Soft Orthogonal Low-Rank Adaptation for Knowledge Sharing in Large Language Model Continual Learning

Yitong Wang[♣*], **Xue Han**[♣*], **Wenchun Gao**[♣], **Qian Hu**[♣], **Jiahui Wang**[♣], **Ziqing Wang**[♠], **Lijun Mei**[♣], **Junlan Feng**[♣†]
[♣]JIUTIAN Research, China Mobile, Beijing, China
[♠]Guanghua School of Management Peking University, Beijing, China
{wangyitong,hanxue,gaowenchun,huqian,wangjiahui,meilijun,fengjunlan}it@chinamobile.com
2400015831@stu.pku.edu.cn

Abstract

When large language models are used in real-world scenarios, continual learning (CL) becomes a non-trivial problem. In particular, continual learning with modern LLMs is challenged both by the substantial computational costs induced by their massive parameter scale, and by the limitations of current CL methods, which are mainly designed to mitigate catastrophic forgetting while neglecting knowledge sharing across tasks. We further observe that models with stronger performance exhibit stronger inter-task connections. In light of the above challenges and findings, we propose Attribution Scores-based Soft Orthogonality Low-Rank Adaptation (ASO-LoRA), an effective and efficient framework that simultaneously facilitates knowledge transfer while mitigating catastrophic forgetting. Specifically, ASO-LoRA initially assigns task-specific parameter subspaces for new tasks utilizing multi-LoRA modules, enabling for efficient training and inference without relying on task labels. Then, ASO-LoRA leverages attribution scores to evaluate task similarity and employs soft orthogonality between task-specific subspaces, guiding gradient updates in directions that promote parameter isolation, achieving a balance between knowledge transfer and preservation. Experiments are carried out on both the T5-large and the LLaMA2-7B, showing ASO-LoRA's superior performance and suitability as a plug-in CL solution for general Transformer-based LLMs. Code is available at https://github.com/736619821/ASO-LORA.

1 Introduction

Continuous learning (CL) is essential for applying language models in real-world scenarios, as presenting training data sequentially from varied distributions of different tasks can result in catastrophic forgetting (Wang et al., 2024a). Despite the remarkable performance of recently published large language models (LLMs) such as GPT-4 (Achiam et al., 2023a), Llama (Touvron et al., 2023), and DeepSeek (Liu et al., 2024a) across various tasks (Han et al., 2023; Bai et al., 2025), CL still remains a significant challenge, as LLMs are not suited for frequent retraining due to the notable training costs related to their large scale (Wu et al., 2024; Han et al., 2025).

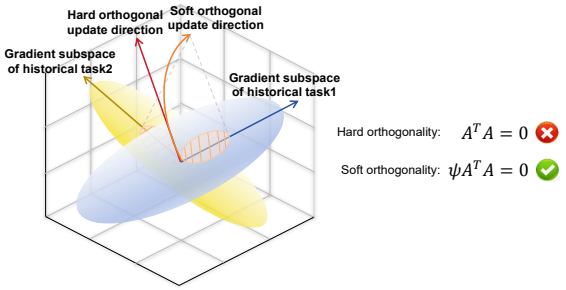


Figure 1: The key innovation lies in ASO-LoRA's dual capability to mitigate catastrophic forgetting of historical tasks while simultaneously facilitating inter-task knowledge transfer. ASO-LoRA updates the parameters subspace of new task through a soft orthogonal direction (orange curved arrow) with the historical tasks, instead of updating through a hard orthogonal direction (red straight arrow), to constrain the gradient updates of the current task to be softly orthogonal to the gradient subspace of the past tasks.

In contrast to its employment in smaller models, efficient CL is of great importance for LLMs since implementing CL with billions of parameters incurs large computing costs. To realize efficient CL, recent research has adopted the Parameter-Efficient Tuning (PET) frameworks for LLMs. Razdaibiedina et al. (2023) learns a new soft prompt for each task and concatenates it with previously learnt prompts, while freezing the base model. O-LoRA (Wang et al., 2023) incrementally learns new tasks in an orthogonal subspace while fixing the LoRA parameters learned from past tasks to mini-

[*]These authors contributed to the work equally.
[†] Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 18472–18487

mize catastrophic forgetting. However, these methods tend to address only catastrophic forgetting, neglecting the possibility of transferring knowledge between tasks. SAPT (Zhao et al., 2024b) aligns the learning and selection of LoRA parameters via the shared attentive learning and selection module, addressing catastrophic forgetting and knowledge transfer simultaneously. Despite being PET-agnostic, SAPT's additional modules increase architectural complexity.

Recent studies (Wang et al., 2023; Saha et al., 2021) further suggest that optimizing along directions orthogonal to historical tasks' gradient subspaces can meet the purpose of mitigating catastrophic forgetting, by minimizing interference with their loss functions. Meanwhile, Low-Rank Adaptation (LoRA) (Hu et al., 2022a) has emerged as an efficient PET method, showing that fine-tuning task-specific low-rank subspaces can achieve competitive performance. Beyond these insights, we further visualize the similarity between the final output distributions of two models with different performance levels in the continual learning setting in Section 3.1. We observe that models achieving higher average task performance exhibit greater similarity across tasks, indicating stronger inter-task connections and more effective knowledge sharing across tasks.

Inspired by these, we propose Attribution Scores-based Soft Orthogonality Low-Rank Adaptation (ASO-LoRA), a simple yet efficient methodology for continual learning in large language models, addressing both catastrophic forgetting and promoting knowledge transfer. We begin by assigning task-specific parameter subspaces for new tasks utilizing multi-LoRA modules, leaving the LLMs' parameters frozen. Multi-LoRA also enables inference without relying on task labels, allowing for generalization to previously unknown tasks. Then we improve the original method, which only takes gradient steps in the orthogonal direction, by hypothesizing that different tasks may share similar knowledge that could be transferred to enhance their task capabilities. As shown in Figure 1, we use attribution scores (Dai et al., 2021) to evaluate task similarity and propose taking gradient steps in a soft orthogonal direction between task-specific subspaces, achieving a balance between knowledge transfer and preservation.

We conduct experiments using the encoder-decoder-based T5-large and decoder-only-based LLaMA2-7B models, demonstrating ASO-LoRA as a plug-in CL solution for general Transformer-based LLMs. Experimental results on CL benchmarks show that ASO-LoRA outperforms other strong baselines.

2 Related works

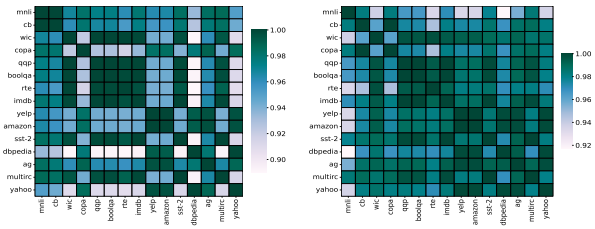
Continual learning in LLMs refers to the paradigm where LLMs sequentially acquire new knowledge from non-stationary data distributions while preserving learned capabilities (Wang et al., 2024a; Wu et al., 2024), and can be categorized into these types: Replay-based approach (Smith et al., 2024; Petit et al., 2023), Regularization-based approach (Zhao et al., 2024c), Optimization-based approach (Wang et al., 2022a), and Architecture-based approach (Han et al., 2025).

Parameter-Efficient Tuning (PET) adapts models by optimizing performance through updates to a minimal set of parameters (without direct modification of original parameters), employing Adapters (Wang et al., 2022b), soft prompts (Liu et al., 2024c), or low-rank adaptations (LoRA) (Liu et al., 2024b), to significantly reduce computational costs (Coleman et al., 2025).

Due to space constraints, please refer to the Appendix A for more detailed related works.

3 Methodology

3.1 Observations



(a) Inferior performance model (b) Better performance model

Figure 2: Inferior vs. Better performance model on the final output distribution similarity across different tasks in the CL scenarios on T5-Large structure.

The motivation behind the ASO-LoRA method is based on the following discovery: we visualize the similarity between the final output distributions of two models with different performance levels, across various tasks in a continual learning setting.

