Navigating Parameter Space with Geodesic Interpolation: A New Approach to Efficient Fine-Tuning

Sophia J. Abraham[1], **Jonathan D. Hauenstein**[2], **Walter J. Scheirer**[1]
[1]Department of Computer Science and Engineering,
University of Notre Dame, Notre Dame, IN 46556
[2]Department of Applied and Computational Mathematics and Statistics,
University of Notre Dame, Notre Dame, IN 46556
sabraha2@nd.edu, hauenstein@nd.edu, walter.scheirer@nd.edu

Abstract

Fine-tuning large-scale pre-trained models presents inherent challenges related to computational complexity and resource inefficiency. In this paper, we introduce Geodesic Low-Rank Adaptation (GLRA), a novel conceptual framework designed to rethink how fine-tuning occurs in deep neural networks. Rather than relying on traditional methods of parameter updates that may fall prey to sharp minima and unstable convergence, GLRA utilizes geodesic paths to smooth transitions within the weight space. Combined with low-rank adaptation, GLRA seeks to minimize computational overhead while promoting flatter minima, potentially improving generalization and stability in fine-tuning. This paper focuses on exploring the theoretical implications of geodesic interpolation, hypothesizing that this method can provide new insights into efficient model adaptation. We demonstrate through mathematical reasoning how GLRA can enhance model stability by avoiding sharp transitions in the optimization landscape. While experimental validation is left as future work, the conceptual framework we introduce opens a pathway for research into the intersection of geometry and parameter-efficient learning, inviting further investigation into its potential.

1 Introduction

Fine-tuning large-scale pre-trained models, such as BERT and GPT, has become a critical process in modern machine learning. These models, often containing billions of parameters, are designed to generalize across various tasks with minimal adjustments. However, this adaptability comes with significant computational overhead and memory costs, particularly when deployed in resource-constrained environments like mobile devices or edge computing platforms. Fine-tuning typically involves updating a large portion of the model's parameters, resulting in high memory usage and prolonged training times.

To address this, parameter-efficient fine-tuning techniques such as Low-Rank Adaptation (LoRA) have been introduced, significantly reducing the number of trainable parameters while preserving task-specific performance. LoRA achieves this by decomposing weight updates into low-rank matrices, lowering the computational complexity from full model updates to more manageable ranks. While LoRA reduces memory and computational requirements, it still relies on linear interpolation between pre-trained and fine-tuned weights, which may lead to convergence to sharp minima in the optimization landscape.

38th Workshop on Fine-Tuning in Machine Learning (NeurIPS 2024).

In this paper, we introduce *Geodesic Low-Rank Adaptation (GLRA)*, a novel approach that leverages geodesic paths to smooth the transition between pre-trained and fine-tuned model states. Geodesic interpolation, in contrast to linear interpolation, provides a continuous and smooth trajectory in the weight space, avoiding the sharp transitions that could lead to unstable optimization. By combining geodesic interpolation with the parameter-efficient nature of LoRA, GLRA aims to improve model stability and generalization while maintaining low computational costs.

In the absence of large-scale experimental results, we present GLRA as a conceptual framework with strong theoretical motivations. Our approach bridges the gap between geometry and fine-tuning, positing that geodesic paths encourage flatter minima in the optimization landscape, which may yield better generalization and more robust performance in practical applications. This work invites the community to explore the intersection of differential geometry and parameter-efficient fine-tuning, opening new avenues for future research in scalable model adaptation.

2 Related Work

Fine-tuning large-scale pre-trained models has played a pivotal role in transferring knowledge across domains. Traditional fine-tuning, which involves updating all model parameters, has proven effective but at the cost of significant computational overhead and memory usage, especially as model sizes continue to grow [Khetan and Karnin, 2020]. These challenges have prompted the development of parameter-efficient fine-tuning methods.

One prominent approach is Low-Rank Adaptation (LoRA) [Hu et al., 2021], which reduces the number of trainable parameters by restricting updates to low-rank matrices. LoRA allows for significant memory and computation savings while retaining much of the task-specific performance. However, LoRA, like many fine-tuning methods, operates within the framework of traditional gradient-based optimization, which does not explicitly control the trajectory that parameters follow during optimization. This lack of control can lead to suboptimal convergence paths that may result in sharp minima, negatively impacting generalization [Zhou et al., 2020].

