computational efficiency for on-device deployment.

4.4.1 Experimental Setup

To evaluate MemLoRA's performance, we integrate it within the Mem0 memory system [18] and follow the same evaluation setup. Specifically, we utilize the Question Answering (QA) task of the LoCoMo benchmark [103] to assess long-term conversational memory in AI agents. This benchmark features 10 extended, multi-session dialogues, each with hundreds of turns, and includes questions categorized as single-hop, multi-hop, temporal, and open-domain. In the context of Mem0 and our method, the evaluation measures the ability of different LLMs to *(i)* extract useful knowledge from conversational data, *(ii)* update memory storage with necessary information, and *(iii)* correctly utilize the retrieved memory context. In our VLM-integrated benchmark, we further introduce a new VQA task, where the model is asked three challenging types of questions on each image present in the conversation, evaluating the model's performance in assisting with visual data.

Data Split. Given the necessity of training our expert adapters to perform the different memory operations, we split the LoCoMo dataset into training, validation, and test sets, following an approximate 70-10-20% split respectively. To prevent data leakage and ensure valid evaluation, we keep entire conversations together within each split. All results reported in our benchmark tables are computed on the held-out 20% test split, on which no hyperparameter tuning was performed.

Metrics. The experimental setup measures performance using two separate metrics: a summary over *lexical* metrics (L)–measured as the average of ROUGE-1 [86], METEOR [7], BERTScore-F1 [165] and SentenceBERT [120]–and LLM-as-a-Judge (J). The former is used to quickly help validating experiments, and to measure similarity with ground-truth answers, while the latter is used as the primary metric, being arguably better at measuring factual accuracy [18, 58]. To allow for reproducibility over time, we do not use API-based models as the evaluator model (Judge), but rather GPT-OSS-120B [1], being one of the most capable open-source models that can fit on a single A100-80GB-GPU. The metric for the VQA task, denoted as V, is the average matching between the predicted one-word answers from the tested model against the ones generated by InternVL3-78B [174] when creating the dataset.

Table 4.1: **Comparison of MemLoRA against Mem0 on LoCoMo.** Evaluation shows average score over lexical metrics (L) and LLM-as-a-judge (J). ΔJ^{base} measures the relative improvement with respect to the base SLM.By equipping 1.5B/2B SLMs with memory adapters, MemLoRA surpasses 27B models, reaching comparable results to 120B ones.

LLM	*KD teacher*	L	J	ΔJ^{base}
Gemma2-27b	-	38.6	39.1	-
GPT-oss-120b	-	38.9	48.9	-
Qwen2.5-1.5b	-	30.5	29.6	-
+Exp (ours)	Gemma2-27b	37.3	36.9	+25%
+Exp (ours)	GPT-oss-120b	38.4	42.1	+42%
Gemma2-2b	-	29.1	24.9	-
+Exp (ours)	Gemma2-27b	**44.5**	**47.2**	+90%
+Exp (ours)	GPT-oss-120b	<u>42.7</u>	<u>44.6</u>	+79%

4.4.2 Benchmark Results

We compare our MemLoRA approach with Mem0, which are both powered by open-source locally-downloaded models for fair comparison and reproducibility.

Language-only Memory Systems. In the setup with language models utilization, we test Mem0 with different baseline models: two large language models (LLMs), namely Gemma2-27B [133] and GPT-OSS-120B [1], and two small language models (SLMs), namely Qwen2.5-1.5B [134] and Gemma2-2B [133]. We test our MemLoRA by equipping memory adapters to the two SLMs, powered via knowledge distillation from teachers' data. Table 4.1 presents these results, showing MemLoRA surpasses the Gemma2-27b baseline by a significant margin on three student-teacher combinations out of four. Here, the leading MemLoRA variant, with Gemma2-2B fine-tuned using Gemma2-27B generated data, achieves a J score of 47.2, much larger than 39.1 of Gemma2-27B, and comparable to 48.9 of GPT-OSS-120B.

Vision-Language-integrated Memory Systems. In our novel Vision-Language integration within the memory system, we compare our VLM-integrated Mem0-V with our VLM-integrated MemLoRA-V. We evaluate these models in both the standard QA task from LoCoMo, and on our newly introduced VQA task. As small VLMs, we use InternVL3-1B and InternVL3-2B [174] equipped with one adapter trained on text-only QA as before, and a new adapter trained on VQAs with images from the training set. To highlight the abilities of VLM-integrated systems, we also compare these methods with text-only Mem0 vision baselines. For this case, we adapt the VQA tasks to use text coming from BLIP [80] captions, as utilized by Mem0 in the extraction stage.

Table 4.2 presents these results. Interestingly, in the text-only QA task, our MemLoRA-V applied on InternVL3-2B and InternVL3-1B, surpasses larger text-only models such as Gemma2-27B. At the same time, in the VQA task, we observe significant improvements for

Table 4.2: **Comparison of MemLoRA-V and Mem0-V, as well as the original Mem0, on LoCoMo benchmark and newly introduced VQA task.** Evaluation done in terms of lexical metrics (*L*), LLM-as-a-judge (*J*), and accuracy in our VQA task (*V*). G-27 stands for Gemma2-27B, IVL3-78B stands for InternVL3-78B. Notice how by training specialized adapters on both tasks, Mem0-V is able to achieve strong performance in both, while keeping resource utilization low. * LLM-based Mem0 baselines, utilize BLIP extracted captions as contextual information on the images.

LLM/VLM	*KD teacher*	*L*	*J*	*V*
Gemma2-27b	-	38.6	39.1	23.7*
GPT-oss-120b	-	38.9	48.9	22.0*
InternVL3-1B	-	13.7	9.0	50.0
+Exp (ours)	G-27B\IVL3-78B	29.1	20.2	69.4
InternVL3-2B	-	32.2	27.0	70.8
+Exp (ours)	G-27B\IVL3-78B	**44.6**	**40.3**	**81.3**

Table 4.3: **Comparison of MemLoRA (purple) and Mem0 in terms of efficiency.** Under the same computational resources, MemLoRA requires 10-20× smaller memory and deliveres 10-20× faster responses with respect to LLM-powered Mem0, while achieving comparable performance

LLM	*size(GB)*	*tok/s*↑	*tok/ans*↓	*s/ans*↓
Gemma2-27b	50.71	9.2	97.63	10.66
GPT-oss-120b	60.77	11.4	209.91	22.82
Qwen2.5-1.5b	2.88	71.0	54.74	0.77
+Exp (ours)	2.92	71.0	**45.26**	**0.64**
Gemma2-2b	4.87	47.4	33.13	0.70
+Exp (ours)	4.92	47.4	**32.73**	**0.69**

these VLMs with dedicated adapters, increasing *V* score from 50.0 to 69.4 for InternVL3-1B, and from 70.8 to 81.3 for InternVL3-2B. In contrast, Mem0 that only uses text-based BLIP captions performs significantly worse than these VLM-integrated variants, reaching a highest value of 23.7, showing one limitation of language-only systems.

4.4.3 Efficiency Measures

One main advantage of MemLoRA is its efficient deployment capability. Specifically, compared to API-based memory systems that rely on cloud-hosted large language models, MemLoRA enables fully local execution with significantly reduced computational requirements, lower latency, and no dependency on network connectivity. By replacing a single large LLM with specialized lightweight adapters on small language models, our solution drastically reduces memory footprint, and inference time—critical factors for on-device deployment scenarios such as mobile applications, edge devices, and privacy-sensitive environments.

In Table 4.3 we report efficiency measures of MemLoRA compared with Mem0 baselines utilizing LLMs of different sizes. Specifically, we report model sizes and operational measures such as tokens per second (*tok/s*), tokens per LLM answer (*tok/ans*), and seconds per answer (*s/ans*). These latter measures are obtained by averaging over all three memory stages of *knowledge extraction*, *memory update*, and *memory-augmented generation*, while operating to a portion of the LoCoMo benchmark. We calculate these metrics by averaging over multiple runs, maintaining the setup unaltered. In standard Mem0, deploying larger models on-device yields strong performance but results in 10-30x slower inference, whereas using smaller models improves efficiency but compromises accuracy, highlighting a fundamental performance-efficiency trade-off. MemLoRA bridges this gap, matching the performance of significantly larger models while retaining the efficiency of small models through task-specialized expert adapters. Furthermore, compared to base SLMs, by formatting their output to match the memory utilization, we are able to reduce the number of tokens per answer, further reducing the operational time of the memory system.

4.4.4 Ablations

We validate our design choices via two comprehensive ablations: (i) we study the contribution of each memory adapter at different stages of the memory pipeline; and (ii) we study the impact of student model size on overall performance.

Per-stage Incremental Performance. To isolate the contribution of each expert adapter, in Table 4.4 we conduct a stage-wise ablation study evaluating performance improvements at each memory operation. We measure the impact of our specialized adapters for knowledge extraction, memory update, and memory-augmented generation independently. In the extraction and update stages, MemLoRA demonstrates strong performance even when trained on data generated by Gemma2-27b, with the trained experts showing notable

Table 4.4: **Ablation of MemLoRA adapters (+Exp) for each operation**, comparing Gemma2-2B (G-2b) equipped with experts against its teacher Gemma2-27B (G-27b). In *extraction* and *update* stages, MemLoRA shows stronger performance than the teacher, being trained on filtered teacher-generated data. In *generation*, specialization on the QA task yields the largest gain, with the expert largely surpassing the teacher model (47.2 vs. 39.1).

extraction	*update*	*generation*	L	J	ΔJ^{prev}
G-2b	G-2b	G-2b	29.1	24.9	-
G-27b	G-2b	G-2b	32.7	30.9	+24%
G-27b	G-27b	G-2b	34.7	34.8	**+13%**
G-27b	G-27b	G-27b	38.6	39.1	+12%
G-2b+Exp	G-2b	G-2b	32.9	32.2	**+29%**
G-2b+Exp	G-2b+Exp	G-2b	35.1	35.6	+11%
G-2b+Exp	G-2b+Exp	G-2b+Exp	44.5	47.2	**+33%**

Table 4.5: **Ablation evaluating the effect of MemLoRA at different students' scales.** As expected, we find that the smallest models lead to the largest improvements, while we see diminishing improvements as the students' size increases.

LLM	*KD teacher*	*L*	*J*	ΔJ^{base}
Qwen2.5-0.5b	-	19.5	11.2	-
+Exp (ours)	Gemma2-27b	28.1	26.6	+138%
Qwen2.5-1.5b	-	30.5	29.6	-
+Exp (ours)	Gemma2-27b	37.3	36.9	+25%
Qwen2.5-3b	-	39.9	35.6	-
+Exp (ours)	Gemma2-27b	42.3	42.1	+18%

robustness across different conversational contexts. Most significantly, in the generation stage, specializing the adapter directly on the QA task yields the largest performance gain, with our generation expert achieving a *J* score of 47.2 compared to the teacher model's 39.1. This substantial improvement—surpassing the teacher by 8.1 points—demonstrates that task-specific specialization through dedicated memory adapters can not only match but exceed the capabilities of general-purpose larger models, particularly when trained on high-quality ground-truth data.

Student's Performance at Different Scales. To understand how student model capacity affects our approach, we evaluate MemLoRA across multiple model sizes in Table 4.5, ranging from compact models for resource-constrained devices to moderately-sized alternatives. Our results reveal that increasing the student model size initially yields substantial performance improvements, with gains progressively decreasing as models grow larger.

4.5 Conclusions

In this work, we introduced MemLoRA, a novel memory system enabling efficient on-device deployment of memory-augmented systems through specialized memory adapters on small models. By treating each memory operation as a distinct task, we demonstrate that lightweight adapters achieve performance comparable to models 10-60x larger while drastically reducing computational requirements and enabling local execution. Our evaluation on the LoCoMo benchmark validates this approach. Our ablation studies reveal that memory experts surpass teacher models and that performance exhibits diminishing returns with increasing student model size. We extend our approach to multimodal settings with MemLoRA-V, the first memory-augmented system featuring native visual understanding via a specialized vision expert adapter. To assess this, we enhanced LoCoMo with challenging VQA tasks, establishing a new benchmark for multimodal memory-augmented systems. Our results show that lightweight, specialized memory systems can effectively replace large cloud-based counterparts, enabling privacy-preserving and efficient deployment on mobile and edge platforms.

5

Conclusions and Discussion

This thesis addresses the fundamental challenge of deploying large-scale foundation models in resource-constrained environments through three interconnected contributions that advance parameter-efficient fine-tuning as both an adaptation and deployment paradigm.

Our investigation began by questioning established assumptions about what makes model adaptation robust. Through empirical analysis, we discovered that effective fine-tuning depends not on preserving hyperspherical energy, as previously hypothesized, but on bounding Frobenius norm deviation from pretrained weights. This insight led to *ETHER*, a parameter-efficient method based on hyperplane reflections that achieves strong non-deteriorating properties with reduced computational cost. However, *ETHER*'s fixed rank and predetermined boundaries revealed the need for greater flexibility.