As shown in the heatmap in Figure 2, the model with higher average task performance exhibits greater similarity between tasks, thereby more effectively strengthening inter-task connections. This

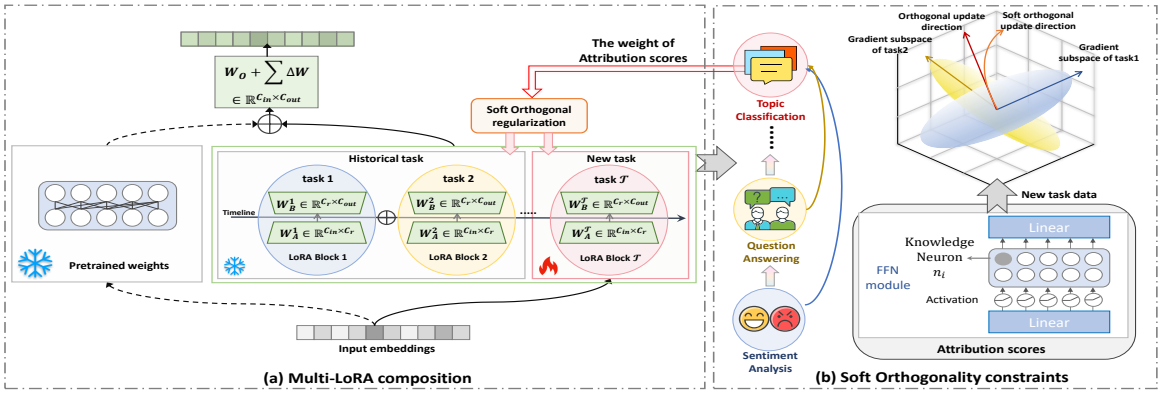


Figure 3: The overall framework of ASO-LoRA, employing multi-LoRA composition with a soft orthogonality constraint. a) We progressively train a new LoRA block for each incoming new task while freezing all historical LoRA blocks, and sequentially stack it upon existing LoRA blocks for the inference stage. b) The parameter updates for the new task's subspace adhere to a soft orthogonality constraint as a regularization term: we derive a soft orthogonality coefficient by quantifying the similarity between the attribution scores of the current and historical tasks, instead of enforcing the subspaces to be strictly orthogonal to mitigate catastrophic forgetting. This soft orthogonality allows for partial overlap in the tasks' subspaces, facilitating the cross-task knowledge transfer.

difference, driven by the distinct interaction patterns between LoRA blocks, necessitates the development of a new method, which could facilitate more effective cross-task knowledge transfer, thereby contributing to improved performance in continual learning scenarios.

Leveraging from the above findings, we propose ASO-LoRA, with Figure 3 illustrating an overview of the whole framework. ASO-LoRA utilizes a multi-LoRA composition mechanism with soft orthogonality constraints to enable continual learning of incremental tasks. Next, we will present a comprehensive exposition of our proposed method.

3.2 Preliminary

Task Definition: Continual learning (CL) refers to a learning paradigm in which a model M, parameterized by θ, sequentially acquires knowledge from a series of tasks' data $\{D_1, D_2, ..., D_{\mathcal{T}}\}$ over $\mathcal{T}$ learning stages. Each task t consists of a set of input and target output pairs $\{(x_i^t, y_i^t)\}_{i=1}^n$. The objective of CL is to optimize the model's parameters θ such that M averagely performs well on all learned tasks:

$$max_\theta \sum_{t=1}^{\mathcal{T}} \sum_{\{(x_i^t, y_i^t)\}_{i=1}^n \in D_t} logp_M(y_i^t | x_i^t) \quad (1)$$

Continual learning using Multi-LoRA composition. We adopt LoRA, an efficient PET method to fine-tune a specific low-rank subspace of trainable parameters for new tasks, while keeping the pre-trained parameters of the LLMs frozen. LoRA does not rely on task IDs during inference, therefore keeping LLMs' generalization ability on unseen tasks. Specifically, LoRA freezes the LLMs' pre-trained weight matrix $W_{base} \in \mathbb{R}^{d \times k}$ without receiving gradient updates to preserve the acquired knowledge. Trainable low-rank decomposition matrices $A_t \in \mathbb{R}^{d \times r}$ and $B_t \in \mathbb{R}^{r \times k}$ are introduced for each new task t learning in parallel pathways, with $r \ll min(d, k)$.

To enable continual task adaptation for pre-trained LLMs, we employ the multi-LoRA mechanism, in which a separate LoRA block $LoRA_t = \{A_t, B_t\}$ is trained for each downstream task t. The base LLM attached with $LoRA_t$ is defined as M_{LoRA_t}. The updated weight matrix W_t of M_{LoRA_t} for specific task t is as follows:

$$W_t = W_{base} + \Delta W_t = W_{base} + A_t B_t \quad (2)$$

As the number of continually learned tasks increases, we can integrate the updated LoRA parameters into the base LLM parameters during inference, as shown in Eq.(3).

$$W_{merge_{\mathcal{T}}} = W_{base} + \sum_{t=1}^{\mathcal{T}} A_t B_t \quad (3)$$

3.3 Enhancing Multi-LoRA with Attribution scores based Soft Orthogonality

The vanilla CL with multi-LoRA, as described in Eq.(3), fails to account for the interactions between LoRA blocks, leading to instability in the fusion process and catastrophic forgetting. To address

the issues raised above, we propose to enhance LoRA with Attribution Scores based Soft Orthogonality (ASO-LoRA), taking into account both catastrophic forgetting and knowledge transfer between tasks while integrating LoRA blocks. In particular, ASO-LoRA uses orthogonal regularization among the subspaces of multi-LoRA blocks to prevent catastrophic forgetting following previous studies (Wang et al., 2023). It further introduces attribution scores to softly adjust the orthogonal regularization, thereby improving positive inter-task knowledge transfer.

Formally, ASO-LoRA starts by learning new tasks in a direction orthogonal to the LoRA subspaces of historical tasks while freezing historical LoRA parameters. For each task t, $A_t = [a_t^1, a_t^2, ..., a_t^r]$ of $LoRA_t = \{A_t, B_t\}$ is approximated as the core of the task-related subspace, where each vector in $B_t = [b_t^1, b_t^2, ..., b_t^r]$ represents the linear weighting coefficients of the column vectors in A_t. Achieving orthogonality between the subspaces of the new task $\mathcal{T}$ and each historical task $t \in \{1, ..., \mathcal{T}-1\}$ can be expressed as:

$$A_t^T A_{\mathcal{T}} = 0 \tag{4}$$

The hard orthogonality defined in Eq (4) simply considers the difference between tasks while ignoring the sharing of knowledge, resulting in inefficiency. We hypothesize that the relationship between LoRA blocks is not strictly orthogonal and that interactions between LoRA blocks may yield knowledge transfer with beneficial effects. To capture the shared knowledge, we further introduce the concept of a dynamic soft orthogonality coefficient $\psi_{t,\mathcal{T}}^{soft}$. $\psi_{t,\mathcal{T}}^{soft}$ decreases as more knowledge is shared between two LoRA blocks, indicating a stronger enhanced parametric correlation. In contrast, the coefficient increases as the LoRA blocks become more orthogonal, indicating a decrease in the relevance of the knowledge of tasks represented by the LoRA block. With the soft coefficient factor $\psi_{t,\mathcal{T}}^{soft}$, the hard orthogonality defined in Eq.(4) can be reformulated as the attribution score-based soft orthogonality $ASO_{t,\mathcal{T}}$:

$$ASO_{t,\mathcal{T}} = \psi_{t,\mathcal{T}}^{soft} A_t^T A_{\mathcal{T}} = 0 \tag{5}$$

The soft coefficient factor $\psi_{t,\mathcal{T}}^{soft}$ is calculated by the weight of attribution scores $w_{t,\mathcal{T}}^{attr}$ between the new task $\mathcal{T}$ and historical task $t (t < \mathcal{T})$ as below:

$$\psi_{t,\mathcal{T}}^{soft} = 1 - w_{t,\mathcal{T}}^{attr} \tag{6}$$

Finally, the CL training objective for the new task $\mathcal{T}$ is defined as:

$$\begin{aligned} \mathcal{L} &= \sum_{(x,y)\in D_{\mathcal{T}}} log p_{M_{LoRA_{\mathcal{T}}}}(y|x) + \lambda \sum_{t=1}^{\mathcal{T}-1} \mathcal{L}_{ASO}(A_t, A_{\mathcal{T}}) \\ \mathcal{L}_{ASO}(A_t, A_{\mathcal{T}}) &= \sum_{j,k} ||ASO_{t,\mathcal{T}}[j,k]||^2 \end{aligned} \tag{7}$$

Where $ASO_{t,\mathcal{T}}$ is the attribution score-based soft orthogonality defined in Eq.(5). $ASO_{t,\mathcal{T}}[j,k]$ denotes the element at the $j-th$ row and $k-th$ column of $ASO_{t,\mathcal{T}}$, and λ is the weight of the soft orthogonality loss.