AdapterFusion [Pfeiffer et al., 2020] and BitFit [Zaken et al., 2021] are further refinements that focus on minimizing trainable parameters through the use of additional modular layers or bias-term updates. While these techniques are highly efficient and offer strong task-specific performance, they similarly depend on linear or stepwise parameter updates. These methods do not inherently account for the geometry of optimization trajectories, which could play a critical role in avoiding sharp minima and enhancing stability.

The geometry of optimization has recently emerged as a promising area of research. Studies on stochastic weight averaging [Izmailov et al., 2018] and the exploration of non-linear weight space paths highlight the importance of guiding optimization toward flatter minima, which are known to promote better generalization [Hochreiter and Schmidhuber, 1997]. While these methods have been incorporated into specific phases of training, they have not yet been fully explored as a framework for fine-tuning large-scale pre-trained models.

Geodesic Low-Rank Adaptation (GLRA) is proposed as a solution to this gap. GLRA offers a departure from purely gradient-based optimization by explicitly controlling the path between pre-trained and fine-tuned weights through geodesic interpolation. This enables smooth, non-linear transitions in the parameter space, contrasting with the sharp transitions often encountered in linear fine-tuning methods. By incorporating LoRA's parameter-efficient updates, GLRA maintains computational efficiency while using geodesic paths to traverse weight space in a way that favors flatter minima. This not only improves convergence stability but also has the potential to enhance generalization performance.

In summary, GLRA aims to build on the success of LoRA by adding an explicit focus on the geometric properties of the optimization process. The introduction of geodesic interpolation provides smoother transitions in the weight space, which we hypothesize leads to better generalization and more stable fine-tuning outcomes compared to existing linear or stepwise approaches.

3 Theoretical Motivation

The motivation for Geodesic Low-Rank Adaptation (GLRA) arises from two main insights in optimization: the importance of smooth trajectories in the loss landscape and the benefits of parameter-efficient methods for fine-tuning. This section explores the theoretical benefits of geodesic interpolation over linear interpolation, particularly with respect to generalization, convergence, and computational efficiency, while acknowledging the inherent challenges.

3.1 Loss Landscape and Sharp Minima

It has been well-established that the shape of the loss landscape plays a crucial role in determining a model's generalization ability [Hochreiter and Schmidhuber, 1997, Keskar et al., 2016]. Sharp minima, characterized by steep and narrow regions, can lead to poor generalization because small perturbations in model parameters can cause significant increases in the loss function. Conversely, flat minima are more stable and less sensitive to noise, thus improving the model's generalization performance.

Traditional fine-tuning methods, which rely on gradient-based optimization without explicit control over parameter trajectories, often converge to sharp minima due to the abrupt nature of parameter updates [Li et al., 2018]. By contrast, geodesic paths, which follow smooth and continuous curves, are less likely to lead to sharp transitions in the optimization landscape. These smooth paths help guide the model toward flatter minima, thereby improving generalization [Draxler et al., 2018, Garipov et al., 2018].

3.2 Geometric Path Advantage

Geodesic paths in weight space offer several geometric advantages over linear interpolation. Linear paths, while simple to compute, may cause discontinuities in the optimization process due to abrupt transitions. Geodesic paths, on the other hand, minimize the distance traveled on a curved manifold, such as the non-Euclidean parameter space of deep neural networks [Amari and Amari, 1985, Nielsen, 2018]. By following a spherical geodesic trajectory, GLRA ensures a smoother transition between pre-trained and fine-tuned weights, avoiding sharp changes in gradient magnitude and potentially improving stability [Zhang et al., 2019].