Building on this foundation, we introduced DeLoRA, which makes Frobenius boundaries explicit and learnable while achieving LoRA-like efficiency. By decoupling angular and magnitude learning, DeLoRA prevents divergence from the underlying model, enabling stable adaptation at aggressive learning rates, while supporting arbitrary rank selection and efficient computation. This positions PEFT methods as robust, flexible building blocks suitable for diverse deployment scenarios requiring continual adaptation.

Finally, we demonstrated that parameter-efficient adaptation can transcend its traditional role when powered by knowledge distillation. Through MemLoRA, we showed that lightweight task-specific expert adapters paired with a compact student model can replace large foundation models entirely, achieving equivalent performance with dramatically reduced memory footprint, faster inference, and minimal deployment cost. This paradigm shift—from adapting large models to deploying compact models with specialist adapters—addresses the fundamental deployment bottleneck.

Collectively, these contributions establish that the path to practical deployment of foundation models lies not in making large models slightly more efficient, but in fundamentally rethinking their architecture: discovering the principles that make adaptation robust, building methods that embody these principles efficiently, and ultimately enabling deployment paradigms where compact models paired with lightweight experts replace massive generalist systems. As foundation models continue to grow in capability and

size, such approaches become increasingly critical for democratizing access to advanced AI systems in real-world, resource-constrained applications.

Implications and Future Directions The contributions of this thesis extend beyond their immediate technical applications, with significant implications for deploying foundation models in resource-constrained environments such as edge devices, personalized AI systems, and multi-task architectures. We discuss these broader implications and identify promising directions for extending the principles of robust, efficient adaptation to new domains and challenges.

Continual Learning Systems. The advancments on Frobenius norm constraints preventing knowledge deterioration suggests new approaches to continual learning where bounded transformations preserve knowledge across sequential task adaptations. Investigating whether learnable boundaries can dynamically adjust across task sequences could enable systems that balance plasticity with stability without complex regularization schemes.

Layer-Specific Adaptation. Our work treats Frobenius norm constraints uniformly across layers, yet different layers may require different deviation bounds depending on their role in the network hierarchy. Exploring layer-specific Frobenius distances and investigating how different tasks naturally focus adaptation on different layers could lead to more efficient and targeted specialization strategies.

Cross-Modal Generalization. While this thesis focuses on vision and language domains, the principles likely generalize to audio processing, scientific computing, robotics control, and other fields with similar adaptation challenges. Investigating how these methods transfer to domains with strict resource constraints or safety requirements would extend their practical impact.

Conclusive Remarks The challenge of deploying foundation models efficiently while maintaining their capabilities remains central to making AI systems practical and accessible. This thesis demonstrates that efficiency and robustness need not be competing objectives—by understanding the geometric principles underlying robust adaptation, we can design methods that are simultaneously more efficient, more stable, and more effective at preserving pretrained knowledge. Moreover, we show how parameter-efficient methods, acting as specialization modules, enable a practical deployment paradigm where lightweight expert adapters—when enhanced through knowledge distillation and paired with compact student models—can replace large foundation models in specialized task scenarios. We hope this work inspires continued research at the intersection of parameter-efficient adaptation, model compression, and robust optimization, bringing us closer to foundation models that are not only powerful but also deployable and accessible across diverse computational environments.

Bibliography

[1] Sandhini Agarwal et al. "gpt-oss-120b gpt-oss-20b Model Card." In: *CoRR* abs/2508.10925 (2025). URL: http://dblp.uni-trier.de/db/journals/corr/corr2508.html#abs-2508-10925 (cit. on pp. 55, 56).

[2] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. *Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning*. arXiv:2012.13255 [cs]. 2020-12. DOI: 10.48550/arXiv.2012.13255. URL: http://arxiv.org/abs/2012.13255 (visited on 2023-11-16) (cit. on pp. 1, 5).

[3] Lightning AI. *LitGPT*. https://github.com/Lightning-AI/litgpt. 2023 (cit. on p. 92).

[4] Anthropic. "System Card: Claude Sonnet 4.5". In: (2025). URL: https://www.anthropic.com/claude-sonnet-4-5-system-card (cit. on p. 1).

[5] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations." In: *NeurIPS*. Ed. by Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. 2020. URL: http://dblp.uni-trier.de/db/conf/nips/neurips2020.html#BaevskiZMA20 (cit. on p. 3).

[6] Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, Yichen Feng, Kelin Fu, Bofei Gao, Hongcheng Gao, Peizhong Gao, Tong Gao, Xinran Gu, Longyu Guan, Haiqing Guo, Jianhang Guo, Hao Hu, Xiaoru Hao, Tianhong He, Weiran He, Wenyang He, Chao Hong, Yangyang Hu, Zhenxing Hu, Weixiao Huang, Zhiqi Huang, Zihao Huang, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yongsheng Kang, Guokun Lai, Cheng Li, Fang Li, Haoyang Li, Ming Li, Wentao Li, Yanhao Li, Yiwei Li, Zhaowei Li, Zheming Li, Hongzhan Lin, Xiaohan Lin, Zongyu Lin, Chengyin Liu, Chenyu Liu, Hongzhang Liu, Jingyuan Liu, Junqi Liu, Liang Liu, Shaowei Liu, T. Y. Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yibo Liu, Yiping Liu, Yue Liu, Zhengying Liu, Enzhe Lu, Lijun Lu, Shengling Ma,

Xinyu Ma, Yingwei Ma, Shaoguang Mao, Jie Mei, Xin Men, Yibo Miao, Siyuan Pan, Yebo Peng, Ruoyu Qin, Bowen Qu, Zeyu Shang, Lidong Shi, Shengyuan Shi, Feifan Song, Jianlin Su, Zhengyuan Su, Xinjie Sun, Flood Sung, Heyi Tang, Jiawen Tao, Qifeng Teng, Chensi Wang, Dinglu Wang, Feng Wang, and Haiming Wang. "Kimi K2: Open Agentic Intelligence." In: *CoRR* abs/2507.20534 (2025). URL: http://dblp.uni-trier.de/db/journals/corr/corr2507.html#abs-2507-20534 (cit. on p. 3).

[7] Satanjeev Banerjee and Alon Lavie. "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments." In: *IEEvaluation@ACL*. Ed. by Jade Goldstein, Alon Lavie, Chin-Yew Lin, and Clare R. Voss. Association for Computational Linguistics, 2005, pp. 65–72. URL: http://dblp.uni-trier.de/db/conf/acl/ieevaluation2005.html#BanerjeeL05 (cit. on p. 55).

[8] Yoshua Bengio, Aaron Courville, and Pascal Vincent. "Representation Learning: A Review and New Perspectives". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 35.8 (2013-08), 1798–1828. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2013.50. URL: https://doi.org/10.1109/TPAMI.2013.50 (cit. on p. 1).

[9] Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John Patrick Cunningham. "LoRA Learns Less and Forgets Less". In: *TMLR*. 2024 (cit. on pp. 4, 30, 31, 42).

[10] Massimo Bini, Leander Girrbach, and Zeynep Akata. "Decoupling Angles and Strength in Low-rank Adaptation". In: *International Conference on Learning Representations (ICLR)*. 2025 (cit. on pp. 49, 101).

[11] Massimo Bini, Karsten Roth, Zeynep Akata, and Anna Khoreva. "ETHER: Efficient Finetuning of Large-Scale Models with Hyperplane Reflections". In: *ICML*. 2024 (cit. on pp. 30–32, 34, 36, 38, 39, 41, 97, 101).

[12] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ B. Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri S. Chatterji, Annie S. Chen, Kathleen Creel, Jared Quincy Davis, Dorottya Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah D. Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark S. Krass, Ranjay Krishna, Rohith Kuditipudi, and et al. "On the Opportunities and Risks of Foundation Models." In: *CoRR* abs/2108.07258 (2021). URL: http://dbl

p.uni-trier.de/db/journals/corr/corr2108.html#abs-2108-07258 (cit. on pp. 3, 5, 12).

[13] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. *Language Models are Few-Shot Learners*. 2020. arXiv: 2005.14165 [cs.CL]. URL: https://arxiv.org/abs/2005.14165 (cit. on p. 3).

[14] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. "Emerging properties in self-supervised vision transformers". In: *CVPR*. 2021 (cit. on pp. 23, 36).

[15] Jiaao Chen, Aston Zhang, Xingjian Shi, Mu Li, Alex Smola, and Diyi Yang. "Parameter-Efficient Fine-Tuning Design Spaces". In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=XSRSWxyJIC (cit. on p. 12).

[16] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. "A simple framework for contrastive learning of visual representations". In: *Proceedings of the 37th International Conference on Machine Learning*. ICML'20. JMLR.org, 2020 (cit. on pp. 2, 3).

[17] Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. "LongLoRA: Efficient Fine-tuning of Long-Context Large Language Models". In: *The International Conference on Learning Representations (ICLR)*. 2024 (cit. on p. 14).

[18] Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. *Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory*. 2025. arXiv: 2504.19413 [cs.CL]. URL: https://arxiv.org/abs/2504.19413 (cit. on pp. 46–48, 55).

[19] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. *Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality*. 2023. URL: https://lmsys.org/blog/2023-03-30-vicuna/ (cit. on p. 12).

[20] Alexandra Chronopoulou, Matthew Peters, Alexander Fraser, and Jesse Dodge. "AdapterSoup: Weight Averaging to Improve Generalization of Pretrained Language Models". In: *Findings of the Association for Computational Linguistics: EACL 2023*. Ed. by Andreas Vlachos and Isabelle Augenstein. Dubrovnik, Croatia: Association for Computational Linguistics, 2023-05, pp. 2054–2063. DOI: 10.18653/v

1/2023.findings-eacl.153. URL: https://aclanthology.org/2023.findings-eacl.153/ (cit. on p. 5).

[21] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. "Think you have solved question answering? try arc, the ai2 reasoning challenge". In: *arXiv*. 2018 (cit. on pp. 25, 37).

[22] Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Fang Zeng, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. "AugGPT: Leveraging ChatGPT for Text Data Augmentation." In: *IEEE Trans. Big Data* 11.3 (2025), pp. 907–918. URL: http://dblp.uni-trier.de/db/journals/tbd/tbd11.html#DaiLLHCWZXZLLLZCSLSLL25 (cit. on p. 49).

[23] DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, and Wangding Zeng. "DeepSeek-V3 Technical Report." In: *CoRR* abs/2412.19437 (2024). URL: http://dblp.uni-trier.de/db/journals/corr/corr2412.html#abs-2412-19437 (cit. on p. 3).

[24] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. "QLORA: efficient finetuning of quantized LLMs". In: *Proceedings of the 37th International Conference on Neural Information Processing Systems*. NIPS '23. New Orleans, LA, USA: Curran Associates Inc., 2023 (cit. on pp. 4, 5, 14).

[25] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, 2019-06, pp. 4171–4186. DOI:

10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423 (cit. on pp. 2, 3, 25).

[26] Ning Ding, Xingtai Lv, Qiaosen Wang, Yulin Chen, Bowen Zhou, Zhiyuan Liu, and Maosong Sun. "Sparse Low-rank Adaptation of Pre-trained Language Models". In: *EMNLP*. 2023 (cit. on p. 42).

[27] Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, Jing Yi, Weilin Zhao, Xiaozhi Wang, Zhiyuan Liu, Hai-Tao Zheng, Jianfei Chen, Yang Liu, Jie Tang, Juanzi Li, and Maosong Sun. "Parameter-efficient fine-tuning of large-scale pre-trained language models." In: *Nat. Mac. Intell.* 5.3 (2023), pp. 220–235. URL: http://dblp.uni-trier.de/db/journals/natmi/natmi5.html#DingQYWYSHCCCYZWLZCLTLS23 (cit. on p. 49).

[28] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=YicbFdNTTy (cit. on pp. 1, 3).

[29] Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, Dahua Lin, and Kai Chen. "VLMEvalKit: An Open-Source ToolKit for Evaluating Large Multi-Modality Models". In: *Proceedings of the 32nd ACM International Conference on Multimedia*. MM '24. Melbourne VIC, Australia: Association for Computing Machinery, 2024, 11198–11201. ISBN: 9798400706868. DOI: 10.1145/3664647.3685520. URL: https://doi.org/10.1145/3664647.3685520 (cit. on p. 53).

[30] Abhimanyu Dubey et al. *The Llama 3 Herd of Models*. 2024. arXiv: 2407.21783 [cs.AI]. URL: https://arxiv.org/abs/2407.21783 (cit. on pp. 3, 6).

[31] Benjamin Feuer and Chinmay Hegde. "WILDCHAT-50M: A Deep Dive Into the Role of Synthetic Data in Post-Training." In: *CoRR* abs/2501.18511 (2025). URL: http://dblp.uni-trier.de/db/journals/corr/corr2501.html#abs-2501-18511 (cit. on p. 49).

