Preserving Principal Subspaces to Reduce Catastrophic Forgetting in Fine-tuning

Jörg K.H. Franke
University of Freiburg
Freiburg, Germany
frankej@cs.uni-freiburg.de

Michael Hefenbrock
RevoAI
Karlsruhe, Germany

Frank Hutter
ELLIS Institute Tübingen & University of Freiburg
Tübingen/Freiburg, Germany

Abstract

In this paper, we address catastrophic forgetting in fine-tuning Large Language Models (LLMs), a process where LLMs lose knowledge and capabilities upon learning new information. Traditional solutions mostly rely on reusing old training data. Such methods are often limited by knowledge about previously used data and possibly limited access to it. In contrast to these approaches, we propose a new strategy focusing on the model's weight matrices. Using Singular Value Decomposition (SVD), we seek to identify and preserve key components within these matrices, particularly the highest magnitude directions, to preserve the most sensitive characteristics. Our approach thus uniquely focuses updates on the space spanned by lower-impact directions. This methodology efficiently mitigates catastrophic forgetting and does not require access to the original training data, offering a more practical solution for LLM fine-tuning applications as it is simpler and more training data efficient. We show the benefit of our approach by fine-tuning an LLM and reducing the performance drop on benchmark tasks induced by fine-tuning.

1 Introduction

In the evolving landscape of Large Language Models (LLMs), fine-tuning has emerged as a critical process, substantially enhancing their specificity and effectiveness across diverse applications. However, the fine-tuning process often encounters a significant impediment known as catastrophic forgetting. In this phenomenon, the model, upon learning new tasks, tends to lose its grip on previously acquired knowledge, alignment, or reasoning capabilities (Kirkpatrick et al., 2017; Zhai et al., 2023). This issue poses a substantial challenge, especially considering the dynamic nature of tasks that LLMs are expected to perform. Traditionally, methods to counteract catastrophic forgetting have relied heavily on utilizing old training data (Parisi et al., 2019; Wang et al., 2021). Similarly, the continuous learning literature offers some respite by suggesting the incorporation of data from previous tasks (Kirkpatrick et al., 2017; Parisi et al., 2019; Saha et al., 2021; Kong et al., 2022). Such approaches, while effective to an extent, face practical limitations. Selecting an appropriate surrogate dataset becomes a complex task, as it requires an understanding of the original training set's distribution, which is typically unknown (Peng et al., 2023). Especially, in the context of fine-tuning recently published LLMs like Mistral7B (Jiang et al., 2023) or LLaMA2 (Touvron et al., 2023) further limitations emerge. Here, access to original training data is either restricted or entirely unavailable. This unavailability is not just a matter of access but also concerns privacy and proprietary information. Additionally, preserving the nuances of instruction tuning, especially in models trained with reinforcement learning from human feedback (RLHF), adds another layer of complexity.

In this paper, we propose *Principal Subspace Preserving* (PSP), a novel approach to address catastrophic forgetting, pivoting away from the conventional reliance on previously used data. Our method focuses on the weight matrices intrinsic to the model. By employing Singular Value Decomposition (SVD), we identify critical components within these matrices - the high magnitude directions. We hypothesize that, by preserving significant subspaces and focussing updates on subspaces with lower impact, we can mitigate the effects of catastrophic forgetting. This approach does not require access to old training data, making it more practical and applicable in various settings. We show the benefit of PSP by fine-tuning an LLM and evaluating the performance before and after the fine-tuning on a set of benchmark tasks (Section 3).

2 METHODOLOGY

2.1 THE SINGULAR VALUE DECOMPOSITION

The singular value decomposition (SVD) is one of the fundamental decompositions of a matrix. It decomposes a matrix $\boldsymbol{W} \in \mathbb{R}^{n \times m}$ in the following way:

$$\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top, \quad \text{with} \quad \boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}_n \quad \text{and} \quad \boldsymbol{V}^\top\boldsymbol{V} = \boldsymbol{I}_m.$$

Here, $\boldsymbol{\Sigma}$ is a diagonal matrix with entries σ_k, $k = 1, \cdots, K$ where $K = \min\{m, n\}$. The values σ_k are referred to as singular values and $\boldsymbol{U}$ and $\boldsymbol{V}$ are arranged such that the singular values display a descending order, with $\sigma_k > \sigma_{k+1}$. One fact underlining the importance of the SVD is its close connection to eigenvalues and eigenvectors. Specifically, the columns of $\boldsymbol{V}$ are the eigenvectors of $\boldsymbol{W}^\top\boldsymbol{W}$ and the columns of $\boldsymbol{U}$ are the eigenvectors of $\boldsymbol{W}\boldsymbol{W}^\top$. Consequently, the respective (nonzero) eigenvalues of both are given by the σ_k^2. The SVD can thus be used similar to the eigenvalue decomposition (or PCA) while also being applicable to nonsquare matrices.