The geodesic distance between two points in parameter space, θ_0 (pre-trained) and θ_{LoRA} (fine-tuned), can be expressed as:

$$d(\theta_0, \theta_{\text{LoRA}}) = \int_0^1 \left\| \frac{d\theta(t)}{dt} \right\| dt, \tag{1}$$

where $\frac{d\theta(t)}{dt}$ represents the rate of change along the geodesic path. This formulation minimizes abrupt changes, resulting in smoother parameter updates that align with the natural geometry of the parameter space.

However, it is important to note that the notion of "distance" in parameter space is not absolute due to inherent symmetries in neural networks, such as permutation invariance of neurons [Entezari et al., 2022, Brea and Senn, 2019]. For example, rearranging neurons in a layer can yield a different parameter configuration that produces the same network output. Thus, while geodesic paths provide a practical way to achieve smoother transitions, they do not necessarily represent the true "distance" between functionally equivalent configurations in parameter space.

3.3 Curvature and Optimization Stability

The curvature of the loss landscape, as described by the eigenvalues of the Hessian matrix, plays a significant role in optimization stability [Sagun et al., 2016, Yao et al., 2020]. Large positive eigenvalues indicate sharp curvature, leading to rapid changes in loss with small parameter shifts, while smaller eigenvalues indicate flatter regions that tend to generalize better.

Geodesic interpolation inherently avoids the sharp curvature often encountered during linear interpolation by following the smoothest possible trajectory between two points on the loss surface. This effect can be understood through the second-order derivative of the loss function, i.e., the Hessian $\nabla^2 L(\theta)$. By guiding the optimization process away from regions of sharp curvature, GLRA promotes

convergence toward flatter minima, where $\nabla^2 L(\theta) \approx 0$, leading to improved generalization [Foret et al., 2021, Cha et al., 2021].

Despite this, we acknowledge that geodesic paths are approximate representations due to parameter symmetries and permutation invariance in neural networks [Brea and Senn, 2019, Entezari et al., 2022]. The practical benefit of geodesic interpolation lies in reducing abrupt changes, which empirically aligns with the goal of reaching flatter minima, even though it may not fully address all symmetry issues.

3.4 Optimal Transport in Weight Space

From a theoretical perspective, geodesic interpolation can be viewed as a form of optimal transport in parameter space. Optimal transport theory seeks to find the most efficient way to move between two distributions, minimizing the energy required for the transition [Villani, 2009, Peyré and Cuturi, 2019]. Similarly, geodesic paths minimize the distance traveled in weight space, leading to more efficient and stable updates.

This approach aligns with the principles of optimal transport, as the path taken between θ_0 and θ_{LoRA} is the shortest possible curve on the parameter manifold. However, given the permutation invariance of neurons in neural networks, this shortest path is an approximation rather than an absolute minimum. This efficiency in parameter updates can still reduce the overall computational overhead of fine-tuning large models, making GLRA both effective and resource-efficient [Gordon et al., 2022].

3.5 Flat Minima and Generalization

Theoretical works have shown that flatter minima tend to generalize better because they make the model less sensitive to perturbations in the data or parameters [Hochreiter and Schmidhuber, 1997, Neyshabur et al., 2017]. By using geodesic-inspired paths that avoid sharp changes in the optimization landscape, GLRA helps models converge to these flatter minima, which are characterized by:

$$\nabla^2 L(\theta) \approx 0.$$

This theoretical insight suggests that by promoting flatter minima, GLRA not only enhances model stability but also contributes to better generalization on unseen data [Keskar et al., 2016, Yao et al., 2020]. Nonetheless, due to the permutation invariance of neural networks, GLRA's ability to converge to a true flat minimum is approximate. Future work may explore more sophisticated methods that explicitly address functional equivalence to provide a more rigorous path to flat minima.

4 Methodology: Geodesic Low-Rank Adaptation (GLRA)

In this section, we introduce *Geodesic Low-Rank Adaptation (GLRA)*, a fine-tuning approach that combines geodesic-inspired interpolation with low-rank adaptation for efficient, generalizable training of large-scale models. Unlike traditional fine-tuning that relies solely on gradient updates, GLRA incorporates spherical geodesic interpolation to enable smooth transitions in parameter space, aimed at avoiding sharp minima that can hinder generalization [Hochreiter and Schmidhuber, 1997, Keskar et al., 2016].