[32] Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. "Born-Again Neural Networks." In: *ICML*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 1602–1611. URL: http://dblp.uni-trier.de/db/conf/icml/icml2018.html#FurlanelloLTIA18 (cit. on p. 6).

[33] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. *An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion*. 2022. arXiv: 2208.01618 [cs.CV] (cit. on p. 14).

[34] Chongyang Gao, Kezhen Chen, Jinmeng Rao, Ruibo Liu, Baochen Sun, Yawen Zhang, Daiyi Peng, Xiaoyuan Guo, and Vs Subrahmanian. "MoLA: MoE LoRA with Layer-wise Expert Allocation". In: *Findings of the Association for Computational Linguistics: NAACL 2025*. Ed. by Luis Chiruzzo, Alan Ritter, and Lu Wang. Albuquerque, New Mexico: Association for Computational Linguistics, 2025-04, pp. 5097–5112. ISBN: 979-8-89176-195-7. DOI: 10.18653/v1/2025.findings-naacl.284. URL: https://aclanthology.org/2025.findings-naacl.284/ (cit. on p. 5).

[35] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. *A framework for few-shot language model evaluation*. Version v0.4.0. 2023-12. DOI: 10.5281/zenodo.10256836. URL: https://zenodo.org/records/10256836 (cit. on p. 92).

[36] Shiping Gao, Fanqi Wan, Jiajian Guo, Xiaojun Quan, and Qifan Wang. "Advantage-Guided Distillation for Preference Alignment in Small Language Models." In: *ICLR*. OpenReview.net, 2025. URL: http://dblp.uni-trier.de/db/conf/iclr/iclr2025.html#GaoWGQW25 (cit. on p. 49).

[37] Ziqi Gao, Qichao Wang, Aochuan Chen, Zijing Liu, Bingzhe Wu, Liang Chen, and Jia Li. "Parameter-Efficient Fine-Tuning with Discrete Fourier Transform". In: *ICML*. 2024 (cit. on p. 42).

[38] Saurabh Garg, Mehrdad Farajtabar, Hadi Pouransari, Raviteja Vemulapalli, Sachin Mehta, Oncel Tuzel, Vaishaal Shankar, and Fartash Faghri. "TiC-CLIP: Continual Training of CLIP Models". In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=TLADT8Wrhn (cit. on p. 12).

[39] Henry Gouk, Timothy M. Hospedales, and Massimiliano Pontil. *Distance-Based Regularisation of Deep Networks for Fine-Tuning*. arXiv:2002.08253 [cs, stat]. 2021-01. DOI: 10.48550/arXiv.2002.08253. URL: http://arxiv.org/abs/2002.08253 (visited on 2023-11-16) (cit. on p. 12).

[40] Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. "MiniLLM: Knowledge Distillation of Large Language Models." In: *ICLR*. OpenReview.net, 2024. URL: http://dblp.uni-trier.de/db/conf/iclr/iclr2024.html#Gu0WH24 (cit. on p. 48).

[41] Demi Guo, Alexander Rush, and Yoon Kim. "Parameter-Efficient Transfer Learning with Diff Pruning". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Ed. by Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli. Online: Association for Computational Linguistics, 2021-08, pp. 4884–4896. DOI: 10.18653/v1/2021.acl-long.378. URL: https://aclanthology.org/2021.acl-long.378/ (cit. on pp. 5, 14).

[42] Xu Han, Zhengyan Zhang, Ning Ding, Yuxian Gu, Xiao Liu, Yuqi Huo, Jiezhong Qiu, Liang Zhang, Wentao Han, Minlie Huang, Qin Jin, Yanyan Lan, Yang Liu, Zhiyuan Liu, Zhiwu Lu, Xipeng Qiu, Ruihua Song, Jie Tang, Ji-Rong Wen, Jinhui Yuan, Wayne Xin Zhao, and Jun Zhu. "Pre-Trained Models: Past, Present and Future." In: *CoRR* abs/2106.07139 (2021). URL: http://dblp.uni-trier.de/db/journals/corr/corr2106.html#abs-2106-07139 (cit. on pp. 1, 3).

[43] Zeyu Han, Chao Gao, Jinyang Liu, Sai Qian Zhang, et al. "Parameter-efficient fine-tuning for large models: A comprehensive survey". In: *arXiv*. 2024 (cit. on pp. 1, 42).

[44] Soufiane Hayou, Nikhil Ghosh, and Bin Yu. "The Impact of Initialization on LoRA Finetuning Dynamics". In: *arXiv*. 2024 (cit. on p. 42).

[45] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. "Towards a Unified View of Parameter-Efficient Transfer Learning." In: *ICLR*. OpenReview.net, 2022. URL: http://dblp.uni-trier.de/db/conf/iclr/iclr2022.html#HeZMBN22 (cit. on p. 5).

[46] Pengcheng He, Jianfeng Gao, and Weizhu Chen. "DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing". In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=sE7-XhLxHA (cit. on pp. 25, 91).

[47] Jeffrey Heaton. "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning: The MIT Press, 2016, 800 pp, ISBN: 0262035618". In: *Genetic Programming and Evolvable Machines* 19 (2017-10). DOI: 10.1007/s10710-017-9314-z (cit. on pp. 1, 2).

[48] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. "Measuring Massive Multitask Language Understanding". In: *ICLR*. 2021 (cit. on pp. 25, 37).

[49] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. *GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium*. 2018. arXiv: 1706.08500 [cs.LG] (cit. on p. 24).

[50] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. *Distilling the Knowledge in a Neural Network*. cite arxiv:1503.02531Comment: NIPS 2014 Deep Learning Workshop. 2015. URL: http://arxiv.org/abs/1503.02531 (cit. on p. 6).

[51] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. "Parameter-efficient transfer learning for NLP". In: *ICML*. 2019 (cit. on pp. 3, 4, 12, 14, 30, 40, 42).

[52] Alston S. Householder. "Unitary Triangularization of a Nonsymmetric Matrix". In: *J. ACM* 5.4 (1958), 339–342. ISSN: 0004-5411. DOI: 10.1145/320941.320947. URL: https://doi.org/10.1145/320941.320947 (cit. on p. 12).

[53] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. *(LoRA) LoRA: Low-Rank Adaptation of Large Language Models*. en. arXiv:2106.09685 [cs]. 2021-10. URL: http://arxiv.org/abs/2106.09685 (visited on 2023-06-21) (cit. on pp. 3, 4).

[54] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. "LoRA: Low-Rank Adaptation of Large Language Models". In: *ICLR*. 2022. URL: https://openreview.net/forum?id=nZeVKeeFYf9 (cit. on pp. 12, 14, 15, 25, 30, 31, 39–42, 49, 92, 96).

[55] Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. "LoraHub: Efficient Cross-Task Generalization via Dynamic LoRA Composition." In: *CoRR* abs/2307.13269 (2023). URL: http://dblp.uni-trier.de/db/journals/corr/corr2307.html#abs-2307-13269 (cit. on pp. 4, 49).

[56] Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats L. Richter, Quentin Anthony, Timothée Lesort, Eugene Belilovsky, and Irina Rish. *Simple and Scalable Strategies to Continually Pre-train Large Language Models*. 2024. arXiv: 2403.08763 [cs.LG] (cit. on p. 12).

[57] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo,

Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, and Ilge Akkaya. "OpenAI o1 System Card." In: *CoRR* abs/2412.16720 (2024). URL: http://dblp.uni-trier.de/db/journals/corr/corr2412.html#abs-2412-16720 (cit. on p. 1).

[58] Denis Janiak, Jakub Binkowski, Albert Sawczyn, Bogdan Gabrys, Ravid Shwartz-Ziv, and Tomasz Kajdanowicz. "The illusion of progress: Re-evaluating hallucination detection in LLMs." In: *CoRR* abs/2508.08285 (2025). URL: http://dblp.uni-trier.de/db/journals/corr/corr2508.html#abs-2508-08285 (cit. on p. 55).

[59] David Kadavy. *Digital Zettelkasten: Principles, Methods, & Examples*. Google Books, 2021 (cit. on p. 48).

[60] Dhiraj Kalamkar, Dheevatsa Mudigere, Naveen Mellempudi, Dipankar Das, Kunal Banerjee, Sasikanth Avancha, Dharma Teja Vooturi, Nataraj Jammalamadaka, Jianyu Huang, Hector Yuen, Jiyan Yang, Jongsoo Park, Alexander Heinecke, Evangelos Georganas, Sudarshan Srinivasan, Abhisek Kundu, Misha Smelyanskiy, Bharat Kaul, and Pradeep Dubey. *A Study of BFLOAT16 for Deep Learning Training*. 2019. arXiv: 1905.12322 [cs.LG] (cit. on p. 25).

[61] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. "Scaling Laws for Neural Language Models." In: *CoRR* abs/2001.08361 (2020). URL: https://arxiv.org/pdf/2001.08361.pdf (cit. on p. 3).

[62] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. *Progressive Growing of GANs for Improved Quality, Stability, and Variation*. 2018. arXiv: 1710.10196 [cs.NE] (cit. on p. 90).

[63] Shyamgopal Karthik, Karsten Roth, Massimiliano Mancini, and Zeynep Akata. *If at First You Don't Succeed, Try, Try Again: Faithful Diffusion-based Text-to-Image Generation by Selection*. 2023. arXiv: 2305.13308 [cs.CV] (cit. on p. 14).

[64] Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. "Continual Pre-training of Language Models". In: *The Eleventh International Conference on Learning Representations*. 2023 (cit. on pp. 4, 12).

[65] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization." In: *ICLR (Poster)*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://dblp.uni-trier.de/db/conf/iclr/iclr2015.html#KingmaB14 (cit. on p. 4).

[66] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. "Segment Anything". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 4015–4026 (cit. on p. 12).

[67] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. "Overcoming catastrophic forgetting in neural networks". In: *Proceedings of the National Academy of Sciences* 114.13 (2017), pp. 3521–3526. DOI: 10.1073/pnas.1611835114. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1611835114. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1611835114 (cit. on pp. 4, 12).

[68] Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. "DistiLLM-2: A contrastive approach boosts the distillation of LLMs." In: *CoRR* abs/2503.07067 (2025). URL: http://dblp.uni-trier.de/db/journals/corr/corr2503.html#abs-2503-07067 (cit. on p. 49).

[69] Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. "DistiLLM: Towards streamlined distillation for large language models." In: *ICML*. OpenReview.net, 2024. URL: http://dblp.uni-trier.de/db/conf/icml/icml2024.html#KoKCY24 (cit. on p. 48).

[70] Dawid Jan Kopiczko, Tijmen Blankevoort, and Yuki M Asano. "VeRA: Vector-based Random Matrix Adaptation". In: *The Twelfth International Conference on Learning Representations*. 2024 (cit. on pp. 12, 14, 25, 26, 42, 49, 92).

[71] Simon Kornblith, Jonathon Shlens, and Quoc V. Le. *Do Better ImageNet Models Transfer Better?* arXiv:1805.08974 [cs, stat]. 2019-06. DOI: 10.48550/arXiv.1805.08974. URL: http://arxiv.org/abs/1805.08974 (visited on 2023-11-16) (cit. on p. 12).

[72] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks". In: *Commun. ACM* 60.6 (2017-05), 84–90. ISSN: 0001-0782. DOI: 10.1145/3065386. URL: https://doi.org/10.1145/3065386 (cit. on p. 2).

[73] Ilja Kuzborskij and Yasin Abbasi Yadkori. *Low-rank bias, weight decay, and model merging in neural networks*. 2025. arXiv: 2502.17340 [cs.LG]. URL: https://arxiv.org/abs/2502.17340 (cit. on pp. 1, 5).

[74] Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. *OpenAssistant Conversations – Democratizing Large Language Model Alignment*. 2023. arXiv: 2304.07327 [cs.CL] (cit. on p. 14).

[75] Jason Lee, Kyunghyun Cho, and Douwe Kiela. "Countering Language Drift via Visual Grounding". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China: Association for Computational Linguistics, 2019-11, pp. 4385–4395. DOI: 10.18653/v1/D19-1447. URL: https://aclanthology.org/D19-1447 (cit. on pp. 4, 12).

[76] Kuang-Huei Lee, Xinyun Chen, Hiroki Furuta, John Canny, and Ian Fischer. "A human-inspired reading agent with gist memory of very long contexts". In: *Proceedings of the 41st International Conference on Machine Learning*. ICML'24. Vienna, Austria: JMLR.org, 2024 (cit. on pp. 46–48).

[77] Tony Lee, Haoqin Tu, Chi Heem Wong, Wenhao Zheng, Yiyang Zhou, Yifan Mai, Josselin Somerville Roberts, Michihiro Yasunaga, Huaxiu Yao, Cihang Xie, and Percy Liang. "VHELM: a holistic evaluation of vision language models". In: *Proceedings of the 38th International Conference on Neural Information Processing Systems*. NIPS '24. Vancouver, BC, Canada: Curran Associates Inc., 2024. ISBN: 9798331314385 (cit. on p. 53).

