OWM-LoRA: Projecting LoRA Gradients onto Orthogonal Subspaces

Yichen Liu [1,2] Liangxuan Guo[1,2] Yuming Dai[1,2] Pengcheng Pan[1,3] Canyang Zhao[1,3] Ziheng Li[1,3] Shan Yu[1,2,3,4†]
[1]Institute of Automation, Chinese Academy of Sciences, Beijing, China.
[2]School of Future Technology, University of Chinese Academy of Sciences, Beijing, China.
[3] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China.
[4]State Key Laboratory of Brain Cognition and Brain-inspired Intelligence Technology

***Abstract*—Large language models (LLMs) have achieved state-of-the-art results across a wide range of NLP tasks, yet they remain prone to catastrophic forgetting when fine-tuned sequentially on multiple tasks. In this paper, we introduce Orthogonal Weight Modification Low-Rank Adaptation (OWM-LoRA), a novel continual learning method that integrates low-rank parameter updates with orthogonal gradient projection. By restricting Low-Rank Adaptation (LoRA) updates to the subspace orthogonal to the gradient directions of previously learned tasks, OWM-LoRA enables the model to acquire new knowledge without perturbing knowledges associated with earlier tasks. Our method retains the parameter efficiency and flexibility of LoRA. We conducted extensive experiments on standard continual-learning benchmarks using various LLMs and demonstrated that OWM-LoRA consistently outperforms existing approaches, significant gains in end-task performance. Furthermore, OWM-LoRA scales gracefully to models with billions of parameters and requires neither storage of historical gradients nor retention of past data, thereby preserving data privacy and computational practicality. These results establish OWM-LoRA as a simple and efficient approach for enabling robust continual learning in LLMs.**

***Index Terms*—Large Language Models, Continual learning, Orthogonal weight modification**

I. Introduction

In recent years, pre-trained models [1], [2] have demonstrated remarkable performance on a wide range of static tasks. However, learning multiple tasks in sequence—commonly referred to as continual learning—remains a significant challenge [3], [4]. As a model learns new tasks, it tends to forget or lose the knowledge it had acquired for earlier tasks, leading to a phenomenon known as *catastrophic forgetting* [5], particularly in LLMs.

To address catastrophic forgetting in continual learning, a variety of prior work [6] has explored mechanisms that balance stability and efficiency across sequential tasks. Existing continual learning approaches can be broadly categorized into regularization-based, replay-based, and optimization-based strategies, alongside approaches focused on representation and architectural adjustments.

Existing Parameter-efficient fine-tuning (PEFT) techniques, notably LoRA [7], have emerged as a promising approach for adapting LLMs by introducing task-specific low-rank adapter matrices updates while freezing the backbone parameters.

† Corresponding author: shan.yu@nlpr.ia.ac.cn

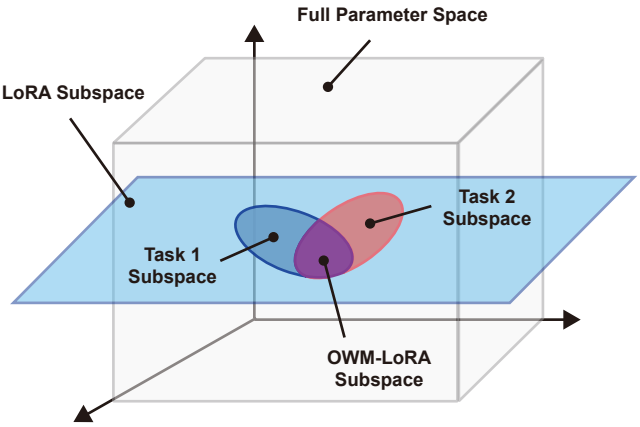


Fig. 1: Illustration of OWM-LoRA. Within the LoRA parameter space, OWM-LoRA constrain each new task's gradient update to remain orthogonal to the subspaces spanned by all previous tasks. The resulting intersection defines the OWM-LoRA parameter subspace, which preserves prior knowledge and markedly reduces catastrophic forgetting.