Next, we present the details about how to derive the weight of attribution scores $w_{t,\mathcal{T}}^{attr}$.

3.4 The weight of Attribution scores

Before introducing the weight $w_{t,\mathcal{T}}^{attr}$ of the attribution score, we first introduce the concept of Knowledge Neuron.

Knowledge Neurons (KNs) are neurons in the Transformer's Feed-forward Network (FFN) memories that store factual knowledge, discovered by recent studies (Dai et al., 2021). FFN is responsible for applying a non-linear transformation to the hidden state H, given by the multi-head attention (MHA) module of Transformer:

$$FFN(H) = GELU(HW^{l^1})W^{l^2} = NeuronsW^{l^2} \tag{8}$$

W^{l^1} and W^{l^2} are weight parameter matrices of the FFN layers. For simplicity, we omit the scaling factor in MHA and the bias terms in FFN.

Attribution scores evaluate the contribution of each knowledge neuron in the LLM to the knowledge expression of data (Dai et al., 2021) based on the integrated gradients. We utilize attribution scores to assess how similar the model with the new-task LoRA is to models with historical-task LoRA blocks, allowing us to establish the proper level of orthogonal constraints between the LoRA blocks.

Given an input data $x_{\mathcal{T}}$ of new task $\mathcal{T}$, we first define the output $P_{M_{LoRA_t}}^{\mathcal{T}}(\hat{n}_t^i)$ as the probability of the correct answer predicted by M_{LoRA_t} (base model attached to a LoRA of each specific task t):

$$P_{M_{LoRA_t}}^{\mathcal{T}}(\hat{n}_t^i) = p_{M_{LoRA_t}}(y^*|x_{\mathcal{T}}, n_t^i = \hat{n}_t^i) \tag{9}$$

where y^* denotes the correct answer; n_t^i denotes the knowledge neuron of M_{LoRA_t} to be calculated, and $\hat{n}_t^i$ is a given constant that n_t^i is assigned to.

To determine the attribution score $Attr(n_t^i)$, we gradually alter n_t^i from 0 to its original value $\bar{n}_t^i$ calculated by the M_{LoRA_t}, while simultaneously integrating the gradients:

$$Attr(n_t^i) = \bar{n}_t^i \int_{\sigma=0}^{1} \frac{\delta P^{\mathcal{T}}_{M_{LoRA_t}}(\sigma \bar{n}_t^i)}{\delta n_t^i} d\sigma \quad (10)$$

$\frac{P^{\mathcal{T}}_{M_{LoRA_t}}(\sigma \bar{n}_t^i)}{\delta n_t^i}$ calculates the gradient of the model output with regard to n_t^i. As σ varies from 0 to 1, $Attr(n_t^i)$ accumulates the cumulative impact of modifying n_t^i on the change of output probability.

Since it is difficult to directly calculate continuous integrals, the Riemann approximation (Roe, 1981) method is used to estimate the value of integrated gradients, and we set the number of approximation steps $m = 20$:

$$Attr(n_t^i) = \frac{\bar{n}_t^i}{m} \sum_{k=1}^{m} \frac{\delta P^{\mathcal{T}}_{M_{LoRA_t}}(\frac{k}{m} \bar{n}_t^i)}{\delta n_t^i} \quad (11)$$

When a knowledge neuron n_t^i significantly impacts the expression of knowledge, the resulting gradient becomes prominent, leading to high integration values $Attr(n_t^i)$.

For M_{LoRA_t}, we assemble the attribution scores corresponding to all knowledge neurons $[n_t^1, ..., n_t^N]$, denoting them as a vector:

$$Attr\_vec_{M_{LoRA_t}} = [Attr(n_t^1), ..., Attr(n_t^N)] \quad (12)$$

Where N represents the number of M_{LoRA_t}'s KNs.

Using the aforementioned, we can calculate the weight of attribution scores $w_{t,\mathcal{T}}^{attr}$ as shown below. We quantify the influence of each historical task $t (t < \mathcal{T})$ on the current task $\mathcal{T}$ by computing the similarity between their attribution vectors for the current task $\mathcal{T}$'s data:

$$w_{t,\mathcal{T}}^{attr} = Sim(Attr\_vec_{M_{LoRA_t}}, Attr\_vec_{M_{LoRA_{\mathcal{T}}}}) \quad (13)$$

$w_{t,\mathcal{T}}^{attr}$ evaluates the transferability of knowledge from previous tasks $\{1, 2, ..., \mathcal{T}-1\}$ to new task $\mathcal{T}$. A higher $w_{t,\mathcal{T}}^{attr}$ indicates greater knowledge sharing between tasks t and $\mathcal{T}$, suggesting that $LoRA_t$ is more likely to positively influence $LoRA_{\mathcal{T}}$ block.

4 Experiments

4.1 Experimental setups

4.1.1 Datasets

CL benchmarks: Following Wang et al. (2023)'s work, we conduct comprehensive evaluations of ASO-LoRA against state-of-the-art baselines on four standard continual learning benchmarks for language models: AG News, Amazon Reviews, DBpedia, and Yahoo Answers (Zhang et al., 2015).

Longer benchmarks: To further validate our approach, we conduct extensive experiments on longer task sequences, incorporating 15 common tasks used for language models (Razdaibiedina et al., 2023), including the AG News, Amazon Reviews, DBpedia, Yahoo Answers, and Yelp reviews from standard CL benchmarks; MNLI, QQP, RTE, SST2 from GLUE benchmark (Wang et al., 2018); WiC, CB, COPA, MultiRC, BoolQ from SuperGLUE (Wang et al., 2019), and the IMDB movie reviews (Maas et al., 2011).

We explore 6 different orders of the benchmarks to validate the methods' efficacy across diverse continual learning scenarios.

4.1.2 Metrics

Following Wang et al. (2024b), we use the Average Accuracy (AA) to evaluate the performance of ASO-LoRA in the CL scenario, which is the average accuracy of all tasks after the model finishes training on the latest task $\mathcal{T}$:

$$AA = \frac{1}{\mathcal{T}} \sum_{i=1}^{\mathcal{T}} a_{\mathcal{T},i} \quad (14)$$

We also employ forgetting measure (FM) and forward transfer (FWT) as evaluation metrics to comprehensively evaluate our approach.

4.1.3 Baselines

Based on Wang et al. (2023)'s work, we evaluate our method against strong competitive baselines: (1) **Non-CL baselines**: SeqFT(de Masson D'Autume et al., 2019) and PerTaskFT. PerTaskFT is considered as the upper bound. (2) **LoRA-based**: SeqLoRA, IncLoRA, MoELoRA(Luo et al., 2024), MoCL(Wang et al., 2024c) and O-LoRA(Wang et al., 2023). (3) **Traditional CL baselines**: Replay(Chaudhry et al., 2019), EWC(Kirkpatrick et al., 2017), LwF(Li and Hoiem, 2017), L2P(Wang et al., 2022c), and LFPT5(Qin and Joty, 2021). See Appendix B.2 for details of these baselines.