4.1 Geodesic-inspired Interpolation

Standard fine-tuning typically updates parameters iteratively without explicit control over the trajectory between pre-trained and fine-tuned weights, which can converge to sharp minima due to abrupt parameter changes [Li et al., 2018]. In contrast, GLRA employs spherical linear interpolation (SLERP), an efficient approximation of geodesic paths, to provide a continuous, smooth trajectory in the high-dimensional weight space from pre-trained weights, θ_0, to low-rank adapted weights, θ_{LoRA} [Amari and Amari, 1985].

The SLERP path is defined as:

$$\theta(t) = \frac{\sin((1-t)\cdot\omega)}{\sin(\omega)} \cdot \theta_0 + \frac{\sin(t\cdot\omega)}{\sin(\omega)} \cdot \theta_{\text{LoRA}} \tag{2}$$

where $t \in [0, 1]$ is the interpolation parameter, and $\omega = \arccos\left(\frac{\theta_0 \cdot \theta_{\text{LoRA}}}{\|\theta_0\| \|\theta_{\text{LoRA}}\|}\right)$ is the angle between θ_0 and θ_{LoRA}, calculated using the dot product [Draxler et al., 2018]. This interpolation smooths transitions during fine-tuning, promoting flatter minima which have been linked to improved generalization [Garipov et al., 2018].

Though SLERP is an efficient approximation of true geodesics, it provides a suitable balance between computational efficiency and stability, avoiding sharp curvature in the weight space [Zhang et al., 2019].

4.2 Low-Rank Adaptation (LoRA)

GLRA also leverages *Low-Rank Adaptation (LoRA)*, which reduces memory and computational costs by introducing low-rank updates to the weights. LoRA decomposes the update into two matrices, $A \in \mathbb{R}^{d \times r}$ and $B \in \mathbb{R}^{r \times d}$, where $r \ll d$. The weight update is thus expressed as:

$$W_{\text{LoRA}} = W_0 + \alpha \cdot AB \tag{3}$$

where W_0 is the original pre-trained weight matrix, and α is a scaling factor. This decomposition reduces the parameter count from $O(d^2)$ to $O(d \cdot r)$, retaining performance while improving efficiency [Hu et al., 2021].

4.3 Algorithm: Geodesic Low-Rank Adaptation (GLRA)

The *Geodesic Low-Rank Adaptation (GLRA)* algorithm integrates spherical geodesic interpolation with low-rank adaptation for effective model fine-tuning. The procedure is outlined below:

Algorithm 1 Geodesic Low-Rank Adaptation (GLRA)

Require: Pre-trained weights θ_0, LoRA weights θ_{LoRA}, homotopy parameter $t \in [0, 1]$, learning rate η

1: Compute the angle $\omega = \arccos\left(\frac{\theta_0 \cdot \theta_{\text{LoRA}}}{\|\theta_0\| \|\theta_{\text{LoRA}}\|}\right)$

2: Perform geodesic interpolation using SLERP to obtain $\theta(t)$:

$$\theta(t) = \frac{\sin((1-t) \cdot \omega)}{\sin(\omega)} \cdot \theta_0 + \frac{\sin(t \cdot \omega)}{\sin(\omega)} \cdot \theta_{\text{LoRA}}$$

3: Initialize the model with $\theta(t)$

4: **for** each training iteration **do**

5: Compute gradients $\nabla_{\theta(t)} \mathcal{L}(\theta(t))$ via backpropagation

6: Update $\theta(t)$ with a gradient-based step:

$$\theta(t) \leftarrow \theta(t) - \eta \cdot \nabla_{\theta(t)} \mathcal{L}(\theta(t))$$

7: **end for**

8: Gradually adjust t over epochs, increasing from 0 to 1

9: **return** Final fine-tuned model parameters $\theta(t)$

By incrementally adjusting t from 0 to 1, GLRA allows a smooth transition from the pre-trained state θ_0 to the adapted state θ_{LoRA}, facilitating a stable and effective optimization pathway [Villani, 2009, Peyré and Cuturi, 2019].