However, when many tasks arrive sequentially, naively learning every adapter in the same latent subspace entangles their gradient directions and amplifies cross-task interference, an effect that is magnified in LLMs whose full-parameter gradients are prohibitively expensive to store or update. In this work, we propose OWM-LoRA, a simple and effective continual learning method for language models. Specifically, we perform LoRA parameter updates exclusively in the subspace orthogonal to the gradient spanned by previous tasks. By confining parameter update to this orthogonal subspace [8], the model can acquire new knowledge without interfering with representations learned on earlier tasks, thereby mitigating catastrophic forgetting while maintaining the efficiency and flexibility of LoRA. Crucially, operating in the compact LoRA space eliminates the need to cache full-precision gradients or to replay data, delivering a solution that is both memory-efficient and privacy-preserving for LLMs with hundreds of billions of parameters. Figure 1 provides a visual representation of how

OWM-LoRA mitigates catastrophic forgetting.

Our main contributions are as follows:

- We evaluate our orthogonal low-rank adaptation across multiple LLMs and show that it consistently and substantially outperforms previous SOTA methods on established continual-learning benchmarks.
- By applying our method to different model architectures, we observe that OWM-LoRA remains consistently effective. Furthermore, our experimental results demonstrate that tuning the parameters of the feed-forward network (FFN) achieves superior performance on continual learning tasks compared to self-attention layers.

Code and data are provided in the supplement to ensure reproducibility and will be open-sourced upon acceptance.

II. Related Work

A. Continual learning

Continual learning aims to accumulate knowledge from non-stationary data streams while mitigating catastrophic forgetting. Existing methodologies can be broadly consolidated into four primary categories: regularization, replay, architecture, and optimization. **Regularization-based** approaches maintain stability by introducing constraints on weight variations, exemplified by EWC [9], or by preserving input-output mappings via knowledge distillation as seen in LwF [10]. **Replay-based** methods approximate historical distributions by interleaving stored exemplars, such as in Experience Replay (ER) [11]. **Architecture-based** strategies isolate inter-task interference by allocating task-specific parameters, utilizing techniques such as the binary masking in HAT [12]. Finally, **Optimization-based** approaches—which are most relevant to our work—explicitly manipulate gradient directions to prevent interference. Prominent methods like GEM [13] project gradients to satisfy constraints defined by replay data, while orthogonal projection techniques, including Orthog-Subspace [14], restrict parameter updates to subspaces orthogonal to prior inputs. This geometric perspective minimizes disruption to previously learned representations, providing the foundational motivation for our proposed OWM-LoRA.

B. Parameter-Efficient Fine-Tuning

Parameter-efficient fine-tuning (PEFT) is an emerging research direction aimed at optimizing model performance while minimizing computational resources and annotation effort, encompassing various methods such as adapters [15], prompt-tuning [16] and LoRA. Notably, LoRA has been widely adopted due to its ability to adapt large models to new tasks efficiently with minimal additional parameters. Building upon the LoRA method, this study introduces an efficient architecture for sustainable learning by projecting gradients orthogonally to input representations, thereby minimizing interference with previously learned representations, effectively mitigating catastrophic forgetting in continual learning scenarios, and achieving a balance between model performance and computational efficiency.

C. Continual Learning for LLMs

Continual learning for LLMs addresses catastrophic forgetting by employing parameter-efficient adaptations and projection methods [17]. Recent adapter-based techniques such as Attentional Mixture of LoRAs (AM-LoRA) [18] introduce an attention mechanism to dynamically integrate task-specific low-rank updates. Beyond these, Sigmoid-Enhanced CUR Decomposition with Uninterrupted Retention and Low-Rank Adaptation (SECURA) [19] refines LoRA through CUR factorization and novel normalization to further enhance retention alongside performance. Moreover, instance-wise LoRA (iLoRA) [20] customizes adapter parameters per-example via gating modules, reducing negative transfer in sequential recommendation scenarios. Parameter Efficient Gradient Projection (PEGP) [21] injects orthogonal gradient projection into each method to mitigate forgetting with minimal extra memory and training overhead.