	CL benchmarks				Longer benchmarks			
	Order-1	**Order-2**	**Order-3**	**Avg.**	**Order-4**	**Order-5**	**Order-6**	**Avg.**
SeqFT	18.9	24.9	41.7	28.5	7.4	7.4	7.5	7.4
PerTaskFT	72.5	72.5	72.5	72.5	78.0	78.0	78.0	78.0
Replay	55.2	56.9	61.3	57.8	55.0	54.6	53.1	54.2
EWC	48.7	47.7	54.5	50.3	45.3	44.5	45.6	45.1
LwF	54.4	53.1	49.6	52.3	50.1	43.1	47.4	46.9
L2P	60.3	61.7	61.1	60.7	57.5	53.8	56.9	56.1
LFPT5	67.6	72.6	77.9	72.7	70.4	68.2	69.1	69.2
SeqLoRA	62.6	61.5	69.3	64.5	56.7	52.7	15.9	41.8
IncLoRA	69.5	64.9	70.9	68.4	59.0	62.6	62.3	61.3
MoCL	75.6	75.4	76.7	75.9	-	-	-	-
MoELoRA	52.8	49.6	59.8	54.1	36.3	31.4	15.1	27.6
O-LoRA	75.6	**78.1**	72.1	75.3	71.6	**69.3**	75.8	72.2
ASO-LoRA	**76.2**	77.5	**77.3**	**77.0**	**73.7**	67.2	**77.4**	**72.8**

Table 1: The main averaged accuracy (AA) results on two series of benchmarks with the T5-large model (T5-710M), after training on the last task. The results of CL baselines are referred from Wang et al. (2023)'s work, while the results of MoCL and MoELoRA are from Du et al. (2024).

4.1.4 Implementation Details

ASO-LoRA employs the generalization-friendly instruction tuning as the training paradigm, capturing the underlying commonalities of tasks. We implement ASO-LoRA on two representative Transformer architectures: the encoder-decoder model T5-large (710M) (Raffel et al., 2020) and the decoder-only model LLaMA2-7B (Touvron et al., 2023), highlighting ASO-LoRA's applicability as a plug-in continual learning solution for general Transformer-based language models. The parameters of a single LoRA module account for only 0.4% of the total model parameters. On standard CL benchmarks, LoRA constitutes merely 1.6% of the parameters. The similarity in Eq.(13) employs Spearman's rank correlation coefficient. We train the models with one epoch, using the AdamW (Loshchilov and Hutter, 2017) optimizer in a batch size of 64 with learning rate 1×10^{-3} for each experiment. The weight decay is 0 while the dropout rate is set as 0.1. The weights of the soft orthogonality loss follow Wang et al. (2023)'s work. Results are reported as the average of 3 runs. Our setup consists of a four-core CPU and eight NVIDIA Tesla A100 GPUs.

For more details on task orders, task details, metrics, and baselines, refer to the Appendix B.

4.2 Main results and analysis

Table 1 displays a comprehensive performance comparison between ASO-LoRA and baselines on both CL benchmarks and extended longer benchmarks. We evaluate the effectiveness of ASO-LoRA for CL scenarios from three perspectives:

Performance on CL benchmarks: As evidenced by the results, ASO-LoRA consistently outperforms all baselines across different task orders. Notably, ASO-LoRA achieves significant improvements of 4.5% and 1.7% compared to PerTaskFT and O-LoRA, respectively. These demonstrate that ASO-LoRA effectively mitigates catastrophic forgetting in continual learning scenarios while successfully leveraging previously acquired knowledge. Notably, unlike conventional CL baselines, ASO-LoRA requires neither full-parameter training nor historical data storage, thereby achieving significant computational efficiency while preserving knowledge from the pertaining stage.

Performance on longer general benchmarks: Following Wang et al. (2023), we assess ASO-LoRA on a more challenging scenario, involving sequential training across 15 extended tasks. As illustrated in Table 1, ASO-LoRA achieves superior performance compared to almost all baselines in addressing longer continual learning problems, demonstrating ASO-LoRA's robust adaptability to more complex scenarios. However, Per-

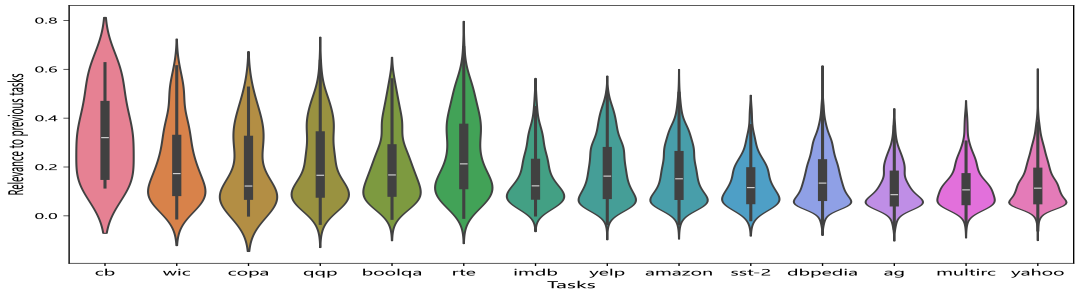


Figure 4: Longer benchmarks' Violin Plot of w^{attr}, displaying the distribution of similarity coefficient w^{attr} across different tasks during the training stages. Central white dot represents the median or mean of the data, while a wider section indicates higher data density (more points near that value).

TaskFT maintains higher performance, indicating that sustained knowledge preservation in long task sequences remains an open challenge.

Table 2: The main AA results on CL benchmarks with the Llama-7B model, after training on the last task.

Model	CL benchmarks			
	Order-1	Order-2	Order-3	Avg.
SeqLoRA	77.2	76.6	78.0	77.2
IncLoRA	32.7	49.1	35.5	39.1
O-LoRA	73.7	75.0	76.5	75.1
ASO-LoRA	**75.0**	74.8	**78.1**	**76.0**

Results on other Transformer-based structures: To further validate the generalizability of ASO-LoRA across Transformer-based architectures, we extend our experiments on the decoder-only LLaMA-7B model using standard CL benchmarks. As evidenced in Table 2, ASO-LoRA achieves a leading performance, with an average improvement of 0.9% over O-LoRA. These empirical evidences confirm ASO-LoRA's plug-and-play adaptability, suggesting its broad applicability across diverse Transformer-based models for complex continual learning scenarios. We include the results of the LLaMA architecture on longer benchmarks in the Appendix C.1.

4.3 The impact of Soft Orthogonality

To facilitate a more intuitive analysis of Soft Orthogonality's effectiveness, we further display the comparative task-specific results after complete model training across three distinct task orderings (Order1-Order3) on both T5 and Llama structures, as listed in Table 3 and 8.

Regardless of task ordering, ASO-LoRA demonstrates superior performance on most tasks compared to O-LoRA and the fine-tuning-only upper bound. Although ASO-LoRA exhibits a lower AA score than O-LoRA on Order2, it achieves greater performance improvements across more individual tasks. Preliminary analysis suggests that potential negative backward transfer from later tasks to the initial task may lead to performance degradation. For Order3, ASO-LoRA achieves a significant 21.8% improvement over O-LoRA on task Amazon. These evidences still prove that the proposed Soft Orthogonality mechanism extends the interaction subspaces between LoRA blocks within PET frameworks, relaxing hard orthogonal constraints while preserving low-rank adaptation benefits. Refer to Appendix C.2 for the results and analysis of Llama structure.