[78] Brian Lester, Rami Al-Rfou, and Noah Constant. "The Power of Scale for Parameter-Efficient Prompt Tuning". In: *EMNLP*. 2021 (cit. on pp. 4, 14, 30, 42).

[79] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. "Retrieval-Augmented Generation for knowledge-intensive NLP tasks". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 9459–9474. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf (cit. on pp. 46, 48).

[80] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. "BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation". In: *ICML*. 2022 (cit. on pp. 52, 56).

[81] Xiang Lisa Li and Percy Liang. "Prefix-Tuning: Optimizing Continuous Prompts for Generation". In: *ACL*. 2021 (cit. on pp. 4, 14, 42).

[82] Xuhong LI, Yves Grandvalet, and Franck Davoine. "Explicit Inductive Bias for Transfer Learning with Convolutional Networks". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 2825–2834. URL: https://proceedings.mlr.press/v80/li18a.html (cit. on p. 12).

[83] Yixing Li, Yuxian Gu, Li Dong, Dequan Wang, Yu Cheng, and Furu Wei. "Direct Preference Knowledge Distillation for Large Language Models." In: *CoRR* abs/2406.19774 (2024). URL: http://dblp.uni-trier.de/db/journals/corr/corr2406.html#abs-2406-19774 (cit. on p. 48).

[84] Vladislav Lialin, Vijeta Deshpande, and Anna Rumshisky. "Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning." In: *CoRR* abs/2303.15647 (2023). URL: http://dblp.uni-trier.de/db/journals/corr/corr2303.html#abs-2303-15647 (cit. on pp. 5, 49).

[85] Vladislav Lialin, Sherin Muckatira, Namrata Shivagunde, and Anna Rumshisky. "Relora: High-rank training through low-rank updates". In: *ICML*. 2023 (cit. on p. 42).

[86] Chin-Yew Lin. "ROUGE: A Package for Automatic Evaluation of Summaries". In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, 2004-07, pp. 74–81. URL: https://aclanthology.org/W04-1013/ (cit. on p. 55).

[87] Stephanie Lin, Jacob Hilton, and Owain Evans. "TruthfulQA: Measuring How Models Mimic Human Falsehoods". In: *ACL*. 2022 (cit. on pp. 25, 37).

[88] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. *Microsoft COCO: Common Objects in Context*. 2015. arXiv: 1405.0312 [cs.CV] (cit. on p. 90).

[89] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. *Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning*. arXiv:2205.05638 [cs]. 2022-08. DOI: 10.48550/arXiv.2205.05638. URL: http://arxiv.org/abs/2205.05638 (visited on 2023-09-12) (cit. on p. 40).

[90] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. *Visual Instruction Tuning*. 2023 (cit. on p. 3).

[91] Peiyang Liu, Xi Wang, Lin Wang, Wei Ye, Xiangyu Xi, and Shikun Zhang. "Distilling Knowledge from BERT into Simple Fully Connected Neural Networks for Efficient Vertical Retrieval". In: *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. CIKM '21. Virtual Event, Queensland, Australia: Association for Computing Machinery, 2021, 3965–3975. ISBN: 9781450384469. DOI: 10.1145/3459637.3481909. URL: https://doi.org/10.1145/3459637.3481909 (cit. on p. 6).

[92] Shih-yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. "DoRA: Weight-Decomposed Low-Rank Adaptation". In: *ICML*. 2024 (cit. on pp. 4, 5, 31, 35, 39, 41, 42, 49, 96).

[93] Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, Yandong Wen, Michael J. Black, Adrian Weller, and Bernhard Schölkopf. "Parameter-Efficient Orthogonal Finetuning via Butterfly Factorization". In: *ICLR*. 2024 (cit. on pp. 1, 5, 23, 25, 26, 33, 92).

[94] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. "P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks". In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by Smaranda Muresan, Preslav Nakov, and Aline Villavicencio. Dublin, Ireland: Association for Computational Linguistics, 2022-05, pp. 61–68. DOI: 10.18653/v1/2022.acl-short.8. URL: https://aclanthology.org/2022.acl-short.8/ (cit. on p. 4).

[95] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. "GPT understands, too". In: *AI Open*. 2023 (cit. on p. 42).

[96] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. *Ro{BERT}a: A Robustly Optimized {BERT} Pretraining Approach*. 2020. URL: https://openreview.net/forum?id=SyxS0T4tvS (cit. on pp. 25, 37).

[97] Yuang Liu, W. Zhang, and Jun Wang. "Adaptive multi-teacher multi-level knowledge distillation". In: *Neurocomputing* 415 (2020), pp. 106–113. URL: https://api.semanticscholar.org/CorpusID:224818016 (cit. on p. 6).

[98] Zequan Liu, Jiawen Lyn, Wei Zhu, and Xing Tian. "ALoRA: Allocating Low-Rank Adaptation for Fine-tuning Large Language Models". In: *NAACL*. 2024 (cit. on p. 42).

[99] Zhiheng Liu, Ruili Feng, Kai Zhu, Yifei Zhang, Kecheng Zheng, Yu Liu, Deli Zhao, Jingren Zhou, and Yang Cao. "Cones: concept neurons in diffusion models for customized generation". In: *Proceedings of the 40th International Conference on Machine Learning*. ICML'23. Honolulu, Hawaii, USA: JMLR.org, 2023 (cit. on p. 14).

[100] Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. "The flan collection: designing data and methods for effective instruction tuning". In: *Proceedings of the 40th International Conference on Machine Learning*. ICML'23. Honolulu, Hawaii, USA: JMLR.org, 2023 (cit. on p. 14).

[101] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. "ViLBERT: pretraining task-agnostic visiolinguistic representations for vision-and-language tasks". In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019 (cit. on p. 3).

[102] Yuchen Lu, Soumye Singhal, Florian Strub, Aaron Courville, and Olivier Pietquin. "Countering Language Drift with Seeded Iterated Learning". In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 6437–6447. URL: https://proceedings.mlr.press/v119/lu20c.html (cit. on p. 12).

[103] Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. "Evaluating Very Long-Term Conversational Memory of LLM Agents". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, 2024-08, pp. 13851–13870. DOI: 10.18653/v1/2024.acl-long.747. URL: https://aclanthology.org/2024.acl-long.747/ (cit. on pp. 46, 47, 55).

[104] Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. *PEFT: State-of-the-art Parameter-Efficient Fine-Tuning methods*. https://github.com/huggingface/peft. 2022 (cit. on pp. 10, 91).

[105] Sanket Vaibhav Mehta, Darshan Patil, Sarath Chandar, and Emma Strubell. *An Empirical Investigation of the Role of Pre-training in Lifelong Learning*. 2022. URL: https://openreview.net/forum?id=D9E8MKsfhw (cit. on p. 12).

[106] Fanxu Meng, Zhaohui Wang, and Muhan Zhang. "Pissa: Principal singular values and singular vectors adaptation of large language models". In: *arXiv*. 2024 (cit. on pp. 34, 42).

[107] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. "Large language models: A survey". In: *arXiv preprint arXiv:2402.06196* (2024) (cit. on p. 46).

[108] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. "T2I-Adapter: learning adapters to dig out more controllable ability for text-to-image diffusion models". In: *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*. AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN: 978-1-57735-887-9. DOI: 10.1609/aaai.v38i5.28226. URL: https://doi.org/10.1609/aaai.v38i5.28226 (cit. on p. 12).

[109] Soumik Mukhopadhyay, Matthew Gwilliam, Vatsal Agarwal, Namitha Padmanabhan, Archana Swaminathan, Srinidhi Hegde, Tianyi Zhou, and Abhinav Shrivastava. *Diffusion Models Beat GANs on Image Classification*. arXiv:2307.08702 [cs]. 2023-07. DOI: 10.48550/arXiv.2307.08702. URL: http://arxiv.org/abs/2307.08702 (visited on 2023-07-19) (cit. on p. 14).

[110] OpenAI. "GPT-4 Technical Report". In: *ArXiv* abs/2303.08774 (2023) (cit. on pp. 3, 12, 14).

[111] Charles Packer, Vivian Fang, Shishir G. Patil, Kevin Lin, Sarah Wooders, and Joseph E. Gonzalez. "MemGPT: Towards LLMs as operating systems." In: *CoRR* abs/2310.08560 (2023). URL: http://dblp.uni-trier.de/db/journals/corr/corr2310.html#abs-2310-08560 (cit. on pp. 46, 48).

[112] Sinno Jialin Pan and Qiang Yang. "A Survey on Transfer Learning". In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359. DOI: 10.1109/TKDE.2009.191 (cit. on pp. 1, 2).

[113] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. "AdapterFusion: Non-Destructive Task Composition for Transfer Learning". In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Ed. by Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty. Online: Association for Computational Linguistics, 2021-04, pp. 487–503. DOI: 10.18653/v1/2021.eacl-main.39. URL: https://aclanthology.org/2021.eacl-main.39/ (cit. on pp. 4, 12, 14, 40).

[114] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. "SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis". In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=di52zR8xgf (cit. on pp. 12, 14).

[115] Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. "Controlling text-to-image diffusion by orthogonal finetuning". In: *NeurIPS*. 2023 (cit. on pp. 1, 4–6, 12–17, 20, 22–24, 26, 30, 31, 33, 36, 39, 91).

[116] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. "Learning transferable visual models from natural language supervision". In: *ICML*. 2021 (cit. on pp. 3, 23, 30, 36).

[117] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. "Improving Language Understanding by Generative Pre-Training". In: *OpenAI* (2018). Accessed: 2024-11-15. URL: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (cit. on pp. 2, 3).

[118] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. "Language Models are Unsupervised Multitask Learners". In: *OpenAI* (2019). Accessed: 2024-11-15. URL: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (cit. on p. 3).

[119] Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. "Zep: A Temporal Knowledge Graph Architecture for Agent Memory." In: *CoRR* abs/2501.13956 (2025). URL: http://dblp.uni-trier.de/db/journals/corr/corr2501.html#abs-2501-13956 (cit. on pp. 46, 48).

[120] Nils Reimers and Iryna Gurevych. "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China: Association for Computational Linguistics, 2019-11, pp. 3982–3992. DOI: 10.18653/v1/D19-1410. URL: https://aclanthology.org/D19-1410/ (cit. on p. 55).

[121] Pengjie Ren, Chengshun Shi, Shiguang Wu, Mengqi Zhang, Zhaochun Ren, Maarten de Rijke, Zhumin Chen, and Jiahuan Pei. "MELoRA: Mini-Ensemble Low-Rank Adapters for Parameter-Efficient Fine-Tuning." In: *ACL (1)*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Association for Computational Linguistics, 2024, pp. 3052–3064. ISBN: 979-8-89176-094-3. URL: http://dblp.uni-trier.de/db/conf/acl/acl2024-1.html#RenS0ZRRCP24 (cit. on pp. 5, 42).

[122] Elad Richardson, Kfir Goldberg, Yuval Alaluf, and Daniel Cohen-Or. "ConceptLab: Creative Concept Generation using VLM-Guided Diffusion Prior Constraints". In: *ACM Trans. Graph.* 43.3 (2024-06). ISSN: 0730-0301. DOI: 10.1145/3659578. URL: https://doi.org/10.1145/3659578 (cit. on p. 14).

[123] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. "High-Resolution Image Synthesis with Latent Diffusion Models." In: *CVPR*. IEEE, 2022, pp. 10674–10685. ISBN: 978-1-6654-6946-3. URL: http://dblp.uni-trier.de/db/conf/cvpr/cvpr2022.html#RombachBLE022 (cit. on pp. 3, 12, 14, 23, 30, 36).

[124] Karsten Roth, Lukas Thede, A. Sophia Koepke, Oriol Vinyals, Olivier J Henaff, and Zeynep Akata. "Fantastic Gains and Where to Find Them: On the Existence and Prospect of General Knowledge Transfer between Any Pretrained Model". In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=m50eKHCttz (cit. on p. 12).

[125] Andreas Rücklé, Gregor Geigle, Max Glockner, Tilman Beck, Jonas Pfeiffer, Nils Reimers, and Iryna Gurevych. "AdapterDrop: On the Efficiency of Adapters in Transformers". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Ed. by Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, 2021-11, pp. 7930–7946. DOI: 10.18653/v1/2021.emnlp-main.626. URL: https://aclanthology.org/2021.emnlp-main.626/ (cit. on p. 4).

[126] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023 (cit. on pp. 12, 14, 17, 23, 36, 38, 39, 91).

[127] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Lit, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Raphael Gontijo-Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. "Photorealistic text-to-image diffusion models with deep language understanding". In: *Proceedings of the 36th International Conference on Neural Information Processing Systems*. NIPS '22. New Orleans, LA, USA: Curran Associates Inc., 2022. ISBN: 9781713871088 (cit. on p. 14).