III. Method

A. Continual Learning Setup

Continual learning is characterized as learning from dynamic data distributions. In supervised continual learning, a sequence of tasks $\{\mathcal{D}_1, \ldots, \mathcal{D}_T\}$ arrive in a streaming fashion. Each task $\mathcal{D}_t = \{(x_t^i, y_t^i)\}_{i=1}^{n_t}$contains a separate target dataset, where $x_t^i \in \mathcal{X}_t, y_t^i \in \mathcal{Y}_t$. A single model needs to adapt to them sequentially, with only access to $\mathcal{D}_t$ at the t^{th} task. In general, given a prediction model h_Θ parameterized by Θ, continual learning seeks to optimize for the following objective across all tasks:

$$\max_{\Theta} \sum_{k=1}^{\top} \sum_{x,y \in \mathcal{D}_k} \log p_\Theta(y|x) \tag{1}$$

B. Orthogonal Weight Modification

To address catastrophic forgetting, we adopt the Orthogonal Weight Modification (OWM) algorithm [8]. Consider a feed-forward network with layers $l = 0, \ldots, L$. Let $W_l \in \mathbb{R}^{d \times k}$ denote the weight matrix between layer $l-1$ and l. We maintain the row-vector convention, where the input $x_{l-1} \in \mathbb{R}^{1 \times d}$ produces the pre-activation $h_l = x_{l-1} W_l$.

The OWM algorithm introduces an orthogonal projector $P \in \mathbb{R}^{d \times d}$ to the input space of layer l. P is updated recursively to encompass the subspace spanned by inputs from all previous tasks. The procedure is shown in Algorithm 1, which utilizes a Recursive Least Squares (RLS) style update to avoid computationally expensive matrix inversions.

C. Methodological Framework of OWM-LoRA

The proposed OWM-LoRA framework is grounded in the LoRA paradigm [7], which posits that the optimization of pre-trained models can be effectively confined to a low intrinsic dimension. Formally, let $W_0 \in \mathbb{R}^{d \times k}$ denote a frozen pre-trained weight matrix, where d and k represent the input and output dimensions, respectively. Rather than modifying the full parameter space, LoRA parameterizes the weight update ΔW through a low-rank decomposition $\Delta W = AB$, where $A \in$

Algorithm 1 Orthogonal Weight Modification on LoRA

Initialization: Frozen Base Model Weight $W_0 \in \mathbb{R}^{d\times k}$.
Initialize for LoRA Layers: $P \in \mathbb{R}^{d\times d} \leftarrow I$, $A \in \mathbb{R}^{d\times r} \leftarrow \mathcal{N}(0, \sigma^2)$, $B \in \mathbb{R}^{r\times k} \leftarrow 0$, hyperparameter α.
for each input batch X in Batch $\mathcal{B}$ **do**
  Forward Pass:
    $Z \leftarrow (XP)A$ {Project input, then Down-project}
    $\Delta \leftarrow ZB$ {Up-project}
    $H \leftarrow XW_0 + \Delta$
  Projection Matrix Update:
    $r \leftarrow \frac{1}{N_B}\sum_{i\in\mathcal{B}} x_i$ {Mean of input batch}
    $r \leftarrow r/\|r\|_2$
    $k \leftarrow Pr^\top$
    $c \leftarrow 1/(\alpha + rk)$
    $P \leftarrow P - ckk^\top$
end for
Return H

$\mathbb{R}^{d\times r}$ and $B \in \mathbb{R}^{r\times k}$ are trainable matrices with rank $r \ll \min(d, k)$. Adopting the row-vector convention for an input $x \in \mathbb{R}^{1\times d}$, the standard forward pass is computed as:

$$h = x(W_0 + \Delta W) = xW_0 + xAB. \tag{2}$$

This formulation implies that the adaptation term corresponds directly to the projection of the input x through the low-rank weight update ΔW.

To adapt this architecture for continual learning, we incorporate a projection-based constraint designed to mitigate catastrophic forgetting. The core objective is to minimize the interference of current updates with representations established by preceding tasks. We achieve this by projecting the input x onto a subspace orthogonal to the input spaces of prior tasks via a projector P. Consequently, the modified adaptation contribution in the forward pass is formulated as:

$$\text{Adaptation}_{\text{owm}} = xP\Delta W = xPAB, \tag{3}$$

where the projector P effectively filters out components of x that reside within the subspaces of historically learned tasks.

The theoretical justification for this approach rests on the geometric interpretation of forgetting [8], where performance degradation is attributed to parameter updates that distort outputs for historical inputs. By confining the effective contribution of ΔW to the null space of the cumulative input covariance matrix, OWM-LoRA guarantees that the influence of ΔW remains orthogonal to prior inputs. This geometric isolation preserves accumulated knowledge while retaining sufficient plasticity for the model to adapt to novel tasks within the remaining orthogonal directions.