5 Limitations and Future Directions

While Geodesic Low-Rank Adaptation (GLRA) offers a promising approach to fine-tuning large-scale models, there are several limitations to consider. In this section, we outline these challenges and propose potential solutions, along with directions for future research.

One primary concern with GLRA is the additional computational complexity introduced by geodesic interpolation. Although the geodesic path promotes smoother transitions in parameter space, calculating these paths in high-dimensional spaces can be computationally expensive, especially for large models. This additional cost could potentially offset the savings gained through Low-Rank Adaptation (LoRA), which is designed to reduce the number of trainable parameters. To mitigate this, efficient approximations of geodesic paths, such as lower-order interpolations, could be explored. Another option is to apply geodesic interpolation selectively during critical phases of training, while using simpler methods during less critical epochs.

Another limitation is the reduced expressivity inherent in low-rank adaptations. While LoRA helps to cut down the number of trainable parameters, it might restrict the model's ability to capture complex task-specific nuances, particularly in domains that require high flexibility. This limitation could be addressed by allowing adaptive rank selection during training, where the rank of the low-rank matrices A and B could vary based on task complexity. This approach would enable the model to increase expressivity when necessary, balancing parameter efficiency with performance.

The homotopy parameter t, which governs the interpolation between the pre-trained weights θ_0 and the fine-tuned weights θ_{LoRA}, introduces another challenge. The optimal schedule for adjusting t is highly task-dependent and requires careful tuning. Poor choices for this parameter could lead to ineffective optimization paths. Adaptive techniques, such as reinforcement learning-based parameter tuning, could dynamically adjust t based on the model's validation performance, thus eliminating the need for extensive manual tuning.

There are also theoretical concerns related to convergence. While geodesic paths offer smoother transitions through parameter space, there are no guarantees that they lead to global minima, especially given the non-convex nature of deep learning loss landscapes. Geodesic paths may not be universally effective in avoiding saddle points or local minima. Future work could explore combining GLRA with second-order optimization methods, such as quasi-Newton methods or trust-region algorithms, which could enhance convergence properties while leveraging the smoothness of geodesic paths.

A key theoretical limitation of GLRA is its reliance on distance in parameter space, which does not fully address functional equivalences due to the permutation invariance of parameters, particularly in neural networks. In other words, different configurations of weights and neurons can lead to functionally equivalent models, yet appear distant in parameter space. This presents a challenge for GLRA, as it may misinterpret functionally similar models as being distant due to parameter permutations. Future research could explore the use of more advanced, functionally aware distance metrics, potentially grounded in optimal transport or functional similarity, to enhance GLRA's interpretability and effectiveness. Addressing this limitation could improve the robustness and utility of GLRA, especially for models with high permutation invariance.

Finally, GLRA may face scalability issues when applied to extremely large models, such as GPT-3. Although LoRA reduces the number of trainable parameters, computing geodesic paths in such high-dimensional spaces can still be costly. Distributed training techniques, where geodesic interpolation and gradient calculations are distributed across multiple GPUs, could alleviate some of this overhead. Memory-efficient fine-tuning strategies, such as adapters or block-structured low-rank updates, could further enhance scalability.

6 Preliminary Proof-of-Concept Experiments

To provide an initial validation of the proposed Geodesic Low-Rank Adaptation (GLRA) method, we conduct preliminary toy experiments on the MNIST dataset using a simple Multi-Layer Perceptron (MLP). These experiments serve as a proof of concept to highlight the potential benefits of GLRA in comparison to standard Low-Rank Adaptation (LoRA) and do not constitute the full evaluation.

6.1 Experimental Setup

We use the MNIST dataset, consisting of grayscale images of handwritten digits (0-9) with a resolution of 28x28 pixels. For these experiments, we preprocess the images by normalizing them to have zero mean and unit variance.