The SVD provides a set of orthogonal basis vectors for the row and column space of $\boldsymbol{W}$ in the form of $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively. Through these bases, the impact of a right (or left) multiplication of a $\boldsymbol{x}$ with the matrix $\boldsymbol{W}$ can be analyzed and bounded: $\sigma_K \cdot \|\boldsymbol{x}\|_2 \leq \|\boldsymbol{W}\boldsymbol{x}\|_2 \leq \sigma_1 \cdot \|\boldsymbol{x}\|_2$. The upper bound is tight for $\boldsymbol{x} = \boldsymbol{v}_1$, while the lower bound is tight for $\boldsymbol{x} = \boldsymbol{v}_K$. For a normed $\boldsymbol{x} \in \mathbb{R}^m$, $\|\boldsymbol{W}\boldsymbol{x}\|_2$ depends on its dot product similarity with the columns of $\boldsymbol{V}$ (basis vectors). The same analysis can be carried out for a vector $\boldsymbol{y} \in \mathbb{R}^n$ with $\|\boldsymbol{y}^\top\boldsymbol{W}\|_2$ using the columns of $\boldsymbol{U}$.

2.2 PRESERVING PRINCIPAL SUBSPACES

In the following, we seek to leverage the SVD to preserve key characteristics of the map $\boldsymbol{W}\boldsymbol{x}$ of an input vector $\boldsymbol{x}$ with a weight matrix $\boldsymbol{W}$. Specifically, we want to preserve directions (subspaces) relating to singular values of the highest magnitude.

This is motivated by the following hypothesis: *To limit the performance loss on previous tasks, the output magnitude along directions that lead to comparably large effects on outputs should be preserved, as these carry the most defining features.*

Following this idea, we limit the updates of weight matrices to act only on the subspace spanned by directions of lower output magnitude. In the scope of this work, the concrete selection of these may be based on certain heuristics. For example, direction $\boldsymbol{U}$ and $\boldsymbol{V}$ relating to the $p\%$ largest singular values may be excluded. Let $\bar{k}$ with $1 < \bar{k} < K$, denote the lowest index associated with the remaining components. Since the singular values are given in descending order, the first index points to the singular value with the highest impact. For brevity, we use $\bar{\boldsymbol{V}} = \boldsymbol{V}[:, \bar{k}:]$ and $\bar{\boldsymbol{U}} = \boldsymbol{U}[:, \bar{k}:]$ in the following. Given an update $\Delta\boldsymbol{W}$ for a weight matrix $\boldsymbol{W}$ in step t, we can modify it to only act on the specified directions by

$$\boldsymbol{W}_{t+1} \leftarrow \boldsymbol{W}_t + \bar{\boldsymbol{U}}\bar{\boldsymbol{U}}^\top \Delta\boldsymbol{W} \bar{\boldsymbol{V}}\bar{\boldsymbol{V}}^\top.$$

Hence, the effect of $\boldsymbol{W}$ on vectors lying in the subspace spanned by the excluded components is preserved, as they are orthogonal to the update direction. For the purpose of fine-tuning, we refer to this idea as *principal subspace preserving* (PSP) fine-tuning. Since fine-tuning of large language models is often done using low-rank approximations of the form $\boldsymbol{B}\boldsymbol{A}$ (LoRA) (Hu et al., 2021), we present an adaptation of this idea for LoRA in Algorithm 1.