IV. Experiments

A. Datasets

Our evaluation methodology is designed to comprehensively assess continual learning performance through a multi-faceted approach. We leverage the TRACE framework [22], a baseline for continual learning in LLMs, to compare approaches across eight distinct tasks: C-STANCE [23], FOMC [24], MeetingBank [25], Py150 [26], ScienceQA [27], NumGLUE-cm, NumGLUE-ds [28], and 20Minuten [29]. To further probe the models' knowledge retention and reasoning capabilities, we incorporate the Massive Multitask Language Understanding (MMLU) benchmark [30]. MMLU provides a broad assessment with questions across 57 subjects of varying difficulty. On this dataset, we measure performance using 5-shot accuracy calculated from answer perplexity.

B. Metrics

In the context of the aforementioned benchmark evaluations, let $R_{i,j}$ denote the testing score on the j^{th} task subsequent to training on the i^{th} task. To rigorously assess the model's continual learning capabilities, we employ two metrics: Overall Performance (OP) [22] and Backward Transfer (BWT) [31].

The **OP** serves as a holistic indicator of the model's average proficiency across all learned tasks upon the completion of the training curriculum. It reflects the model's final capacity to handle the entire sequence of tasks and is defined as:

$$OP_t = \frac{1}{t}\sum_{i=1}^{t} R_{t,i},$$

where $R_{t,i}$ represents the performance on the i^{th} task after the model has been trained on the final task t.

Complementing this, **BWT** is utilized to quantify the stability of previously acquired knowledge, measuring the extent to which learning new tasks influences the performance on earlier ones. This metric captures the severity of catastrophic forgetting and is calculated using the following formula:

$$BWT_t = \frac{1}{t-1}\sum_{i=1}^{t-1}\left(R_{t,i} - R_{i,i}\right),$$

where $R_{t,i}$ denotes the score on task i after the entire sequence up to task t has been learned, and $R_{i,i}$ serves as the baseline performance on task i immediately after its initial training.

C. Baselines

We benchmark OWM-LoRA against four parameter-efficient continual learning baselines within the LoRA framework to ensure a fair comparison: LoRA, replay-based MbPA++ [32], and two orthogonality methods, O-LoRA [33] and OGD [34]. Unlike O-LoRA's soft regularizer or OGD's projection against stored gradients, OWM-LoRA employs a single, dynamically updated projection matrix to filters gradients into a provably orthogonal space. The proposed approach combines the low memory footprint of soft regularization with the robustness of hard orthogonal constraints.

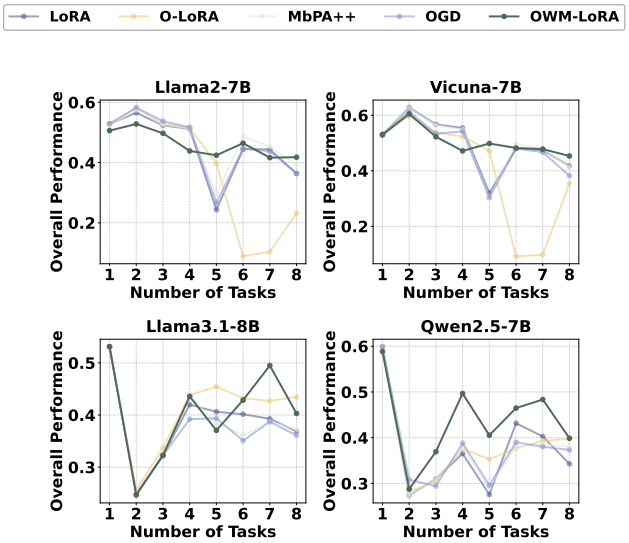
D. Implementation Details

OWM-LoRA is an architecture-agnostic method applicable to any transformer-based model. We evaluated its effectiveness on the TRACE and MMLU [30] benchmark against various