The model used is a three-layer MLP with ReLU activations:

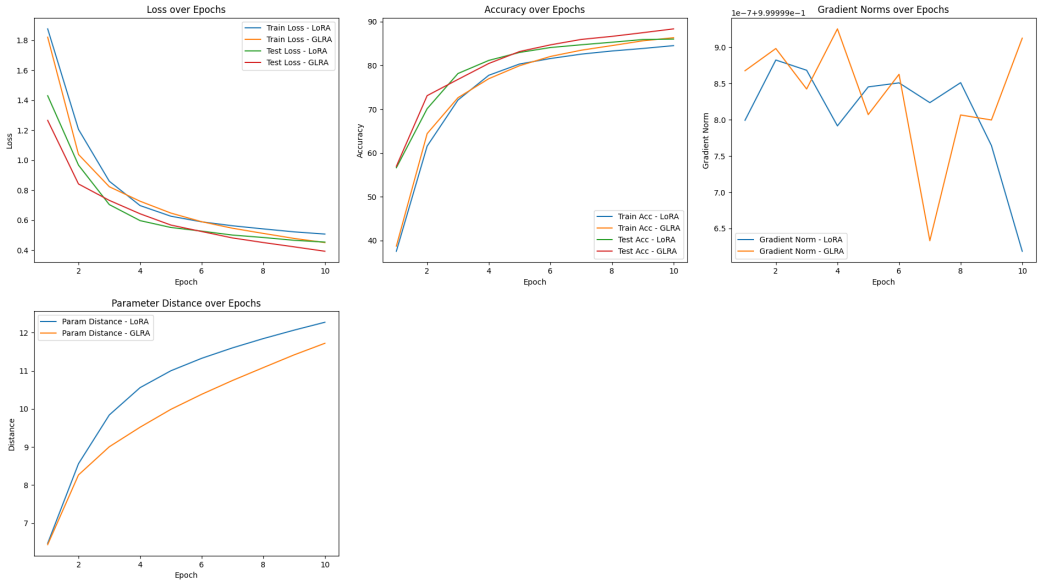

Figure 1: Results of preliminary experiments on MNIST. (a) Loss over epochs, (b) Accuracy over epochs, (c) Gradient norms, and (d) Parameter distances for LoRA and GLRA.

- Input Layer: 784 units (28x28 flattened image)
- Hidden Layer 1: 128 units
- Hidden Layer 2: 64 units
- Output Layer: 10 units (one per class)

6.2 Toy Adaptation Methods

LoRA Configuration. LoRA is applied to the first and second hidden layers with a low-rank dimension of $r = 8$ and scaling factor $\alpha = 16$. The goal is to reduce the number of trainable parameters while preserving model capacity.

GLRA with SLERP Interpolation. In GLRA, we use spherical linear interpolation (SLERP) to gradually interpolate between the pre-trained parameters θ_0 and the LoRA-adapted parameters θ_{LoRA}. This is intended to provide a smoother transition in parameter space and potentially achieve flatter minima. The interpolation parameter t is adjusted from 0 to 1 over a fixed number of stages.

6.3 Evaluation Metrics

Given the exploratory nature of these experiments, we monitor the following key metrics to assess the impact of GLRA:

- **Training and Test Loss**: To observe model convergence and generalization.
- **Accuracy**: Measured on both the training and test sets to evaluate performance.
- **Gradient Norms and Parameter Distance**: These metrics track the smoothness of parameter updates and the deviation from the original parameters.

6.4 Results and Observations

The results from these toy experiments are summarized in Fig. 1. While limited in scope, these preliminary findings demonstrate that GLRA achieves faster convergence and slightly improved test accuracy compared to LoRA. Notably, GLRA exhibits more stable gradient norms and a smaller overall parameter shift, suggesting a smoother adaptation path.

These toy experiments provide early evidence that GLRA's geodesic-inspired interpolation can offer benefits in terms of stability and generalization, even in small-scale setups. However, comprehensive evaluation on more complex tasks and larger models is necessary to fully establish its effectiveness, as presented in the subsequent experimental sections.

7 Experimental Setup

While full experimental validation of Geodesic Low-Rank Adaptation (GLRA) is left as future work, we outline the experimental setup we anticipate using to validate the efficiency and effectiveness of this approach.