Algorithm 1 Principal Subspace Preserving fine-tuning with LoRA

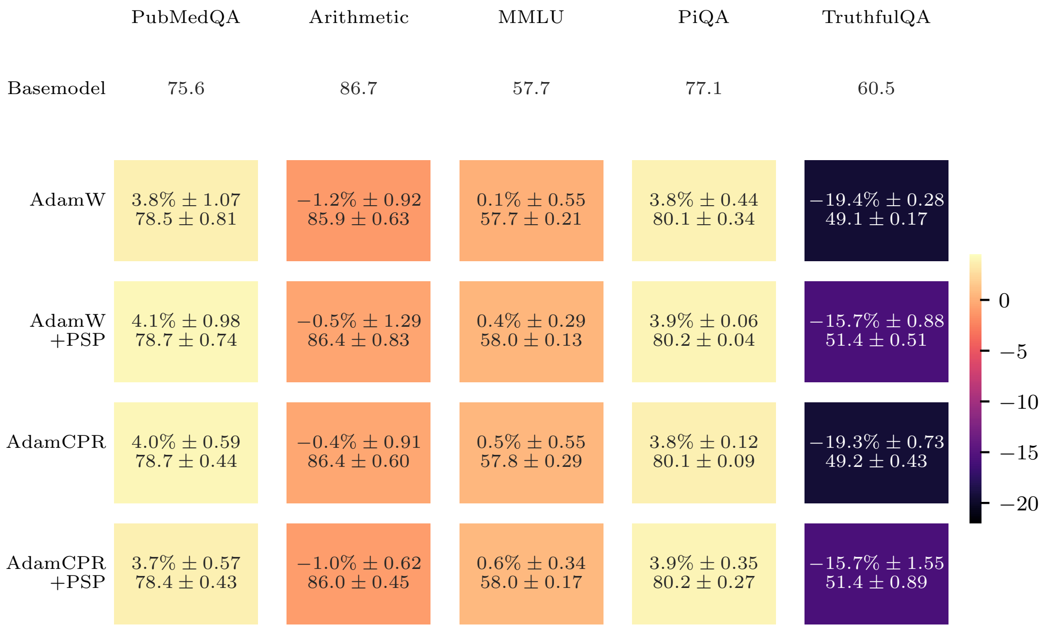
```
Require: Pre-trained parameters $\boldsymbol{\theta}^j$, trainable parameters $\boldsymbol{B}^j$ and $\boldsymbol{A}^j$ $j = 1, \cdots, J$, loss function
    $L(\boldsymbol{B}, \boldsymbol{A}, \boldsymbol{X}, \boldsymbol{y})$ and a fine-tuning dataset $\mathcal{D} = \{(\boldsymbol{X}_t, \boldsymbol{y}_t)\}_{t=0}^{T}$.
Require: Hyperparameters: Learning rate $\eta \in \mathbb{R}^+$ and index $\bar{k}$ for component selection.
Require: Optimizer $\mathrm{Opt}(\cdot)$ for minimization.
# Initialization
for each pre-trained parameter $\boldsymbol{\theta}^j$ in $\boldsymbol{\theta}$ do
    $\boldsymbol{B}^j,\ \boldsymbol{A}^j \leftarrow \mathrm{Initialize}(\cdot)$
    $\boldsymbol{U}^j \boldsymbol{\Sigma}^j \boldsymbol{V}^{j\top} \leftarrow \mathrm{SVD}(\boldsymbol{\theta}^j)$    ▷ Calculating SVD for each pre-trained, trainable parameters
    $\bar{\boldsymbol{U}}^j, \bar{\boldsymbol{V}}^j \leftarrow \boldsymbol{U}^j[:, \bar{k}:], \boldsymbol{V}^j[:, \bar{k}:]$    ▷ Select lower impact directions to be modified
    $\boldsymbol{B}^j,\ \boldsymbol{A}^j \leftarrow \bar{\boldsymbol{U}}^j \bar{\boldsymbol{U}}^{j\top} \boldsymbol{B}^j,\ \boldsymbol{A}^j \bar{\boldsymbol{V}}^j \bar{\boldsymbol{V}}^{j\top}$    ▷ Project initialization
end for

# fine-tuning
$t \leftarrow 0$
for $\boldsymbol{X}_t, \boldsymbol{y}_t \sim \mathcal{D}$ do
    for trainable parameters $\boldsymbol{B}^j$, $\boldsymbol{A}^j$ in $\boldsymbol{B}, \boldsymbol{A}$ do
        $\nabla_{\boldsymbol{A}_t^j} L(\cdot), \nabla_{\boldsymbol{B}_t^j} L(\cdot) \leftarrow \mathrm{Backward}(L(\cdot))$    ▷ Calculate gradients for LoRA parameters
        $\nabla_{\boldsymbol{B}_t^j} L(\cdot) \leftarrow \bar{\boldsymbol{U}}^j \bar{\boldsymbol{U}}^{j\top} \nabla_{\boldsymbol{B}_t^j} L(\cdot)$    ▷ Project update direction (including regularization)
        $\nabla_{\boldsymbol{A}_t^j} L(\cdot) \leftarrow \nabla_{\boldsymbol{A}_t^j} L(\cdot) \bar{\boldsymbol{V}}^j \bar{\boldsymbol{V}}^{j\top}$
        $\boldsymbol{A}_{t+1}^j \leftarrow \boldsymbol{A}_t^j + \mathrm{Opt}(\nabla_{\boldsymbol{A}_t^j} L(\cdot), \eta)$    ▷ Gradient-based update with projected gradients
        $\boldsymbol{B}_{t+1}^j \leftarrow \boldsymbol{B}_t^j + \mathrm{Opt}(\nabla_{\boldsymbol{B}_t^j} L(\cdot), \eta)$
    end for
    $t \leftarrow t + 1$
end for
```