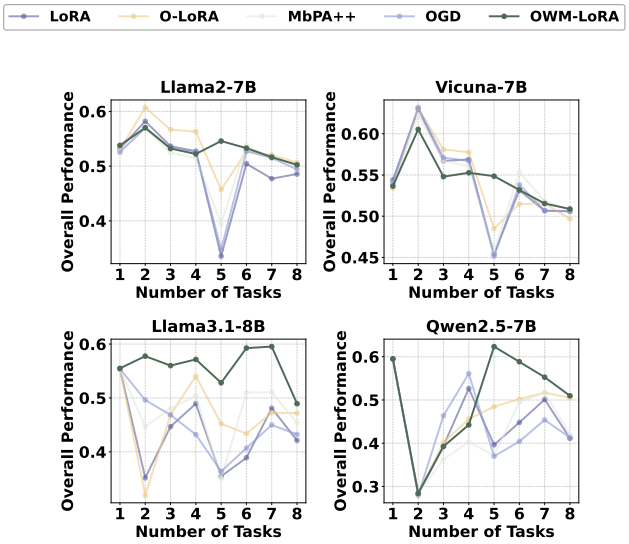
baselines, using a diverse set of models to demonstrate its versatility: the decode-only Llama2-7B [35] and its fine-tuned variant Vicuna-7B [36], the newer Llama3.1-8B [37], and the Mixture-of-Experts (MoE) model Qwen2.5-7B [38]. For robust evaluation, we report the median performance across three independent runs.

Inspired by work [39] showing that optimal LoRA placement is task- and model-dependent, we investigated selective application as an alternative to uniform application. Given that self-attention and FFN layers are the primary components of modern large language models (excluding embeddings), we conducted experiments by applying LoRA to: (1) the query and value (QV) projections (W_q, W_v) within self-attention layers, and (2) the FFN layers (W_{up_proj}, W_{gate_proj}).

V. RESULTS



(a) OP of QV Layers Over Tasks



(b) OP of FFN Layers Over Tasks

Fig. 2: Task-wise OP on QV and FFN Layers

A. OWM-LoRA Improves Performance and Mitigates Forgetting in QV Layers

As shown in Table I, we strictly adhere to the conventional implementation established in the original LoRA framework by integrating adaptation modules solely into the W_q and W_v components within the attention layers. To rigorously evaluate the efficacy of the proposed approach, we conduct a comparative analysis against established baselines, focusing on the OP and BWT metrics across a diverse range of models fine-tuned on the TRACE benchmark. The empirical findings indicate that our OWM-LoRA method consistently achieves a superior overall performance relative to existing baselines across nearly all tested architectures. Such results suggest that the strategy of constraining parameter updates within a subspace orthogonal to those of preceding tasks effectively mitigates interference, thereby enhancing the capacity of large language models for continual learning. Furthermore, the significant improvement observed in the BWT metric highlights the advantage of our method in preserving previously acquired knowledge, demonstrating its robustness against the challenge of catastrophic forgetting.

Figure 2 (Panel 2a) delineates the longitudinal performance dynamics across the sequential task curriculum. In contrast to baselines which exhibit marked volatility characterized by a precipitous decline during the fifth task, OWM-LoRA maintains a consistently stable trajectory. This empirical stability substantiates the hypothesis that orthogonal projection functions as a geometric regularizer for parameter updates. By constraining optimization to subspaces orthogonal to prior representations, the mechanism prevents drastic deviations that would otherwise compromise historical performance, effectively prioritizing the preservation of accumulated knowledge over the unconstrained maximization of peak scores for individual novel tasks.

B. Generalizability of OWM-LoRA on FFN Layers and Superior FFN Adaptation over QV

To evaluate our method's efficacy on Feed-Forward Network (FFN) layers, we applied LoRA adaptation modules to W_{up_proj} and W_{gate_proj}. The results (Table I) demonstrate that OWM-LoRA achieves the highest overall performance (OP) on the TRACE benchmark across nearly all models, confirming its effectiveness. While OWM-LoRA showed strong backward transfer (BWT) compared to most baselines, it did not surpass O-LoRA, highlighting a potential area for future improvement. Consistent with observations from the QV layers, OWM-LoRA also shows smaller performance fluctuations on tasks prone to catastrophic forgetting (Figure 2), underscoring its generalizability.