Datasets: We plan to evaluate GLRA on standard natural language processing (NLP) benchmarks, such as the Stanford Sentiment Treebank (SST-2) and the Multi-Genre Natural Language Inference (MNLI) dataset.

Baselines: Comparisons will be drawn against traditional fine-tuning methods and parameter-efficient techniques, including Low-Rank Adaptation (LoRA), AdapterFusion, and BitFit.

Metrics: Performance will be measured based on model accuracy, convergence speed, memory usage, and computational overhead. Generalization will be evaluated by assessing the model's performance on unseen data.

8 Future Work

This paper explores the theoretical underpinnings of Geodesic Low-Rank Adaptation (GLRA) and there are several avenues for future research and experimentation. Key areas include:

- **Empirical Validation:** Rigorous experimental validation is needed to evaluate the practical benefits of GLRA across different tasks and domains. This includes both language models and computer vision applications.
- **Hyperparameter Tuning:** An exploration of the effect of the homotopy parameter t and how to dynamically adjust it during training is critical for optimizing the performance of GLRA.
- **Extensions to Other Architectures:** GLRA could be extended to other architectures beyond transformers, such as convolutional neural networks (CNNs) or recurrent neural networks (RNNs), to examine its generality.
- **Optimization of Geodesic Paths:** Further work can be done to explore alternative geodesic paths or interpolation techniques that might be more efficient in different settings.

By addressing these open questions, we aim to provide a more comprehensive understanding of the benefits and limitations of incorporating geometric principles in the fine-tuning process.

References

Shun-ichi Amari and Shun-ichi Amari. α-divergence and α-projection in statistical manifold. *Differential-Geometrical Methods in Statistics*, pages 66–103, 1985.

Johanni Brea and Walter Senn. Weight-space symmetry in neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 30(1):119–128, 2019.

Junsoo Cha, Joong-Ho Lee, Jinwoo Shin, and Sung Ju Hwang. Swad: Stochastic weight averaging in deep learning. In *International Conference on Learning Representations (ICLR)*, 2021.

Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred A Hamprecht. Essentially no barriers in neural network energy landscape. In *International Conference on Machine Learning (ICML)*, pages 1309–1318, 2018.

Nazanin Entezari, Mahdi Soltanolkotabi, and Anima Anandkumar. The role of permutation invariance in neural networks. *arXiv preprint arXiv:2204.11271*, 2022.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations (ICLR)*, 2021.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Alex Gordon et al. Smooth interpolations between weight spaces. *arXiv preprint arXiv:2201.05662*, 2022.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997.

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. *arXiv preprint arXiv:2106.09685*, 2021.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Ashish Khetan and Zohar Karnin. schubert: Optimizing elements of bert. *arXiv preprint arXiv:2005.06628*, 2020.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nathan Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Frank Nielsen. Natural gradient and applications in deep learning. *Springer*, pages 257–292, 2018.

Gabriel Peyré and Marco Cuturi. *Computational Optimal Transport*. Now Publishers Inc., 2019.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. Adapterfusion: Non-destructive task composition for transfer learning. *arXiv preprint arXiv:2005.00247*, 2020.

Levent Sagun, Utku Evci, V Ugur Guney, Yann Dauphin, and Leon Bottou. Eigenvalues of the hessian in deep learning: Singularity and beyond. *arXiv preprint arXiv:1611.07476*, 2016.

Cédric Villani. *Optimal Transport: Old and New*, volume 338. Springer, 2009.

Zhiyuan Yao, Amir Gholami, Sheng Shen, Kurt Keutzer, and Michael W Mahoney. Pyhessian: Neural networks through the lens of the hessian. *arXiv preprint arXiv:2002.06419*, 2020.

Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021.

Hongyi Zhang, Yann N Dauphin, and Tengyu Ma. All you need is a good init. *International Conference on Learning Representations (ICLR)*, 2019.

Pan Zhou, Jiashi Feng, Chao Ma, Caiming Xiong, Steven Chu Hong Hoi, et al. Towards theoretically understanding why sgd generalizes better than adam in deep learning. *Advances in Neural Information Processing Systems*, 33:21285–21296, 2020.