3 Experiments

We evaluate PSP with fine-tuning Mistral7B Jiang et al. (2023) with the use of low-rank adaptation Hu et al. (2021). We choose as a fine-tuning task to train on 50k artificially generated biomedical question-answering (QA) pairs from the PubMedQA dataset (Jin et al., 2019) and evaluate the fine-tuning performance on the expert-annotated *PubMedQA* QA benchmark. The dataset is collected from PubMed abstracts and the artificial QA pairs are generated by converting statement titles into questions and labeled with yes/no answers through a simple heuristic. To access the catastrophic forgetting, we evaluate the LLM before and after the fine-tuning and report the change in answer accuracy in Figure 1. We selected a diverse set of benchmark tasks from the literature. Besides the expert-annotated *PubMedQA* QA instances, we use the *Arithmetic* dataset with 10 tests that involve simple arithmetic problems in natural language (Brown et al., 2020), the comprehensive *MMLU* benchmark (Hendrycks et al., 2020), the *PiQA* benchmark on reasoning about physical commonsense in natural language (Bisk et al., 2020), and the *TruthfulQA* benchmark, which evaluates models' abilities to mimic human falsehoods (Lin et al., 2022).

We optimize our training with the use of Adam (Kingma & Welling, 2014) and evaluate our method using two regularization techniques, decoupled weight decay (AdamW) (Loshchilov & Hutter, 2019) and constrained parameter regularization (AdamCPR) (Franke et al., 2023). We performed a hyperparameter optimization for the learning rate (0.001, *0.0005*, 0.0001) with the AdamW setting, decoupled weight decay for AdamW (1e-4, 1e-3, *1e-2*, 1e-1), and warm start steps for AdamCPR (50, *100*, 200) and select the italic values for each parameter. In preliminary experiments, we parameterized PSP to preserve the subspace associated with 10% to 50% of the highest singular values and select 40% for our experiments. We fine-tune all attention and feed-forward weights, using a learning rate schedule with a learning rate warm-up of 50 or 100 steps ($7\%/14\%$ of total training



	PubMedQA	Arithmetic	MMLU	PiQA	TruthfulQA
Basemodel	75.6	86.7	57.7	77.1	60.5
AdamW	$3.8\% \pm 1.07$ 78.5 ± 0.81	$-1.2\% \pm 0.92$ 85.9 ± 0.63	$0.1\% \pm 0.55$ 57.7 ± 0.21	$3.8\% \pm 0.44$ 80.1 ± 0.34	$-19.4\% \pm 0.28$ 49.1 ± 0.17
AdamW +PSP	$4.1\% \pm 0.98$ 78.7 ± 0.74	$-0.5\% \pm 1.29$ 86.4 ± 0.83	$0.4\% \pm 0.29$ 58.0 ± 0.13	$3.9\% \pm 0.06$ 80.2 ± 0.04	$-15.7\% \pm 0.88$ 51.4 ± 0.51
AdamCPR	$4.0\% \pm 0.59$ 78.7 ± 0.44	$-0.4\% \pm 0.91$ 86.4 ± 0.60	$0.5\% \pm 0.55$ 57.8 ± 0.29	$3.8\% \pm 0.12$ 80.1 ± 0.09	$-19.3\% \pm 0.73$ 49.2 ± 0.43
AdamCPR +PSP	$3.7\% \pm 0.57$ 78.4 ± 0.43	$-1.0\% \pm 0.62$ 86.0 ± 0.45	$0.6\% \pm 0.34$ 58.0 ± 0.17	$3.9\% \pm 0.35$ 80.2 ± 0.27	$-15.7\% \pm 1.55$ 51.4 ± 0.89