Notably, our experiments reveal that the choice of layer for adaptation is critical (Table I). Under identical continual learning settings, applying LoRA to the FFN layers yielded superior performance compared to applying it to the QV layers. This finding suggests that fine-tuning FFN weights is a more effective strategy for continual learning, implying that these layers may store richer knowledge representations.

Method	Llama2-7B		Vicuna-7B		Llama3.1-8B		Qwen2.5-7B	
	QV	FFN	QV	FFN	QV	FFN	QV	FFN
LoRA	0.364 / -0.183	0.486 / -0.068	0.418 / -0.140	0.506 / -0.054	0.369 / -0.132	0.421 / -0.109	0.343 / -0.121	0.412 / -0.150
O-LoRA	0.232 / -0.188	**0.507** / **-0.018**	0.355 / -0.105	0.497 / **-0.025**	**0.435** / -0.057	0.472 / **-0.050**	0.398 / **-0.029**	0.505 / -0.029
MbPA++	0.392 / -0.154	0.502 / -0.056	0.413 / -0.143	0.508 / -0.061	0.372 / -0.107	0.454 / -0.108	0.380 / -0.067	0.420 / -0.128
OGD	0.364 / -0.182	0.494 / -0.066	0.382 / -0.169	0.506 / -0.056	0.361 / -0.127	0.432 / -0.149	0.373 / -0.140	0.413 / -0.126
OWM-LoRA	**0.418** / **-0.072**	0.502 / -0.049	**0.447** / **-0.035**	**0.509** / -0.040	0.403 / **-0.056**	**0.489** / -0.111	**0.399** / -0.054	**0.510** / **-0.029**

TABLE I: Comparison of Overall Performance (OP) and Backward Transfer (BWT). Format: **OP / BWT**.

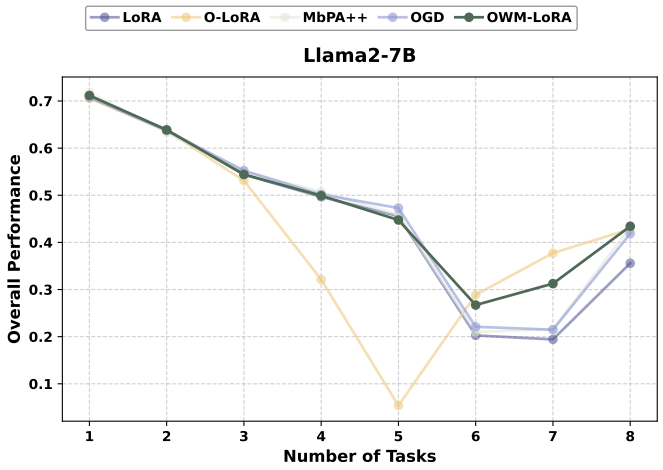


Fig. 3: OP of shuffled tasks on QV layers

C. OWM-LoRA Robustness to Task Order and Similarity

To assess task-order sensitivity, we evaluated Llama2-7B on a randomized TRACE benchmark sequence (Figure 3). Consistent with our main findings (Sections V-A, V-B), OWM-LoRA outperformed all baselines. It uniquely avoided severe forgetting on fourth and fifth tasks and achieved the highest final performance. Since the relative performance gains were invariant to the task order, we confirm OWM-LoRA's efficacy stems from its orthogonal update mechanism rather than task sequencing, highlighting its robustness and applicability.

Method	LoRA	O-LoRA	OGD	MbPA++	Ours
OP	0.6425	0.6179	0.6178	0.6425	**0.6364**

(a) Overall Performance on NumGLUE

Method	LoRA	O-LoRA	OGD	MbPA++	Ours
OP	0.7184	0.7178	0.7170	0.7181	**0.7200**

(b) Overall Performance on MMLU

TABLE III: Overall Performance Comparison of Different Methods

To directly probe the boundary conditions of the orthogonality hypothesis, we conducted an ablation study on task similarity. We experimented on the NumGLUE dataset from the TRACE benchmark, which features high task similarity, using the Qwen2.5-7B model. As shown in Table III(panel IIa), while our OWM-LoRA method exhibits a marginal performance gap compared to baselines such as LoRA and MbPA++, it demonstrates a significant improvement over other orthogonality-based methods like OGD and O-LoRA under identical settings. On the MMLU benchmark for cross-task validation, OWM-LoRA consistently achieves state-of-the-art performance with Qwen2.5-7B (Table III, panel IIb). This result indicates that its orthogonal constraint improves performance by preserving cross-task stability. The underlying mechanism, orthogonal gradient projection, minimizes inter-task interference, thus allowing the model to retain acquired knowledge while adapting to new tasks.