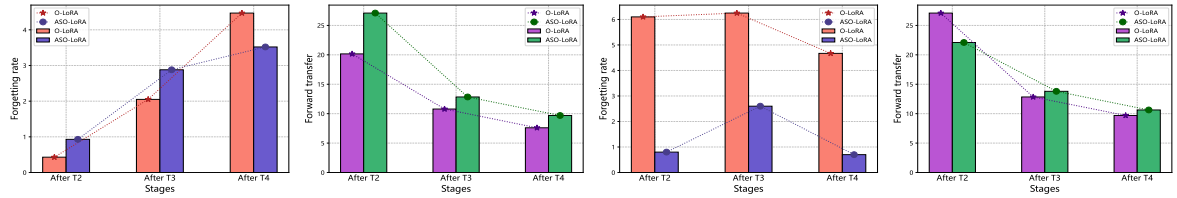
Table 3: Results on individual tasks after completing training on the final task with T5-large across Order 1&2&3. The left-to-right ordering of benchmarks corresponds to the task training order.

Model	CL benchmarks per-task results				
Order1	Sequences	Dbpedia	Amazon	Yahoo	Ag
	PerTaskFT	97.6	34.7	70.0	87.5
	IncLoRA	81.3	38.6	68.0	89.9
	O-LoRA	89.6	56.4	67.5	88.8
	ASO-LoRA	88.1	**57.9**	**71.1**	87.5
Order2	Sequences	Dbpedia	Amazon	Ag	Yahoo
	PerTaskFT	97.6	34.7	87.5	70.0
	IncLoRA	72.0	44.2	70.1	73.5
	O-LoRA	96.8	56.4	87.7	71.8
	ASO-LoRA	91.3	**58.5**	**87.9**	**72.1**
Order3	Sequences	Yahoo	Amazon	Ag	Dbpedia
	PerTaskFT	70.0	34.7	87.5	97.6
	IncLoRA	65.1	44.9	75.0	98.4
	O-LoRA	69.7	33.3	86.6	98.7
	ASO-LoRA	**70.3**	**55.1**	85.2	**98.9**

4.4 The relationships among tasks on Longer benchmarks

We employ the violin plot to visualize the similarity coefficient w^{attr} between the new task and historical tasks, as proposed in Eq.(13), analyzing their inter-task correlations and mutual influences.

Figure 4 presents the cross-task correlation co-



(a) FM_T5 (b) FWT_T5 (c) FM_Llama (d) FWT_Llama

Figure 5: ASO-LoRA vs. O-LoRA on standard CL benchmarks using metrics FM and FWT. Lower FM values indicate stronger resistance to catastrophic forgetting and better knowledge retention capabilities, while higher FWT values demonstrate more effective utilization of prior knowledge and superior knowledge transfer.

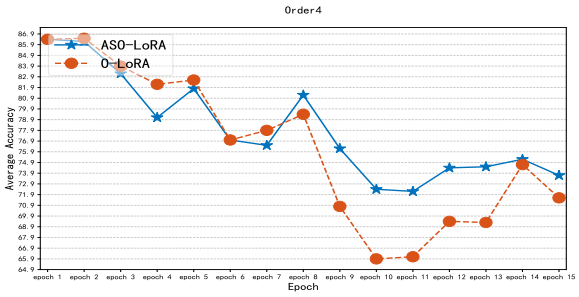
efficients between new and historical tasks under Order4. Task Cb exhibits a strong correlation with the preceding task MNLI. Since both tasks belong to the Natural Language Inference (NLI) category within the GLUE benchmark, this finding aligns with our hypothesis. Likewise, the high similarity between the tasks of RTE and BoolQA can be attributed to their common source from Wikipedia.

4.5 The effect on catastrophic forgetting and knowledge transfer

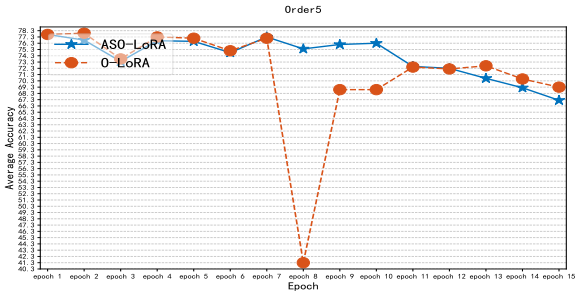
To comprehensively evaluate Soft Orthogonality's efficacy in mitigating catastrophic forgetting and facilitating knowledge transfer, we introduce additional metrics: FM and FWT. These metrics enable systematic comparison with the mere orthogonal-constrained strong baseline O-LoRA. Figure 5 reveals that ASO-LoRA obviously achieves superior knowledge transfer over O-LoRA by more effectively leveraging prior task knowledge on all structures. As the tasks increase, ASO-LoRA exhibits progressively reduced catastrophic forgetting, ultimately outperforming O-LoRA in long-term knowledge preservation on T5 structure. These substantiate our hypothesis that soft orthogonality offers a principled solution for balancing knowledge preservation and transfer in CL.

4.6 The trend of Performance in the Continual learning stages

Longer benchmarks: Figure 6 further dipicts the Average Accuracy trajectories of ASO-LoRA and O-LoRA during CL stages on longer benchmarks. ASO-LoRA demonstrates greater overall stability and outperforms O-LoRA during longer continual learning phases. On Order4, although ASO-LoRA initially lags behind O-LoRA, its ability to mitigate catastrophic forgetting becomes increasingly evident as the number of tasks grows, eventually surpassing O-LoRA. On Order5, while ASO-LoRA does not ultimately outperform O-LoRA, it exhibits a more balanced performance and avoids the sharp performance drop observed in O-LoRA at epoch 8.



(a) Order4



(b) Order5

Figure 6: ASO-LoRA vs. O-LoRA on AA during the CL stage on CL longer benchmarks, as new tasks arrive continuously. Each arrival of a new task corresponds to one epoch of training. The blue line represents the trend of ASO-LoRA'AA as tasks increment, while the red line corresponds to that of O-LoRA

5 Conclusion

In this work, we present ASO-LoRA, an innovative continual learning framework that incorporates Attribution Score-based Soft Orthogonality for parameter-efficient adaptation. Experimental results demonstrate that ASO-LoRA outperforms strong baselines, effectively mitigating catastrophic forgetting while facilitating robust knowledge transfer across sequential tasks.

Limitations

While ASO-LoRA has demonstrated strong capabilities in continual learning scenarios through empirical evaluation, several limitations warrant discussion: 1) The observed performance variation across different Transformer architectures, raising an open question regarding whether encoder-decoder frameworks inherently facilitate better knowledge storage. These phenomena suggest underlying mechanistic differences, requiring further investigation. 2) The potential negative impacts of task overlap need to be further explored. 3) The performance degradation observed in longer benchmarks remains a significant challenge for scaling to more complex real-world applications, such as hundreds of tasks.

We aim to address these limitations in future work to further enhance our method's performance in continual learning scenarios.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023a. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023b. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ruiqiao Bai, Xue Han, Shuo Lei, Junlan Feng, Yanyan Luo, and Chao Deng. 2025. Self-attention-based graph-of-thought for math problem solving. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 6112–6125.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. 2019. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*.

Eric Nuertey Coleman, Luigi Quarantiello, Ziyue Liu, Qinwen Yang, Samrat Mukherjee, Julio Hurtado, and Vincenzo Lomonaco. 2025. Parameter-efficient continual fine-tuning: A survey. *arXiv preprint arXiv:2504.13822*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*.

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.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

Wenyu Du, Shuang Cheng, Tongxu Luo, Zihan Qiu, Zeyu Huang, Ka Chun Cheung, Reynold Cheng, and Jie Fu. 2024. Unlocking continual learning abilities in language models. *arXiv preprint arXiv:2406.17245*.

Xue Han, Yi-Tong Wang, Jun-Lan Feng, Chao Deng, Zhan-Heng Chen, Yu-An Huang, Hui Su, Lun Hu, and Peng-Wei Hu. 2023. A survey of transformer-based multimodal pre-trained modals. *Neurocomputing*, 515:89–106.

Xue Han, Yitong Wang, Junlan Feng, Qian Hu, Chao Deng, and 1 others. 2025. Loire: Lifelong learning on incremental data via pre-trained language model growth efficiently. In *The Thirteenth International Conference on Learning Representations*.