[128] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. 2019-10. URL: https://www.semanticscholar.org/paper/a54b56af24bb4873ed0163b77df63b92bd018ddc (cit. on p. 6).

[129] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15 (2014-06), pp. 1929–1958 (cit. on p. 91).

[130] Zafir Stojanovski, Karsten Roth, and Zeynep Akata. *Momentum-based Weight Interpolation of Strong Zero-Shot Models for Continual Learning*. 2022. arXiv: 2211.03186 [cs.LG] (cit. on p. 12).

[131] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. "A survey on deep transfer learning". In: *International conference on artificial neural networks*. Springer. 2018, pp. 270–279 (cit. on p. 1).

[132] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. *Stanford Alpaca: An Instruction-following LLaMA model*. https://github.com/tatsu-lab/stanford_alpaca. 2023 (cit. on pp. 12, 14, 25, 37, 49, 92, 97).

[133] Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, et al. *Gemma 2: Improving Open Language Models at a Practical Size*. 2024. arXiv: 2408.00118 [cs.CL]. URL: https://arxiv.org/abs/2408.00118 (cit. on p. 56).

[134] Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, et al. *Qwen2.5 Technical Report*. 2025. arXiv: 2412.15115 [cs.CL]. URL: https://arxiv.org/abs/2412.15115 (cit. on p. 56).

[135] Sebastian Thrun and L. Pratt. "Learning to Learn: Introduction and Overview". In: *Learning To Learn*. Ed. by S. Thrun and L. Pratt. Kluwer Academic Publishers, 1998, pp. 3 –17 (cit. on p. 2).

[136] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. "Llama: Open and efficient foundation language models". In: *arXiv* (2023) (cit. on pp. 3, 12, 14, 30).

[137] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. "Llama 2: Open foundation and fine-tuned chat models". In: *arXiv*. 2023 (cit. on pp. 12, 25, 30, 37, 92, 97).

[138] Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. *DyLoRA: Parameter Efficient Tuning of Pre-trained Models using Dynamic Search-Free Low-Rank Adaptation*. arXiv:2210.07558 [cs]. 2023-04. DOI: 10.48550/arXiv.2210.07558. URL: http://arxiv.org/abs/2210.07558 (visited on 2023-09-12) (cit. on pp. 12, 14).

[139] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 2017, 6000–6010. ISBN: 9781510860964 (cit. on p. 3).

[140] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding". In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Ed. by Tal Linzen, Grzegorz Chrupała, and Afra Alishahi. Brussels, Belgium: Association for Computational Linguistics, 2018-11, pp. 353–355. DOI: 10.18653/v1/W18-5446. URL: https://aclanthology.org/W18-5446 (cit. on pp. 3, 25, 37).

[141] Hanqing Wang, Zeguan Xiao, Yixia Li, Shuo Wang, Guanhua Chen, and Yun Chen. "MiLoRA: Harnessing Minor Singular Components for Parameter-Efficient LLM Finetuning". In: *arXiv*. 2024 (cit. on p. 42).

[142] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. "A Comprehensive Survey of Continual Learning: Theory, Method and Application". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 46.8 (2024-08), 5362–5383. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2024.3367329. URL: https://doi.org/10.1109/TPAMI.2024.3367329 (cit. on p. 4).

[143] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. *Self-Instruct: Aligning Language Models with Self-Generated Instructions*. arXiv:2212.10560 [cs]. 2023-05. DOI: 10.48550/arXiv.2212.10560. URL: http://arxiv.org/abs/2212.10560 (visited on 2023-11-17) (cit. on p. 14).

[144] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. "Emergent Abilities of Large Language Models." In: *Trans. Mach. Learn. Res.* 2022 (2022). URL: http://dblp.uni-trier.de/db/journals/tmlr/tmlr2022.html#WeiTBRZBYBZMCHVLDF22 (cit. on p. 3).

[145] Jason Weston, Sumit Chopra, and Antoine Bordes. "Memory Networks". In: *CoRR* abs/1410.3916 (2014). URL: https://api.semanticscholar.org/CorpusID:2926851 (cit. on pp. 46, 48).

[146] Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. "Video models are zero-shot learners and reasoners." In: *CoRR* abs/2509.20328 (2025). URL: http://dblp.uni-trier.de/db/journals/corr/corr2509.html#abs-2509-20328 (cit. on p. 1).

[147] Muling Wu, Wenhao Liu, Xiaohua Wang, Tianlong Li, Changze Lv, Zixuan Ling, Zhu JianHao, Cenyuan Zhang, Xiaoqing Zheng, and Xuanjing Huang. "Advancing Parameter Efficiency in Fine-tuning via Representation Editing". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, 2024-08, pp. 13445–13464. DOI: 10.18653/v1/2024.acl-long.726. URL: https://aclanthology.org/2024.acl-long.726 (cit. on pp. 37, 39, 40, 97).

[148] Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. *ReFT: Representation Finetuning for Language Models*. 2024. arXiv: 2404.03592 [cs.CL]. URL: https://arxiv.org/abs/2404.03592 (cit. on pp. 39, 40, 96, 97).

[149] Wenhan Xia, Chengwei Qin, and Elad Hazan. "Chain of lora: Efficient fine-tuning of language models via residual learning". In: *arXiv*. 2024 (cit. on p. 42).

[150] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. "Unified Perceptual Parsing for Scene Understanding". In: *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part V*. Munich, Germany: Springer-Verlag, 2018, 432–448. ISBN: 978-3-030-01227-4. DOI: 10.1007/978-3-030-01228-1_26. URL: https://doi.org/10.1007/978-3-030-01228-1_26 (cit. on pp. 24, 37).

[151] Yi Xin, Siqi Luo, Haodi Zhou, Junlong Du, Xiaohong Liu, Yue Fan, Qing Li, and Yuntao Du. "Parameter-efficient fine-tuning for pre-trained vision models: A survey". In: *arXiv*. 2024 (cit. on p. 42).

[152] Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. "A-mem: Agentic memory for LLM agents". In: *Advances in Neural Information Processing Systems*. NeurIPS'25. 2025 (cit. on pp. 46, 48).

[153] Yuhui Xu, Lingxi Xie, Xiaotao Gu, Xin Chen, Heng Chang, Hengheng Zhang, Zhengsu Chen, XIAOPENG ZHANG, and Qi Tian. "QA-LoRA: Quantization-Aware Low-Rank Adaptation of Large Language Models". In: *International Conference on Representation Learning*. Ed. by B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun. Vol. 2024. 2024, pp. 52401–52418. URL: https://proceedings.iclr.cc/paper_files/paper/2024/file/e6c2e85db1f1039177c4495ccd399ac4-Paper-Conference.pdf (cit. on p. 14).

[154] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. "How transferable are features in deep neural networks?" In: *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'14. Montreal, Canada: MIT Press, 2014, 3320–3328 (cit. on pp. 1, 2).

[155] Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. "BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Language-models". In: *ACL*. 2022 (cit. on pp. 40, 42).

[156] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. "SLCA: Slow Learner with Classifier Alignment for Continual Learning on a Pre-trained Model". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 19148–19158 (cit. on p. 12).

[157] Jingfan Zhang, Yi Zhao, Dan Chen, Xing Tian, Huanran Zheng, and Wei Zhu. "MiLoRA: Efficient Mixture of Low-Rank Adaptation for Large Language Models Fine-tuning". In: *Findings of the Association for Computational Linguistics: EMNLP 2024*. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association for Computational Linguistics, 2024-11, pp. 17071–17084. DOI: 10.18653/v1/2024.findings-emnlp.994. URL: https://aclanthology.org/2024.findings-emnlp.994/ (cit. on p. 5).

[158] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. "Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation." In: *ICCV*. IEEE, 2019, pp. 3712–3721. ISBN: 978-1-7281-4803-8. URL: http://dblp.uni-trier.de/db/conf/iccv/iccv2019.html#ZhangSGCBM19 (cit. on p. 6).

[159] Lvmin Zhang and Maneesh Agrawala. "Adding conditional control to text-to-image diffusion models". In: *ICCV*. 2023 (cit. on pp. 12, 14, 24, 36, 91).

[160] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. "Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning". In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=lq62uWRJjiY (cit. on pp. 5, 14, 42).

[161] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 586–595. DOI: 10.1109/CVPR.2018.00068 (cit. on p. 23).

[162] Rongzhi Zhang, Jiaming Shen, Tianqi Liu, Haorui Wang, Zhen Qin, Feng Han, Jialu Liu, Simon Baumgartner, Michael Bendersky, and Chao Zhang. "PLaD: Preference-based large language model distillation with pseudo-preference pairs." In: *ACL (Findings)*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Association for Computational Linguistics, 2024, pp. 15623–15636. ISBN: 979-8-89176-099-8. URL: http://dblp.uni-trier.de/db/conf/acl/acl2024f.html#ZhangS0W0HLBBZ24 (cit. on p. 49).

[163] Ruiyi Zhang, Rushi Qiang, Sai Ashish Somayajula, and Pengtao Xie. "AutoLoRA: Automatically Tuning Matrix Ranks in Low-Rank Adaptation Based on Meta Learning". In: *NAACL*. 2024 (cit. on p. 42).

[164] Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. *Instruction Tuning for Large Language Models: A Survey*. arXiv:2308.10792 [cs]. 2023-10. DOI: 10.48550/arXiv.2308.10792. URL: http://arxiv.org/abs/2308.10792 (visited on 2023-11-17) (cit. on p. 14).

[165] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. "BERTScore: Evaluating text generation with BERT." In: *ICLR*. OpenReview.net, 2020. URL: http://dblp.uni-trier.de/db/conf/iclr/iclr2020.html#ZhangKWWA20 (cit. on p. 55).

[166] Yuhui Zhang, Yuchang Su, Yiming Liu, Xiaohan Wang, James Burgess, Elaine Sui, Chenyu Wang, Josiah Aklilu, Alejandro Lozano, Anjiang Wei, Ludwig Schmidt, and Serena Yeung-Levy. "Automated Generation of Challenging Multiple-Choice Questions for Vision Language Model Evaluation." In: *CVPR*. Computer Vision Foundation / IEEE, 2025, pp. 29580–29590. ISBN: 979-8-3315-4364-8. URL: http://dblp.uni-trier.de/db/conf/cvpr/cvpr2025.html#ZhangSLWBS0ALWS25 (cit. on p. 53).

[167] Yuxin Zhang, Nisha Huang, Fan Tang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. "Inversion-based Style Transfer with Diffusion Models". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*

(CVPR). 2023, pp. 10146–10156. DOI: 10.1109/CVPR52729.2023.00978 (cit. on p. 14).

[168] Bingchen Zhao, Haoqin Tu, Chen Wei, Jieru Mei, and Cihang Xie. "Tuning LayerNorm in Attention: Towards Efficient Multi-Modal LLM Finetuning". In: *ICLR*. 2024 (cit. on p. 42).

[169] Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. "GaLore: memory-efficient LLM training by gradient low-rank projection". In: *Proceedings of the 41st International Conference on Machine Learning*. ICML'24. Vienna, Austria: JMLR.org, 2024 (cit. on p. 12).

[170] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. *A Survey of Large Language Models*. arXiv:2303.18223 [cs]. 2023-09. DOI: 10.48550/arXiv.2303.18223. URL: http://arxiv.org/abs/2303.18223 (visited on 2023-11-17) (cit. on pp. 14, 46).

[171] Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. "MemoryBank: enhancing large language models with long-term memory". In: *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*. AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN: 978-1-57735-887-9. DOI: 10.1609/aaai.v38i17.29946. URL: https://doi.org/10.1609/aaai.v38i17.29946 (cit. on pp. 46, 48).

[172] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. "Semantic Understanding of Scenes Through the ADE20K Dataset". In: *Int. J. Comput. Vision* 127.3 (2019-03), 302–321. ISSN: 0920-5691. DOI: 10.1007/s11263-018-1140-0. URL: https://doi.org/10.1007/s11263-018-1140-0 (cit. on pp. 24, 36).

[173] Jiacheng Zhu, Kristjan Greenewald, Kimia Nadjahi, Haitz Sáez de Ocáriz Borde, Rickard Brüel Gabrielsson, Leshem Choshen, Marzyeh Ghassemi, Mikhail Yurochkin, and Justin Solomon. "Asymmetry in Low-Rank Adapters of Foundation Models". In: *ICML*. 2024 (cit. on p. 42).

[174] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. "Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models". In: *arXiv preprint arXiv:2504.10479* (2025) (cit. on pp. 53, 55, 56).

[175] Bojia Zi, Xianbiao Qi, Lingzhi Wang, Jianan Wang, Kam-Fai Wong, and Lei Zhang. "Delta-lora: Fine-tuning high-rank parameters with the delta of low-rank matrices". In: *arXiv*. 2023 (cit. on p. 42).