Figure 1: The accuracy and percentage of performance change on different benchmarks before and after fine-tuning Mistral 7B with PubMedQA artificial data with the use of AdamW and AdamCPR in combination with PSP. We configure PSP to preserve the subspace associated with the 40% of the highest singular values. We show the mean and standard deviation of the change across three seeds.

steps), followed by cosine annealing. We found a longer learning rate warm-up beneficial when using PSP. The fine-tuning was performed on four A100 GPUs for about 1h. Each configuration is trained across three random seeds and we report the mean change to the base model performance.

The results of our experiment are presented in Figure 1. We see a performance increase on the *PubMedQA* benchmark due to the fine-tuning. Surprisingly, we found that catastrophic forgetting does not appear equally in all benchmark tasks. While the performance drops on *TruthfulQA* and *Arithmetic*, it increases on *PiQA* and slightly on *MMLU*. When applying PSP, we found a less drastic drop in performance on *TruthfulQA* and *Arithmetic*. CPR itself already provides a benefit in terms of avoiding catastrophic forgetting. However, when combined with PSP, it reduces the performance loss on *TruthfulQA* and improves the performance on other benchmarks too. We also see a reduced variance across the three seeds when training with CPR. The downside is a slight drop in performance in the target task.

4 Discussion & Conclusion

We introduce *principal subspace preserving* (PSP) as a promising approach to migrate catastrophic forgetting when fine-tuning a model, where the pre-train data is unknown or not accessible, like in the case of fine-tuning an open-access but proprietary trained LLM. Through an SVD, we identify subspaces relating to the largest singular values of the parameters. We hypothesize that these directions are the most important to preserve to limit the performance decrease when fine-tuning. Consequently, we avoid modifications to these subspaces by projecting updates on spaces spanned by the remaining singular vectors. We show the benefits of PSP, by measuring the performance of an LLM on a set of benchmark tasks before and after fine-tuning. In both settings, when trained with AdamW and AdamCPR, PSP reduces catastrophic forgetting. However, this improvement is accompanied by a trade-off in terms of increased memory requirements due to the need to store the projection matrices $\bar{\boldsymbol{U}}$ and $\bar{\boldsymbol{V}}$. The degree of memory increase is contingent upon the number of components selected for preservation. For future work, the selection of the concrete subspaces or the number of components to be preserved should be investigated further.

REFERENCES

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7432–7439, 2020.

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.

Jörg KH Franke, Michael Hefenbrock, Gregor Koehler, and Frank Hutter. Constrained parameter regularization. *arXiv preprint arXiv:2311.09058*, 2023.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2020.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. PubMedQA: A dataset for biomedical research question answering. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2567–2577. Association for Computational Linguistics, Nov 2019.

D. Kingma and M. Welling. Auto-encoding variational bayes. In *Proceedings of the International Conference on Learning Representations (ICLR'14)*. CBLS, 2014.

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. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

Yajing Kong, Liu Liu, Zhen Wang, and Dacheng Tao. Balancing stability and plasticity through advanced null space in continual learning. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision – ECCV 2022*, pp. 219–236, Cham, 2022. Springer Nature Switzerland.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR'19)*, 2019. Published online: `iclr.cc`.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural networks*, 113:54–71, 2019.

Baolin Peng, Linfeng Song, Ye Tian, Lifeng Jin, Haitao Mi, and Dong Yu. Stabilizing rlhf through advantage model and selective rehearsal. *arXiv preprint arXiv:2309.10202*, 2023.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021.

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. *arXiv preprint arXiv:2307.09288*, 2023.

Shipeng Wang, Xiaorong Li, Jian Sun, and Zongben Xu. Training networks in null space of feature covariance for continual learning. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pp. 184–193, 2021.

Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. Investigating the catastrophic forgetting in multimodal large language model fine-tuning. In *Conference on Parsimony and Learning (Proceedings Track)*, 2023.