Shwai He, Run-Ze Fan, Liang Ding, Li Shen, Tianyi Zhou, and Dacheng Tao. 2023. Mera: Merging pre-trained adapters for few-shot learning. *arXiv preprint arXiv:2308.15982*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022a. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022b. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

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

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

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024b. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2024c. Gpt understands, too. *AI Open*, 5:208–215.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Tongxu Luo, Jiahe Lei, Fangyu Lei, Weihao Liu, Shizhu He, Jun Zhao, and Kang Liu. 2024. Moelora: Contrastive learning guided mixture of experts on parameter-efficient fine-tuning for large language models. *arXiv preprint arXiv:2402.12851*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Grégoire Petit, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. 2023. Fetril: Feature translation for exemplar-free class-incremental learning. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3911–3920.

Chengwei Qin and Shafiq Joty. 2021. Lfpt5: A unified framework for lifelong few-shot language learning based on prompt tuning of t5. *arXiv preprint arXiv:2110.07298*.

Yujia Qin, Jiajie Zhang, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2022. Elle: Efficient lifelong pre-training for emerging data. *arXiv preprint arXiv:2203.06311*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. 2023. Progressive prompts: Continual learning for language models. *arXiv preprint arXiv:2301.12314*.

Philip L Roe. 1981. Approximate riemann solvers, parameter vectors, and difference schemes. *Journal of computational physics*, 43(2):357–372.

Gobinda Saha, Isha Garg, and Kaushik Roy. 2021. Gradient projection memory for continual learning. *ICLR*.

James Seale Smith, Lazar Valkov, Shaunak Halbe, Vyshnavi Gutta, Rogerio Feris, Zsolt Kira, and Leonid Karlinsky. 2024. Adaptive memory replay for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3605–3615.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. 2024a. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. 2024b. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Mingyang Wang, Heike Adel, Lukas Lange, Jannik Strötgen, and Hinrich Schütze. 2024c. Rehearsal-free modular and compositional continual learning for language models. *arXiv preprint arXiv:2404.00790*.

Qing Wang, Xue Han, Jiahui Wang, Lehao Xing, Qian Hu, Lianlian Zhang, Chao Deng, and Junlan Feng. 2025. Multipl-moe: Multi-programming-lingual extension of large language models through hybrid mixture-of-experts. *arXiv preprint arXiv:2508.19268*.

Runqi Wang, Yuxiang Bao, Baochang Zhang, Jianzhuang Liu, Wentao Zhu, and Guodong Guo. 2022a. Anti-retroactive interference for lifelong learning. In *European Conference on Computer Vision*, pages 163–178. Springer.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuan-Jing Huang. 2023. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10658–10671.

Yaqing Wang, Sahaj Agarwal, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. 2022b. Adamix: Mixture-of-adaptations for parameter-efficient model tuning. *arXiv preprint arXiv:2205.12410*.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. 2022c. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 139–149.

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*.

Yiqun Yao, Zheng Zhang, Jing Li, and Yequan Wang. 2023. Masked structural growth for 2x faster language model pre-training. *arXiv preprint arXiv:2305.02869*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Weixiang Zhao, Shilong Wang, Yulin Hu, Yanyan Zhao, Bing Qin, Xuanyu Zhang, Qing Yang, Dongliang Xu, and Wanxiang Che. 2024a. Dapt: A dual attention framework for parameter-efficient continual learning of large language models. *arXiv e-prints*, pages arXiv–2401.

Weixiang Zhao, Shilong Wang, Yulin Hu, Yanyan Zhao, Bing Qin, Xuanyu Zhang, Qing Yang, Dongliang Xu, and Wanxiang Che. 2024b. Sapt: A shared attention framework for parameter-efficient continual learning of large language models. *arXiv preprint arXiv:2401.08295*.

Xuyang Zhao, Huiyuan Wang, Weiran Huang, and Wei Lin. 2024c. A statistical theory of regularization-based continual learning. *arXiv preprint arXiv:2406.06213*.

Kai Zhu, Wei Zhai, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. 2022. Self-sustaining representation expansion for non-exemplar class-incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9296–9305.

A Related works

A.1 Continual learning in LLMs

Continual learning in LLMs refers to the paradigm where LLMs sequentially acquire new knowledge from non-stationary data distributions while preserving learned capabilities (Wang et al., 2024a; Wu et al., 2024), and can be categorized into these types: **Replay-based approach** approximates and recovers old knowledge by storing old training samples (Chaudhry et al., 2019; Smith et al., 2024), or extracting information from prior feature representations (Zhu et al., 2022; Petit et al., 2023). **Regularization-based approach** adds explicit regularization terms to balance the old and new tasks, usually requiring storing a frozen copy of the old model (Kirkpatrick et al., 2017; Zhao et al., 2024c). **Optimization-based approach** explicitly designs the optimization programs to achieve continual learning (Wang et al., 2022a). **Architecture-based approach** investigates dynamic model expansion (Yao et al., 2023; Han et al., 2025; Qin et al., 2022; Wang et al., 2025), combined with parameter isolation mechanisms (Wang et al., 2023; Zhao et al., 2024a) to minimize task interference in sequential learning scenarios.

A.2 Parameter-Efficient Tuning (PET)

PET has emerged as a resource-efficient approach for model adaptation (Coleman et al., 2025), aiming to optimize performance while updating only a small number of trainable parameters to reduce computational costs. Some works train adapters for the downstream task while keeping the pre-trained model parameters frozen (He et al., 2023; Wang et al., 2022b). Another line of research explores the integration of trainable tensors, called soft prompts, into model input representations (Lester et al., 2021; Liu et al., 2024c). Recently, reparameterization methods such as low-rank adaptations (LoRA) have garnered significant attention (Hu et al., 2022b; Dettmers et al., 2023; Liu et al., 2024b). These approaches avoid direct modification of original weight matrices, instead updating parameters through transformation functions operating on smaller parameter sets.

B Experiemtal setups

B.1 Metrcis

Forgetting measure(FM) is also applied to calculate the memory stability of models. The forgetting of a task is calculated by the difference between its maximum performance obtained in the past and its current performance:

$$f_{j,} = \max_{i \in \{1,\ldots,t-1\}} (a_{i,j} - a_{t,j}), \forall j < t \qquad (15)$$

FM at the t-th task is the average forgetting of

Table 4: Instructions for different tasks

Task	Prompt
NLI	What is the logical relationship between the "sentence 1" and the "sentence 2"? Choose one from the options.
QQP	Whether the "first sentence" and the "second sentence" have the same meaning? Choose one from the options.
SC	What is the sentiment of the following paragraph? Choose one from the options.
TC	What is the topic of the following paragraph? Choose one from the options.
BoolQA	According to the following passage, is the question true or false? Choose one from the options.
MultiRC	According to the following passage and question, is the candidate answer true or false? Choose one from the options.
WiC	Given a word and two sentences, whether the word is used with the same sense in both sentences? Choose one from the options.
COPA	Given a prompt sentence, a question, and two possible answers, which option is more reasonable to answer the question. Choose one from two options.

all old tasks:

$$FM_t = \frac{1}{t-1}\sum_{j=1}^{t-1} f_{j,t} \tag{16}$$

FWT evaluates the average influence of all old tasks on the current t-th task:

$$FWT_t = \frac{1}{t-1}\sum_{j=2}^{t}(a_{j,j} - \tilde{a}_j) \tag{17}$$

where $\tilde{a}_j$ is the accuracy of a base model trained with D_j for the j-th task.

B.2 Baselines

We conduct comprehensive comparisons between our method and 10 baseline models, whose introductions are detailed as follows:

SeqFT (de Masson D'Autume et al., 2019): trains all model parameters sequentially across tasks without employing any regularization or replay techniques.

PerTaskFT: trains a separate model for each task.

Replay (Chaudhry et al., 2019): is fine-tuned on all parameters with a memory buffer mechanism, replaying stored prior samples to prevent knowledge forgetting.

EWC (Kirkpatrick et al., 2017): performs full-model fine-tuning with regularization constraints designed to preserve parameters critical for previously learned tasks.