Model	QV Layers	FFN Layers
Llama2-7B	$10^{-1} \rightarrow 10^{-3}$	10^{-1}
Vicuna-7B	10^{-1}	10^{-1}
Llama3.1-8B	10^{-1}	10^{-1}
Qwen2.5-7B	$10^{-1} \rightarrow 10^{-3}$	10^{-1}

TABLE IV: Task-specific α values for OWM-LoRA. The notation $10^{-1} \rightarrow 10^{-3}$ indicates an annealing schedule where early tasks use $\alpha = 10^{-1}$ and later tasks relax to $\alpha = 10^{-3}$.

D. Hyperparameter Sensitivity and Adaptive Scheduling Dynamics

The regularization coefficient α governs the stability–plasticity trade-off through the projection matrix $\mathbf{P}$. Larger α improves retention but reduces adaptability, while smaller α accelerates adaptation at the cost of increased forgetting. We therefore use a fixed α for low-conflict settings and a scheduled α to gradually relax regularization over training. Grid search shows that $\alpha = 10^{-1}$ gives the best balance between Overall Performance and Backward Transfer, particularly in FFN layers. Higher values cause intransigence, whereas values below 10^{-2} worsen forgetting. In QV layers of models such as Llama2-7B, decaying α from 10^{-1} to 10^{-3} further improves retention.

VI. Conclusion

We present OWM-LoRA, a continual learning method for LLMs that combines LoRA with orthogonal gradient projection to reduce catastrophic forgetting without storing past data or gradients. Results on TRACE and MMLU show consistent gains over strong baselines. We also find that FFN-based LoRA adaptation is more effective than QV-based adaptation for continual learning. The strong performance under shuffled task orders and high task similarity further demonstrates the robustness and scalability of OWM-LoRA.

VII. Acknowledgment

The authors used AI-assisted tools for language polishing and editing during manuscript preparation.

References

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al., "Language models are few-shot learners," Advances in neural information processing systems, vol. 33, pp. 1877–1901, 2020.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al., "Gpt-4 technical report," arXiv preprint arXiv:2303.08774, 2023.

[3] Tongtong Wu, Massimo Caccia, Zhuang Li, Yuan Fang Li, Guilin Qi, and Gholamreza Haffari, "Pretrained language model in continual learning: A comparative study," in International Conference on Learning Representations 2022. OpenReview, 2022.

[4] Yun Luo, Zhen Yang, Xuefeng Bai, Fandong Meng, Jie Zhou, and Yue Zhang, "Investigating forgetting in pre-trained representations through continual learning," arXiv preprint arXiv:2305.05968, 2023.

[5] Michael McCloskey and Neal J Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in Psychology of learning and motivation, vol. 24, pp. 109–165. Elsevier, 1989.

[6] iyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu, "A comprehensive survey of continual learning: Theory, method and application," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024.

[7] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al., "Lora: Low-rank adaptation of large language models.," ICLR, vol. 1, no. 2, pp. 3, 2022.

[8] Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu, "Continual learning of context-dependent processing in neural networks," Nature Machine Intelligence, vol. 1, no. 8, pp. 364–372, 2019.

[9] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences, 114(13):3521–3526.657

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

[11] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Greg Wayne. 2018. Experience replay for continual learning. CoRR, abs/1811.11682.

[12] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. 2018. Overcoming catastrophic forgetting with hard attention to the task. In International conference on machine learning, pages 4548–4557. PMLR.

[13] David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. Advances in neural information processing systems, 30.

[14] Arslan Chaudhry, Naeemullah Khan, Puneet Dokania, and Philip Torr. 2020. Continual learning in low-rank orthogonal subspaces. Advances in Neural Information Processing Systems, 33:9900–9911

[15] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In International conference on machine learning, pages 2790–2799. PMLR.

[16] Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. arXiv preprint arXiv:2104.08691.

[17] Tongtong Wu, Linhao Luo, Yuan-Fang Li, Shirui Pan, Thuy-Trang Vu, and Gholamreza Haffari. 2024. Continual learning for large language models: A survey. arXiv preprint arXiv:2402.01364.