A

ETHER: Efficient Finetuning of Large-Scale Models with Hyperplane Reflections

In this appendix, we augment the main paper with additional, qualitative evidence for the learning rate robustness of *ETHER* transformations in section A.1. In addition, we also provide benchmark-specific qualitative examples for subject-driven and controllable image generation in section A.2. For all experiments - both those in the main paper and supplementary results, we then list all relevant details in section A.3 for our studies on finetuning in subject-driven image generation (§A.3.1), controllable image synthesis (§A.3.2), natural language understanding tasks (§A.3.3) and instruction tuning (§A.3.4). We then provide two additional *ETHER* ablations in section A.4 - for the number of block-diagonals and the specific double-sided application in *ETHER+*. Finally, we present preliminary results on the Visual Task Adaptation Benchmark (§A.5).

A.1 Qualitative Evidence of Learning Rate Robustness

As introduced in Sec. 2.3, when finetuning with *ETHER* transformation, by construction, the learning rate only controls the speed with which reflection angels change. As a consequence, *ETHER* methods are much more robust to learning rate choices, and less likely to diverge and cause model deterioration. This allows for user control over the convergence speed while minimizing the risk of model collapse during training. To demonstrate this, Sec. 2.4 introduced both a qualitative example comparing the impact of minimal and maximal perturbation strength on the model output in Fig. 2.3, and quantitative evaluations on the Semantic Map to Image task against learning rate choices in Figs. 2.5 and 2.6.

In this section, we augment Sec. 2.4 and provide additional qualitative results and impressions to highlight the non-deteriorating nature of *ETHER* transformation. For this, we showcase subject-driven generation results using different finetuning methods in Fig. A.1, with default generations using the best learning rate. We then systematically increase the finetuning learning rate by 10 and by 100 times, and visualize the correspondingly generated output. As can be seen, for 10× higher learning rates OFT and Naive fail to follow the text prompt, while LoRA finetuning quickly collapses. With 10× lower learning rates instead, OFT, Naive and *ETHER* are not able to generate the subject correctly in the predefined number of iterations.

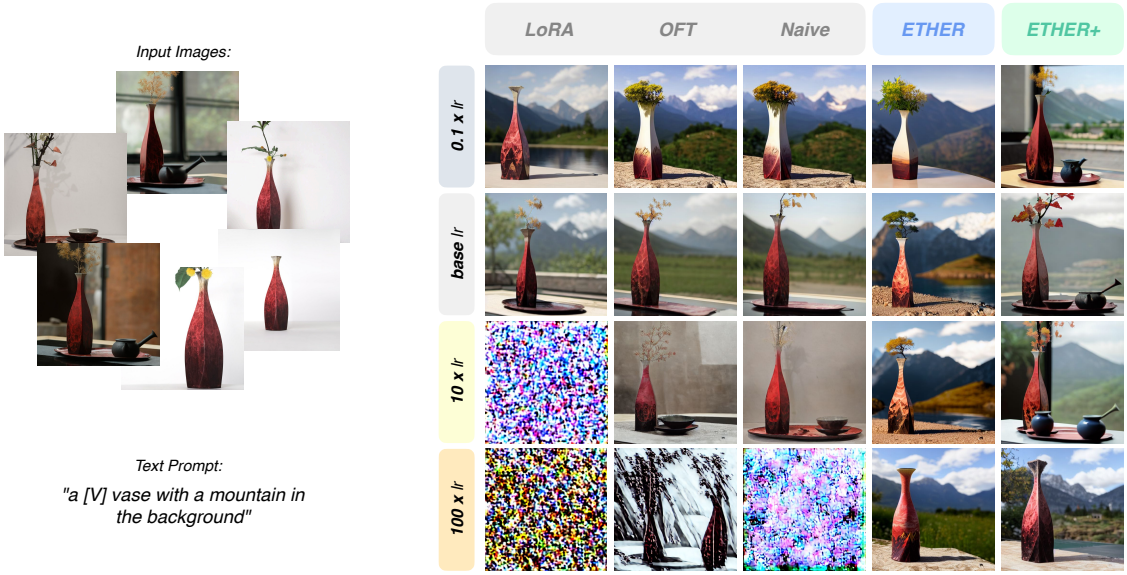

Figure A.1: Qualitative visualization of learning rate robustness of *ETHER* and *ETHER*+ in subject-driven generation finetuning. We see how *ETHER* methods are able to consistently produce good results avoiding model deterioration. Specifically, *ETHER*+ shows impressive capabilities, being able to follow the subject-prompt instructions in the widest learning rate range.

A.2 Qualitative Examples for ETHER Finetuning

We show some qualitative results by using the finetuning methods proposed in this paper.

A.2.1 Subject-driven Generation.

In Figure A.2 we report subject-driven generation examples. In particular, for a fair comparison, we report images which come from the same noise vector in the Stable Diffusion latent space. For the *sunglasses* images, we see how non-*ETHER* methods manage to reproduce the subject, but fail to follow the text prompt in most cases. Interestingly in the first row, we notice how *ETHER*+ is able to properly control the generation, by transforming the yellow area (associated to a beer in other models) in an enlightened Eiffel Tower. For the *teapot* images instead, we see how *ETHER*+ is able to better keep the appearances of the subject.

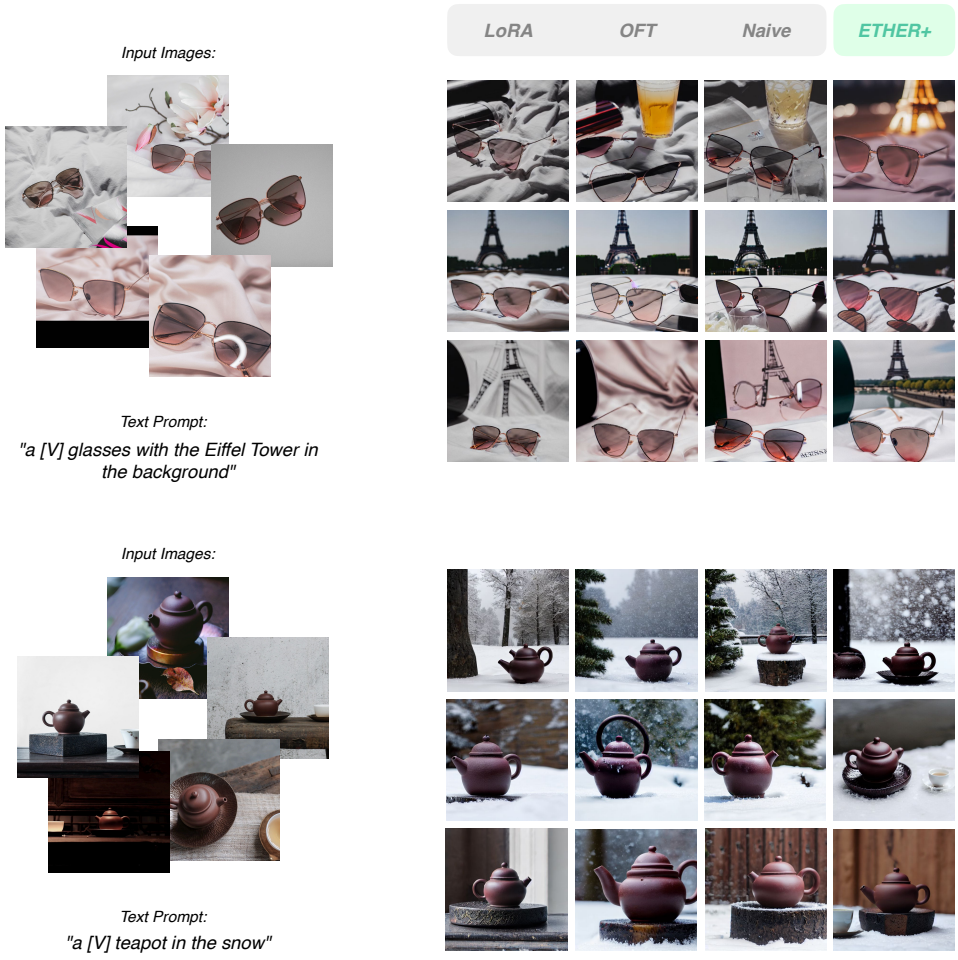

Figure A.2: Subject-driven Generation results. Each row shares initial latent noise (notice row-wise similarities). We can see that *ETHER*+ method is better at adapting the model to the subjects. Notice how for the pink sunglasses, OFT and Naive fail in following the prompt.

A.2.2 Controllable Generation.

In Figure A.3 we show some examples from the Semantic Map to Image task. In particular, we notice how in the first row all models but *ETHER*+ fail to control the image correctly, not being able to separate the land from the water. Additionally, in the second row OFT fails to generate the sky, while Naive presents a halo effect. These examples showcase the abilities of *ETHER*+ finetuning over the other methods.

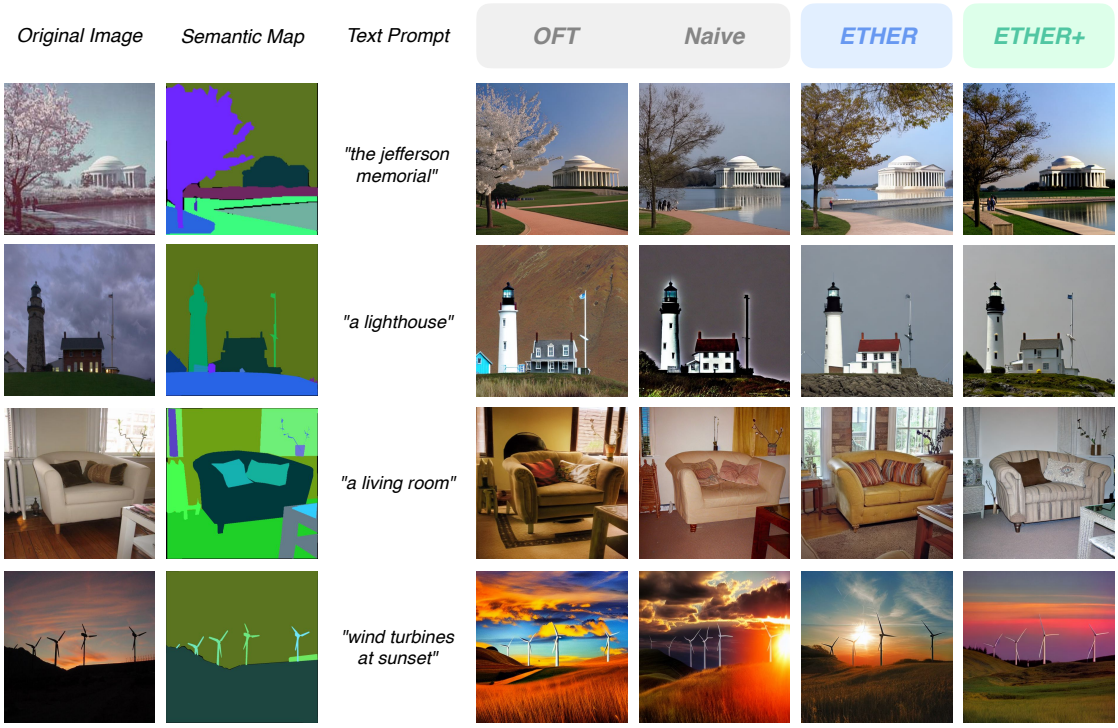

Figure A.3: Semantic Map to Image Qualitative Results. We notice how in the first row all models but *ETHER*+ fail to control the image correctly. Overall *ETHER*+ controlled images show better control.

To show broader controllable capabilities, we also report few qualitative examples with *ETHER* methods trained with Landmarks and Canny Edge Maps control signals on CelebA-HQ [62] and COCO 2017 [88] datasets respectively.

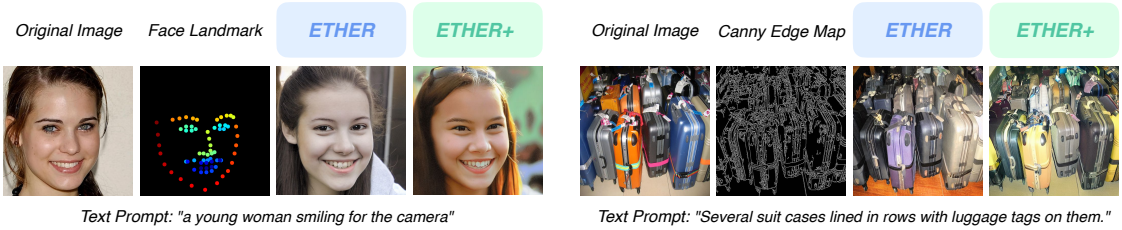

Figure A.4: Examples of Landmark to Face (left) and Canny Edge Map to Image (right) controlled generation with *ETHER* methods.