LwF (Li and Hoiem, 2017): constrains the shared representation layer to be similar to its original state before learning new tasks.

L2P (Wang et al., 2022c): dynamically selects and updates prompts from the pool in an instance-wise manner based on input characteristics.

LFPT5 (Qin and Joty, 2021): implements continuous soft prompt training for direct task solution and training sample generation.

SeqLoRA: trains the fixed-size LoRA parameters on a sequence of tasks without any regularization or replaying techniques.

IncLoRA: sequentially acquires a series of new tasks through incremental LoRA parameter expansion, without any regularization or replaying techniques.

MOCL (Wang et al., 2024c): continually adds new trainable PEFT parameters (LoRA) to language models and composes them with existing modules.

MoELoRA (Luo et al., 2024): considers LoRA as a Mixture of Experts, harnessing the collective modeling capacity of multiple experts to handle different domains while retaining LoRA's parameter-efficient characteristics.

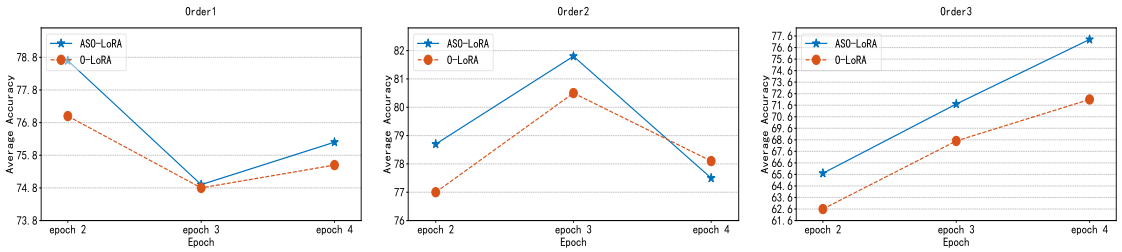
O-LoRA (Wang et al., 2023): learns new tasks in different vector subspaces (low-rank) that are kept orthogonal to each other to prevent catastrophic forgetting.

B.3 Task details

We list the sequences of tasks used in our experiments in Table 5, while Table 4 provides representative task instruction templates.

B.4 Comparison with CL methods

In Table 6, we compare ASO-LoRA with common CL methods. Our approach shows four distinct advantages: rehearsal-free, parameter-efficient, task-id-available, and knowledge-transferable.



(a) Order1 (b) Order2 (c) Order3

Figure 7: ASO-LoRA vs. O-LoRA on Average Accuracy (AA) during the CL stage on CL standard benchmarks (T5 structure), as new tasks arrive continuously. Each arrival of a new task corresponds to one epoch of training. The blue line represents the trend of ASO-LoRA'AA as tasks increment, while the red line corresponds to that of O-LoRA.

Table 5: Six different orders of task sequences used for continual learning experiments. These orders are following Wang et al. (2023) and Razdaibiedina et al. (2023)'s works.

Order	Task Sequence
1	dbpedia→amazon→yahoo→ag
2	dbpedia→amazon→ag→yahoo
3	yahoo→amazon→ag→dbpedia
4	mnli → cb → wic → copa → qqp → boolqa → rte → imdb → yelp → amazon → sst-2 → dbpedia → ag → multirc → yahoo
5	multirc → boolqa → wic → mnli → cb → copa → qqp → rte → imdb → sst-2 → dbpedia → ag → yelp → amazon → yahoo
6	yelp → amazon → mnli → cb → copa → qqp → rte → imdb → sst-2 → dbpedia → ag → yahoo → multirc → boolqa → wic

C Supplementary Experiments

C.1 Performance on other Transformer-based structures

To further validate the generalizability of ASO-LoRA across Transformer-based architectures, we extend our experiments on the decoder-only LLaMA-7B model using Longer CL benchmarks. As evidenced in Table 7, ASO-LoRA achieves significantly superior performance, with an average accuracy improvement of 5.4% compared to O-LoRA. These experimental results further confirm the plug-and-play adaptability of ASO-LoRA, demonstrating its broad applicability across various Transformer-based models in more complex continual learning scenarios.

Table 6: The comparison between ASO-LoRA and other continual learning methods. Specifically, RF indicates whether the method is rehearsal-free. PE indicates whether the method is parameter efficient. TIF indicates whether task identify is available during inference. KT indicates whether the method enables knowledge transfer.

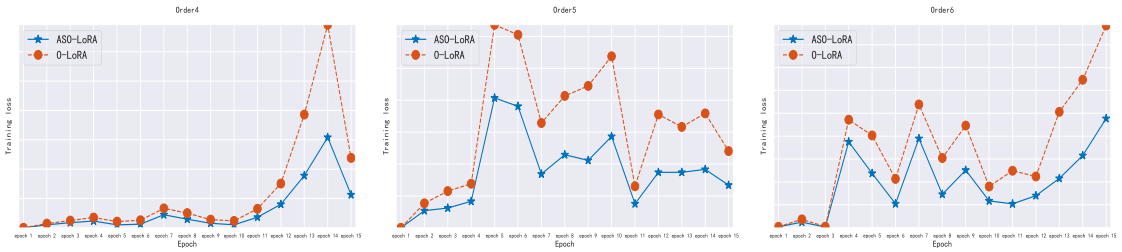
Method	RF	PE	TIF	KT
EWC (Kirkpatrick et al., 2017)	✓		✓	✓
LwF (Li and Hoiem, 2017)	✓			✓
L2P (Wang et al., 2022c)	✓	✓	✓	
LFPT5 (Qin and Joty, 2021)		✓	✓	✓
MoELoRA (Luo et al., 2024)	✓	✓		
O-LoRA (Wang et al., 2023)	✓	✓	✓	
ASO-LoRA	✓	✓	✓	✓

Table 7: The main AA results on Longer benchmarks with the Llama-7B model, after training on the last task.

Model	**Longer benchmarks**			
	Order-4	**Order-5**	**Order-6**	**Avg.**
SeqLoRA	0	11.9	9.0	6.9
IncLoRA	26.8	28.3	36.2	30.4
O-LoRA	46.4	61.9	58.1	55.5
ASO-LoRA	**59.1**	**64.5**	**59.2**	**60.9**

C.2 The impact of Soft Orthogonality on Llama structure

We further analyze the task-specific performance of the final model on the LLaMA architecture after completing all training tasks, as detailed in Table 8. ASO-LoRA outperforms other baselines on almost all individual tasks, proving that our soft orthogonality can generally leverage the inter-task knowledge and potential knowledge transfer on broader structures.



(a) Order4 (b) Order5 (c) Order6

Figure 8: ASO-LoRA vs. O-LoRA on training loss during the CL stage on CL longer benchmarks, as new tasks arrive continuously. Each arrival of a new task corresponds to one epoch of training. The blue line represents the trend of ASO-LoRA's training loss, while the red line corresponds to that of O-LoRA.

Table 8: The results on individual tasks after completing training on the final task with Llama, across Order 1, Order 2, and Order 3. The left-to-right ordering of benchmarks corresponds to the task training order.

Model		CL benchmarks per-task results			
	Sequences	Dbpedia	Amazon	Yahoo	Ag
Order1	IncLoRA	8.7	0.0	31.4	90.7
	O-LoRA	96.3	41.6	66.1	90.9
	ASO-LoRA	**98.3**	**56.3**	65.7	80.0
	Sequences	Dbpedia	Amazon	Ag	Yahoo
Order2	IncLoRA	52.2	23.5	51.5	69.4
	O-LoRA	97.8	45.4	89.0	67.9
	ASO-LoRA	**97.9**	**51.3**	78.9	**70.6**
	Sequences	Yahoo	Amazon	Ag	Dbpedia
Order3	IncLoRA	4.7	26.2	12.6	98.5
	O-LoRA	64.0	61.1	82.4	98.6
	ASO-LoRA	**69.4**	58.2	**86.0**	**98.9**

C.3 The impact of Soft Orthogonality on Generalization Ability of LLMs

Following the experimental setup of Wang et al. (2023), we investigate the impact of ASO-LoRA on the generative performance of LLMs in continual learning scenarios. We fine-tune LLaMA-7B using the Alpaca dataset (Taori et al., 2023) and subsequently evaluate the model on the MMLU benchmark (Hendrycks et al., 2020). As shown in Table 9, ASO-LoRA achieves an average accuracy of 34.5%, which is a 0.9% improvement compared to O-LoRA without soft orthogonal fine-tuning, demonstrating its efficacy in preserving generative capabilities for unknown tasks.