[18] Jialin Liu, Jianhua Wu, Jie Liu, and Yutai Duan. 2024. Learning attentional mixture of loras for language model continual learning. arXiv preprint arXiv:2409.19611.

[19] Yuxuan Zhang. 2025. Secura: Sigmoid-enhanced cur decomposition with uninterrupted retention and low-rank adaptation in large language models. arXiv preprint arXiv:2502.18168.

[20] Xiaoyu Kong, Jiancan Wu, An Zhang, Leheng Sheng, Hui Lin, Xiang Wang, and Xiangnan He. 2024. Customizing language models with instance-wise lora for sequential recommendation. arXiv preprint arXiv:2408.10159.

[21] Jingyang Qiao, Zhizhong Zhang, Xin Tan, Yanyun Qu, Wensheng Zhang, Zhi Han, and Yuan Xie. 2024. Gradient projection for continual parameter-efficient tuning. arXiv preprint arXiv:2405.13383.

[22] Xiao Wang, Yuansen Zhang, Tianze Chen, Songyang Gao, Senjie Jin, Xianjun Yang, Zhiheng Xi, Rui Zheng, Yicheng Zou, Tao Gui, et al., "Trace: A comprehensive benchmark for continual learning in large language models," arXiv preprint arXiv:2310.06762, 2023.

[23] Chenye Zhao, Yingjie Li, and Cornelia Caragea, "C-stance: A large dataset for chinese zero-shot stance detection," in Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2023, pp. 13369–13385.

[24] Agam Shah, Suvan Paturi, and Sudheer Chava, "Trillion dollar words: A new financial dataset, task& market analysis," arXiv preprint arXiv:2305.07972, 2023.

[25] Yebowen Hu, Tim Ganter, Hanieh Deilamsalehy, Franck Dernoncourt, Hassan Foroosh, and Fei Liu, "Meetingbank: A benchmark dataset for meeting summarization," arXiv preprint arXiv:2305.17529, 2023.

[26] Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, et al., "Codexglue: A machine learning benchmark dataset for code understanding and generation," arXiv preprint arXiv:2102.04664, 2021.

[27] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan, "Learn to explain: Multimodal reasoning via thought chains for science question answering," Advances in Neural Information Processing Systems, vol. 35, pp. 2507–2521, 2022.

[28] Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan, "Numglue: A suite of fundamental yet challenging mathematical reasoning tasks," arXiv preprint arXiv:2204.05660, 2022.

[29] Annette Rios Gonzales, Nicolas Spring, Tannon Kew, Marek Kostrzewa, Andreas S¨auberli, Mathias M¨uller, and Sarah Ebling, "A new dataset and efficient baselines for document-level text simplification in german," in Proceedings of the Third Workshop on New Frontiers in Summarization, 2021, pp. 152–161.

[30] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Man- tas Mazeika, Dawn Song, and Jacob Steinhardt, "Measuring massive multitask language understanding," arXiv preprint arXiv:2009.03300, 2020.

[31] David Lopez-Paz and Marc'Aurelio Ranzato, "Gradient episodic memory for continual learning," Advances in neural information processing systems, vol. 30, 2017.

[32] Cyprien de Masson D'Autume, Sebastian Ruder, Lingpeng Kong, and Dani Yogatama. 2019. Episodic memory in lifelong language learning. Advances in Neural Information Processing Systems, 32.

[33] Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. 2023a. Orthogonal subspace learning for language model continual learning. arXiv preprint arXiv:2310.14152.

[34] Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. 2020. Orthogonal gradient descent for continual learning. In International conference on artificial intelligence and statistics, pages 3762–3773. PMLR.

[35] 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.

[36] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al., "Judging llm-as-a-judge with mt-bench and chatbot arena," Advances in Neural Information Processing Systems, vol. 36, pp. 46595–46623, 2023.

[37] AI@Meta, "Llama 3 model card," 2024.

[38] Team Qwen, "Qwen2.5: A party of foundation models," September 2024.

[39] Vlad Fomenko, Han Yu, Jongho Lee, Stanley Hsieh, and Weizhu Chen, "A note on lora," arXiv preprint arXiv:2404.05086, 2024.