A.3 Experimental Details

This section provides additional experimental details for replication not listed in the main benchmark experimental section 2.5. It is worth noting that while in most of our experiments we do not employ regular dropout [129], **[liu2023parameterefficient]** proposes a multiplicative dropout form specifically designed for multiplicative finetuning methods, which we did not test in this study. We hypothesize that this specialized dropout technique could potentially work better than regular dropout for *ETHER* and *ETHER+* as well. We also note that [115] report OFT's number of parameters as half of the actual trainable parameters due to the redundancy in the skew symmetric matrices S^B in the Cayley parametrization of Q^B. Basically, we they report the storage parameters for Q^B rather than the training parameters. For consistency and fair comparisons, we follow the same convention for OFT throughout our paper.

A.3.1 Subject-driven Generation

For subject-driven generation, we follow the same setting listed in DreamBooth [126], using DreamBooth and OFT [115] baselines as implemented in official OFT GitHub repository. The additional trainable layers follow [115] and are added to the Q,K,V layers and the projection layer inside every attention module. The training is performed over 1400 iterations for each method, evaluating the generation results every 200 iterations at selecting the best one (typically around 1200 iterations). For DreamBooth and OFT, we follow the original implementations and use a learning rate of 5×10^{-6} and 6×10^{-5} respectively, with a batch size of 1. For Naive - the non-orthogonal OFT variant - we use the same setting of OFT for a fair comparison. For *ETHER* and *ETHER+*, we use a learning rate of 6×10^{-3}. We perform the training on a Tesla V100-32GB GPU.

A.3.2 Controllable Generation

For our experiments on controllable image generation we follow the setting of [115], using the signal encoder from ControlNet [159] (comprising 8 trainable convolutional layers, accounting for 3.1M additional learnable parameters). Finetuning parameters are added to the Q,K,V layers as well as the projection layer of the attention modules and the subsequent feedforward layers. As baselines, we use the official implementation of OFT. Similarly to [115], for OFT and Naive we use a learning rate of 1×10^{-5}. For ETHER and ETHER+ we use a larger learning rate of 1×10^{-3}. For all experiments, we upper bound the learning rate of the signal encoder to 1×10^{-4}. We perform all the training runs on a single Nvidia-A100-40GB with a batch size of 10.

A.3.3 Natural Language Understanding

For our GLUE benchmark experiments finetuning DeBERTaV3-base [46], we make use of the peft repository [104] as the basis for our implementations. To compare our results

with those of [93], we follow their implementation and apply *ETHER* and *ETHER+* to all the linear layers in every transformer block. The relevant hyperparameters for each task are reported in Tab. A.2. All training runs are conducted on a single Nvidia-A100-40GB GPU.

Table A.1: GLUE benchmark hyperparameters.

Method	Hyperparameters	MNLI	SST-2	CoLA	QQP	QNLI	RTE	MRPC	STS-B
ETHER	Learning Rate	8e-4	1e-3	1e-3	3e-4	1e-3	1e-3	3e-4	2e-3
	Batch Size	32	32	32	8	8	32	32	8
	Num. Epochs	9	14	10	20	7	13	14	8
	Dropout	1e-3	1e-3	1e-1	1e-1	1e-3	1e-2	1e-1	1e-1
	Max Seq. Len.	256	128	64	320	512	320	320	128
ETHER+	Learning Rate	8e-4	1e-4	1e-3	3e-3	3e-3	3e-4	8e-4	8e-4
	Batch Size	8	8	8	32	32	8	32	8
	Num. Epochs	8	10	6	16	5	35	17	11
	Dropout	1e-3	1e-3	1e-1	1e-3	1e-3	1e-3	1e-2	1e-3
	Max Seq. Len.	256	128	64	320	512	320	320	128

A.3.4 Instruction Tuning

For our Instruction Tuning experiments, we use the LoRA [54] finetuning implementation in the lit-gpt repository [3] as baseline. For evaluations, we make use of [35]'s benchmark implementations. For the recently proposed VeRA [70] baseline, we reproduce the model implementation following their best performing method as described in the paper: sampling random A and B matrices with uniform kaiming initialization scaled by the matrix dimension, and a learnable, non-zero diagonalized vector initialized as a vector of all zeros apart for one element equal to 0.1. Same for OFT, for which we follow the implementation in the official repository oft, selecting the number of block-diagonal matrices such that the overall number of parameters becomes comparable with *ETHER+* and LoRA rank 8. For all experiments, we use a cosine annealing learning rate scheduler, no dropout, and 1000 warmup steps. For LoRA, VeRA, and OFT we use AdamW optimizer with a weight decay of 0.01, while for *ETHER* methods, given the normalization happening on the parameters, weight decay would have limited impact and thus we set it to 0. For LoRA and VeRA, we keep α fixed with respect to the learning rate by setting it equal to the rank. For all experiments, we conduct an extensive grid search over learning rates and batch sizes. For each combination, we perform the LLama-2-7B [137] finetuning over Alpaca [132] for one epoch. All training runs are conducted on a single Nvidia-A100-40GB GPU, but could also be run on a consumer NVIDIA GeForce-RTX-3090-24G GPU.

Table A.2: Instruction Tuning hyperparameters.

	VeRA$_{r=64}$	VeRA$_{r=256}$	LoRA$_{r=1}$	LoRA$_{r=8}$	OFT$_{n=256}$	*ETHER*$_{n=32}$	*ETHER*+$_{n=32}$
Learning Rate	5e-3	1e-3	3e-3	5e-4	5e-4	2e-3	5e-3
Batch Size	32	32	8	8	16	8	16

A.4 *ETHER* Ablations

This section details additional ablation experiments on the impact of the block-diagonality degree on the final performance, as well as experimental support to the theoretical motivation in Sec. 2.3.3 to apply the relaxed Householder transformation on both the left and right side of the weight matrix.

A.4.1 Block-diagonal *ETHER* Performances

In table A.3 and table A.4, we compare the usage of multiple diagonal blocks for *ETHER* finetuning to allow for fast performance, especially in large models domain. Both tables augment our method description in Sec. 2.3.4 and the shortened results in Tab. 2.1. In all cases, we notice that performance remains almost unaffected by the choice of block number, while on the other hand, the computational efficiency consistently increases (8.22 TFLOPs for $n = 32$ versus 25.26 TFLOPs for $n = 1$ for Llama-2-7B). It is worth noting that results for *ETHER*+ with $n = 32$ show better performance with respect to less diagonalized counterparts. This could be due to the different strength of using more/less diagonalized matrices.

Table A.3: Semantic Map to Image (S2I) results for different number of diagonal blocks n on *ETHER* finetuning at epoch 10

ETHER	#params	mIoU ↑	Acc ↑	FID↓
$n = 1$	0.1M	23.1	61.23	31.7
$n = 4$	0.1M	22.9	60.92	30.5
$n = 16$	0.1M	22.3	60.35	30.7

Table A.4: Instruction Tuning results for different number of diagonal blocks n on ETHER finetuning

ETHER+	#params	TFLOPs	MMLU ↑	ARC ↑	Tru-1↑	Tru-2 ↑
$n = 1$	1.04M	51.65	43.75	46.76	28.03	41.06
$n = 4$	1.04M	18.66	43.91	45.73	27.54	40.46
$n = 32$	1.04M	9.04	44.87	46.50	29.38	43.51

A.4.2 Double-sided Application of ETHER+

Finally, we provide a brief ablation study in Tab. A.5, comparing the *ETHER*+ performance when applying the relaxed Householder transformations H^+ on only one side versus

both sides. Although the parameter count doubles, we observe a significant increase in performance (e.g. 0.666 vs 0.618 in DINO score) as higher transformation distances can be achieved.

Table A.5: Subject-driven Generation image quality results comparison (at iteration 1200) among standard ETHER+ and its version only applied on one side of the weight matrix.

	#params	DINO ↑	CLIP-I ↑
ETHER+ (one-sided)	0.2M	0.618	0.777
ETHER+	0.4M	**0.666**	**0.800**

A.5 VTAB results

We also perform a small evaluation over a subset of the popular Visual Task Adaptation Benchmark (VTAB), using an ImageNet-21k pretrained ViT-B. As can be seen, *ETHER* and *ETHER+* perform comparably to OFT with $n = 256$ and LoRA rank 8, while using a fraction of the trainable parameters.

Table A.6: VTAB results

	#params	**Natural**				**Specialized**	**Structured**
		Caltech101	DTD	Flowers102	SVHN	EuroSAT	sNORB-Elev
Full FT	85.8M	96.26	73.03	98.71	73.71	96.16	63.36
Linear Prob.	0	95.96	72.34	**99.12**	52.55	95.03	34.09
LoRA$_{r=8}$	1.33M	97.69	**77.50**	99.10	**97.40**	98.92	74.89
OFT$_{n=256}$	0.29M	96.95	75.80	98.60	96.58	98.83	74.37
ETHER	0.08M	97.64	75.85	98.83	95.81	98.80	74.17
ETHER+	0.33M	**98.27**	76.92	98.88	96.84	**99.15**	**78.41**

B

Decoupling Angles and Strength in Low-rank Adaptation

B.1 *ETHER* and *ETHER+* low-rank limitation

In *ETHER* and *ETHER+*, even if the applied transformation matrices $I - uu^\top$ are full-rank, the resulting weight updates to the pretrained layers are limited to be low-rank. We can show this by rewriting the transformation result in a residual form.

For *ETHER* the matrix multiplication can be written as:

$$\begin{aligned} HW &= (I - 2uu^\top)W \\ &= W - 2uu^\top W \end{aligned}$$

where the second term on the right-hand side, by multiplying the pretrained matrix with a rank-1 transformation, restricts the learnable weight updates, which are driven by u, to be rank-1.

Similarly, for *ETHER+*:

$$\begin{aligned} &H^+W\tilde{H}^+ \\ &= (W - uu^\top W + vv^\top W)\tilde{H}^+ \\ &= W - uu^\top W + vv^\top W - (W - uu^\top W + vv^\top W)\tilde{u}\tilde{u}^\top + (W - uu^\top W + vv^\top W)\tilde{v}\tilde{v}^\top \end{aligned}$$

where the rank-1 residual matrices on the right-hand side will lead to rank-4 overall weight updates.

This simple mathematical derivation, demonstrates that *ETHER* and *ETHER+* methods are limited to be low-rank, arguably limiting the expressivity and the learning capacity of the two methods.

B.2 Experimental Details

In this section we report further details about experiments in Section 4.4, along with hyperparameter choices, and standard deviation results.

Subject-Driven Generation. To find the best hyperparameters, we trained and evaluated on the first 3 subjects (10% of the data) for each method among LoRA, DoRA and DeLoRA, all with rank 16. Then, we used best hyperparameters to evaluate each method on all 30 subjects, for 3 different seeds. For LoRA and DoRA we followed best practices and fixed lambda to twice the rank during hyperparemeter search. Optimal learning rate for both methods is 6e-4. For DeLoRA we fixed the λ scaling parameter to 1e-3, and found an optimal learning rate of 2e-2 for the BA matrices. Results with standard deviations are reported in table B.1.

Method		DINO	CLIP-I
LoRA$_{r=16}$	[54]	$\underline{0.686}_{\pm.0012}$	$0.818_{\pm.0017}$
DoRA$_{r=16}$	[92]	$\mathbf{0.687}_{\pm.0015}$	$\underline{0.819}_{\pm.0015}$
DeLoRA$_{r=16}$	(ours)	$\underline{0.686}_{\pm.0056}$	$\mathbf{0.820}_{\pm.0027}$

Table B.1: Results with standard deviation for subject-driven image generation trained methods. Best scores are highlighted in bold, and second-best scores are underlined.

GLUE. Following [148], for each benchmark task, we split the publicly available validation set in two subsets as reported in Table B.2. When validation sets are larger than 2K, a 1K subset is used as new validation set, and the remaining as test set, otherwise the validation is split in two equally sized subsets. We use the new validation set to tune the hyperparameters on seed 42. Then, best hyperparameters are used to evaluate test performance for seeds 42, 43, 44, 45, 46. For each training run, we use checkpointing to save the best training run, and evaluate with that. For all experiments we use a max sequence length of 512. For larger datasets (MNLI, SST-2, QNLI, QQP) we fix the λ scaling learning rate to 3e-3, while for smaller datasets we fix it to 1e-2. For other hyperparameters we run a small grid search. Best values are reported in Table B.4. We highlight that with respect to [148], we don't discard any underperforming seed. Experiments with standard deviation details are reported in Table B.3.

Splits Sizes	MNLI	SST-2	MRPC	CoLA	QNLI	QQP	RTE	STS-B
Training Set	393K	67K	3.7K	8.5K	105K	364K	2.5K	5.7K
New Validation Set	1K	436	204	522	1K	1K	139	750
New Test Set	8K	436	204	521	4.5K	39K	138	750

Table B.2: GLUE dataset sizes, with new validation and test splits following [148] setup.