C.4 The correlation between task similarity and CL performance

To validate that the relaxation of orthogonality induced by increased task similarity enhances performance in CL, we evaluate the correlation between task similarity and CL performance using the T5 model on both CL and longer benchmarks.

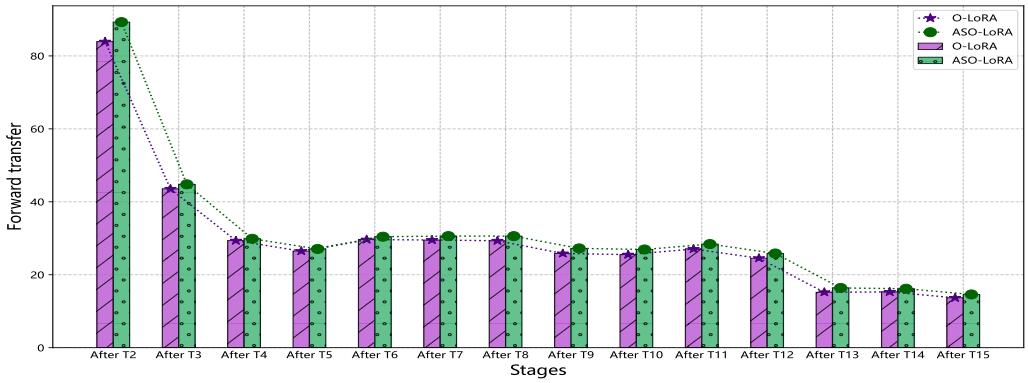
Table 9: The performance of different CL methods applied to the LLaMA-7B model fine-tuning with LoRA using the Alpaca dataset. These methods are evaluated on MMLU(zero-shot).

Method	MMLU
Llama-7B	34.4
Llama-7B(w/ LoRA)	37.5
Llama-7B(w/ LoRA & CL)	23.3
O-LoRA(w/ LoRA & CL)	33.6
ASO-LoRA(w/ LoRA & CL)	34.5

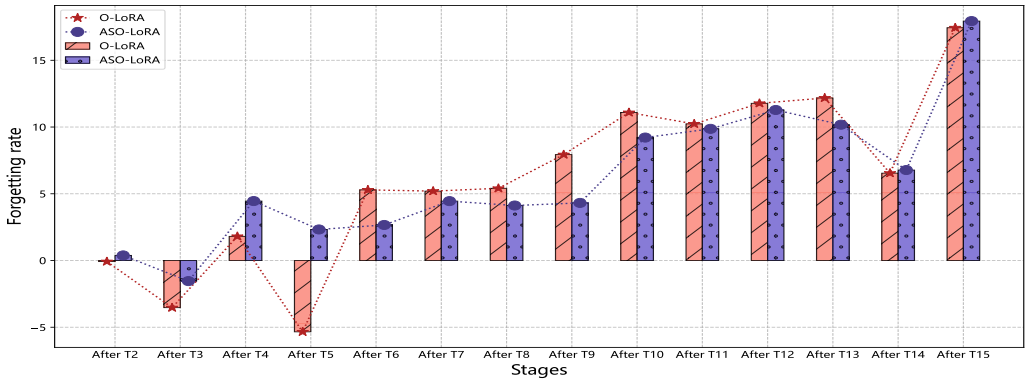
Table 10: The Pearson correlation between task similarity and CL performance.

Correlation	CL benchmarks	Longer benchmarks
Pearson Coefficient(r)	0.94	0.74
P-value	0.20	0.002

As illustrated in the Table 10, a clear positive correlation is observed. This suggests that moderate overlap between task-specific subspaces facilitates effective knowledge transfer, thereby enhancing overall CL performance. As the task sequence lengthens, the representation spaces of individual tasks exhibit increasingly complex interactions, leading to a slight reduction in the correlation coefficients. This trend aligns with our expectation, reflecting the inherent difficulty of managing inter-task interference in long-term CL.



(a) FWT



(b) FM

Figure 9: ASO-LoRA vs. O-LoRA on Order4 using metrics FWT and FM. Higher FWT values demonstrate more effective utilization of prior knowledge and superior knowledge transfer. Lower FM values indicate stronger resistance to catastrophic forgetting and better knowledge retention capabilities.

D Visualization

D.1 The relationships among tasks on Longer benchmarks

We employ the violin plot to visualize the similarity coefficient w^{attr} between the new task and historical tasks, as proposed in Eq.(13), analyzing their inter-task correlations and mutual influences.

As illustrated in Figure 10, the higher similarity coefficient between Task Ag News and historical tasks indicates a positive influence of Ag News on earlier tasks. As evidenced in Table 3, ASO-LoRA obviously achieves a 21.8% performance improvement on Task Ag's previous task Amazon compared to O-LoRA. These results demonstrate that Soft Orthogonality productively leverages previously acquired task knowledge and effectively facilitates knowledge transfer.

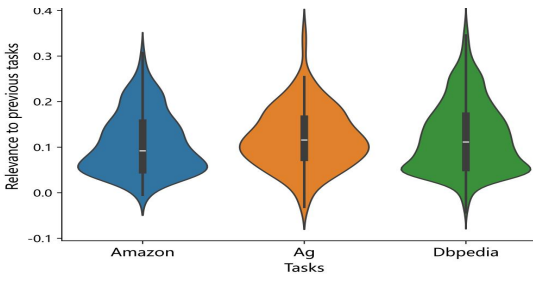


Figure 10: The Violin Plot of w^{attr} on T5-large, displaying the distribution of similarity coefficient w^{attr} across different tasks during the training stages. Central white dot represents the median or mean of the data, while a wider section indicates higher data density (more points near that value).

D.2 The trend of Performance in the Continual learning stages

Standard CL benchmarks: As shown in Figure 7, we compare Average Accuracy trajectories of ASO-LoRA and O-LoRA during CL stages on standard CL benchmarks. Throughout the whole CL stages, ASO-LoRA exhibits consistent performance and generally outperforms O-LoRA in overall results. This further demonstrates that soft orthogonality between LoRA blocks corresponding to different tasks generally enhances the expression of related knowledge, while also achieving a favorable balance between mitigating catastrophic forgetting and facilitating knowledge transfer.

We also investigate the training loss trajectories of ASO-LoRA and O-LoRA during CL stages on longer benchmarks, depicted in Figure 8. While following the same overall trend, ASO-LoRA exhibits lower and more stable training loss than O-LoRA, further reinforcing the suitability of ASO-LoRA for continual learning scenarios.

D.3 The effect on catastrophic forgetting and knowledge transfer

To further comprehensively evaluate Soft Orthogonality's efficacy in mitigating catastrophic forgetting and facilitating knowledge transfer, we demonstrate the FM and FWT results of ASO-LoRA and O-LoRA on longer benchmarks.

Figure 9(a) reveals that ASO-LoRA more effectively leverages knowledge from previous tasks even in more complex continual learning scenarios, achieving significantly superior knowledge transfer over O-LoRA across all stages. These results confirm the advantage of our method in knowledge transfer.

As illustrated in Figure 9(b), as the number of tasks increases, ASO-LoRA outperforms O-LoRA in mitigating knowledge forgetting while preserving acquired knowledge across most stages. However, in the final two stages, O-LoRA exhibits less knowledge forgetting than ASO-LoRA. These findings indicate that striking an optimal balance between knowledge transfer and knowledge retention remains an issue worthy of further investigation.

E Usage of AI Assistants

We appreciate the assistance provided by GPT-4 (Achiam et al., 2023b) in writing aid and sentence-level polishing.