	#param	MNLI	SST-2	MRPC	CoLA	QNLI	QQP	RTE	STS-B	Avg
Full Finet.	125M	$87.3_{\pm.34}$	$94.4_{\pm.96}$	$87.9_{\pm.91}$	$62.4_{\pm3.29}$	$92.5_{\pm.22}$	$91.7_{\pm.19}$	$78.3_{\pm3.20}$	$90.6_{\pm.59}$	85.6
BitFit	0.1M	$84.7_{\pm.08}$	$\mathbf{94.0}_{\pm.87}$	$88.1_{\pm1.57}$	$54.0_{\pm3.07}$	$91.0_{\pm.05}$	$87.3_{\pm.02}$	$69.8_{\pm1.51}$	$89.5_{\pm.35}$	82.3
IA3	0.06M	$85.4_{\pm-}$	$93.4_{\pm-}$	$86.4_{\pm-}$	$57.8_{\pm-}$	$91.1_{\pm-}$	$88.5_{\pm-}$	$73.5_{\pm-}$	$88.5_{\pm-}$	83.1
LoReFT	0.02M	$83.1_{\pm.26}$	$93.4_{\pm.64}$	$\mathbf{89.2}_{\pm2.62}$	$60.4_{\pm2.60}$	$91.2_{\pm.25}$	$87.4_{\pm.23}$	$\mathbf{79.0}_{\pm2.76}$	$90.0_{\pm.29}$	84.2
RED	0.02M	$83.9_{\pm.14}$	$93.9_{\pm.31}$	$\mathbf{89.2}_{\pm.98}$	$61.0_{\pm2.96}$	$90.7_{\pm.35}$	$87.2_{\pm.17}$	$78.0_{\pm2.06}$	$90.4_{\pm.32}$	84.3
LoRA	0.3M	$86.6_{\pm.23}$	$93.9_{\pm.49}$	$88.7_{\pm.76}$	$59.7_{\pm4.36}$	$\mathbf{92.6}_{\pm.10}$	$90.4_{\pm.08}$	$75.3_{\pm2.79}$	$90.3_{\pm.54}$	84.7
$\text{Adapter}^{\text{FFN}}$	0.3M	$\mathbf{87.1}_{\pm.10}$	$93.0_{\pm.05}$	$88.8_{\pm1.38}$	$58.5_{\pm1.69}$	$92.0_{\pm.28}$	$90.2_{\pm.07}$	$77.7_{\pm1.93}$	$90.4_{\pm.31}$	84.7
Adapter	0.4M	$87.0_{\pm.28}$	$93.3_{\pm.40}$	$88.4_{\pm1.54}$	$60.9_{\pm3.09}$	$92.5_{\pm.02}$	$\mathbf{90.5}_{\pm.08}$	$76.5_{\pm2.26}$	$90.5_{\pm.35}$	85.0
DeLoRA(ours)	0.3M	$86.9_{\pm.21}$	$93.7_{\pm.79}$	$88.6_{\pm1.49}$	$\mathbf{64.7}_{\pm2.33}$	$\mathbf{92.6}_{\pm.53}$	$90.2_{\pm.17}$	$77.3_{\pm1.96}$	$\mathbf{90.6}_{\pm.38}$	**85.6**

Table B.3: **GLUE benchmark.** Comparisons of different methods finetuning RoBERTa-base, with standard deviations. Results of all baselines are taken from [147] and [148].

Hyperparameters	MNLI	SST-2	MRPC	CoLA	QNLI	QQP	RTE	STS-B
λ	12	12	4	4	12	4	12	12
Learning Rate	1e-3	1e-3	3e-2	1e-2	3e-3	1e-3	1e-2	1e-2
Batch Size	32	32	32	8	32	256	8	8
Num. Epochs	30	30	40	80	25	25	80	40
Dropout	0	0.1	0.2	0.2	0.25	0.25	0	0.2

Table B.4: GLUE benchmark hyperparameters.

Instruction Tuning. To assess the performance of DeLoRA in finetuning LLMs for Instruction Tuning, we adopted the experimental setup from [11], finetuning Llama-2-7B [137] on the Alpaca dataset [132] for one epoch, and searching for hyperparameters that deliver the best average performance across MMLU, ARC, and TruthfulQA. For DoRA we used a learning rate of 3e-4, a batch size of 8, and 100 warmup steps. For DeLoRA we used an initial scaling λ of 8, learning rates of 1e-2 for BA and 5e-3 for λ, and other hyperparameters as DoRA. All additional reported results are sourced from [11].

B.3 Fixing the magnitude term in DoRA

In the following section we provide preliminary experiments testing if fixing the magnitude in DoRA could lead to similar robustness properties as DeLoRA.

Performance. We first evaluate if fixing the magnitude term could be detrimental in terms of performance. Following the setting of our small-scale ablation in section 3.3.2, we run a small scale experiment comparing DoRA with its variation.

Method	DINO	CLIP-I
$\text{DoRA}_{r=16}$(fixed-magnitude)	0.681	0.822
$\text{DoRA}_{r=16}$	0.683	0.820

Table B.5: Subject-driven Image Generation small-scale ablation

We notice how DoRA results without updating the magnitude term seem to lead to only slightly underperforming results with respect to standard DoRA.

Robustness. We then run the same robustness analysis as reported in fig. 3.2. We see how fixing the magnitude term does not lead to a behavior similar to DeLoRA, but rather still follows DoRA behavior.

Plots in fig. B.1 show that simply fixing the magnitude term does not alter DoRA robustness properties (fig. B.1, Left), while actually in higher learning rate regimes seems to lead to further divergence (fig. B.1 Right), not allowing the magnitude to counterbalance the divergent trend. This behavior suggests that keeping column norms constant might not be restrictive enough. In this regard, DeLoRA inner normalization in terms of Frobenius distance seems to be a more promising strategy to avoid model divergence.

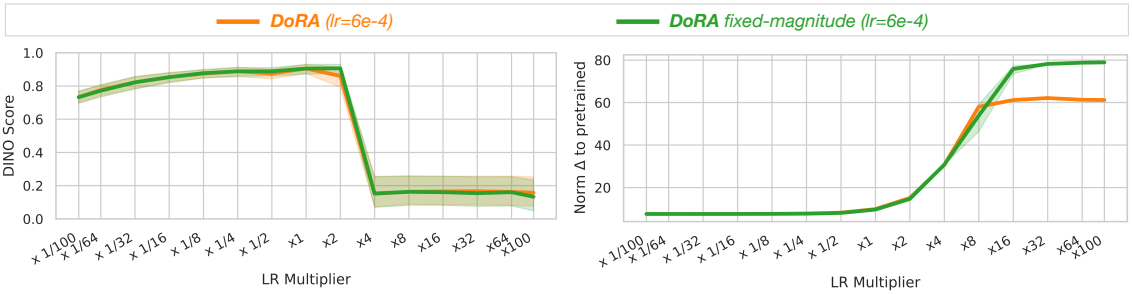

Figure B.1: Robustness analysis between DoRA with and without magnitude updates, with respect to learning rate changes from the optimal learning rate.

B.4 Robustness Ablation on DeLoRA's boundary and angles

We additionally conducted an ablation on DeLoRA's setting, where we run the same robustness analysis of section 3.3.4 by varying the learning rate of the scaling term λ (affecting the boundary), and the weights BA (angular component). We notice how all methods lead to convergence, additionally demonstrating DeLoRA's robustness properties.

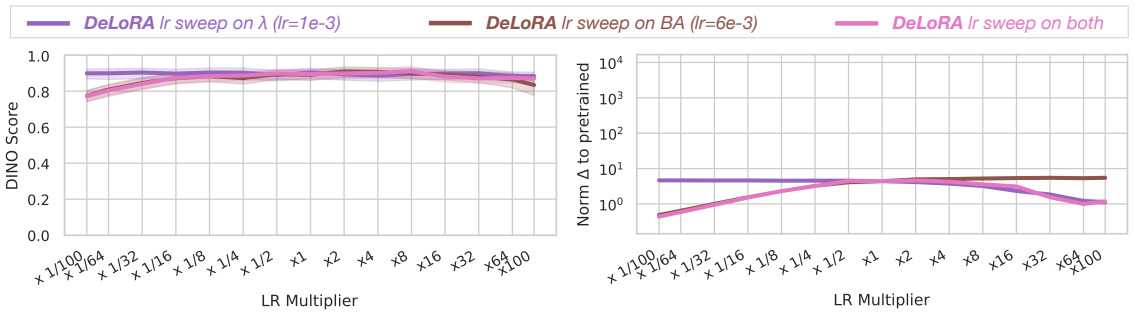

Figure B.2: Learning rate robustness plots for DeLoRA in Subject-driven generation task in terms of DINO scores (Left) and Euclidean distance finetuned vs pretrained weights of a projection layer (Right). Ablation testing impact of increasing learning rate for boundary (λ) or angular weights (BA).

B.5 Qualitative Examples

We report in fig. B.3 qualitative examples generated by our propopsed DeLORA finetuning Stable Diffusion for the tasks of Subject-driven Generation and Semantic Map to Image. While in Figure 8 we report qualitative examples of prolonged genearation with DeLoRA, LoRA and DoRA methods.

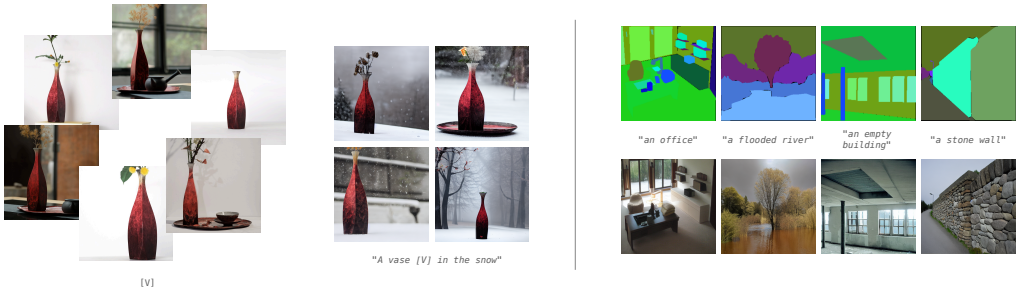

Figure B.3: Examples generated by DeLoRA-finetuned Stable Diffusion for personalized generation on a small set of subject-specific images (left), and for semantic map to image on ADE20K (right).

Figure B.4: Prolonged finetuning generated examples generated by DeLoRA, LoRA, and DoRA methods, up to time step 2600.

C

Publications and contributions

C.1 Publications

This thesis is based on the following publications. An overview of the contributions can be found in Sec. 1.4. Bold names correspond to the name of the author of this thesis.

1. [11] **M. Bini**, K. Roth, Z. Akata, A. Khoreva "ETHER: Efficient Finetuning of Large-Scale Models via Hyperplane Reflections". In: *The International Conference on Machine Learning (ICML)*. 2024.

2. [10] **M. Bini**, L. Girrbach, Z. Akata, "Decoupling Angles and Strength in Low-rank Adaptation". In: *The International Conference on Learning Representations (ICLR)*. 2025.

3. [under review] **M. Bini**, O. Bohdal, U. Michieli, Z. Akata, M. Ozay, T. Ceritli, "MemLoRA: Distilling Expert Adapters for On-Device Memory Systems". Under review.

C.2 Contributions

This section presents the contributions of the authors for the publications included in this thesis, as mentioned in Sec. C.1

Chapter 2: *ETHER* - Efficient Finetuning of Large-Scale Models via Hyperplane Reflections. This work was done in collaboration with Karsten Roth, Zeynep Akata, and Anna Khoreva. Massimo Bini was the first author and contributed to the ideation, implementation, and the majority of the experiments. Karsten Roth was the second author and participated in running the experiments for the GLUE benchmark, along with providing his domain expertise in transfer and representation learning. Zeynep Akata and Anna Khoreva had a supervisory role, guiding the project progress and establishing key milestones. All authors contributed to the writing of the paper.

Chapter 3: Decoupling Angles and Strength in Low-rank Adaptation. This work was done in collaboration with Leander Girrbach and Zeynep Akata. Massimo Bini was the first author and contributed to the ideation, and the majority of implementations and

experiments. Leander Girrbach was the second author and implemented the initial version of the code for the GLUE benchmark, along with scripts for analysis and visualizations. Zeynep Akata had a supervisory role, guiding the project progress and establishing key milestones. All authors contributed to the writing of the paper.

Chapter 4: MemLoRA - Distilling Expert Adapters for On-Device Memory Systems. Massimo Bini was the first author and contributed to the enhancement of the original idea, proposing the separation into specialized adapter-experts, in addition to all the vision augmentation ideas. He also implemented the code and run all the experiments. Umberto Michieli, Ondrej Bohdal and Taha Ceritli, contributed in proposing the idea of using Knowledge Distillation in Memory Systems. Ondrej Bohdal, Umberto Michieli, Zeynep Akata, Mete Ozay, and Taha Ceritli all had a supervisory role, guiding the project progress and establishing key milestones. All authors contributed to the writing of the paper.