Learn more, but bother less:

parameter efficient continual learning

Fuli Qiao and Mehrdad Mahdavi

Machine Learning and Optimization Lab
Department of Computer Science & Engineering
Pennsylvania State University

PennState

NEURAL INFORMATION PROCESSING SYSTEMS

Introduction

Significance and Limitation of Large Language Models

- Large Language Models (LLMs) have demonstrated exceptional performance across a broad spectrum of tasks, significantly revolutionizing the landscape in diverse areas driven by artificial intelligence. **However:**
  - **Limitation of Full Fine-Tuning**: Computationally expensive in adapting pre-trained models to a large number of downstream tasks.
  - **Continual Learning Challenges**:
    - **Catastrophic Forgetting**: when learning multiple sequential tasks, model performance on previous tasks significantly deteriorates upon training with new data.
    - **Forward Transfer**: harnessing knowledge from old tasks to enhance the learning of new tasks.

Parameter-Efficient Tuning (PET) for Continual Learning (CL)

- **Low-rank Adaptation**: LoRA [1] and its variants have been proposed to prompt parameter-efficient learnings for LLMs.
- **Existing PET methods for CL** primarily focused on mitigating forgetting issue [2], often overlook the equally important objective of facilitating forward knowledge transfer.

References

[1] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685, 2021.

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

Introduction

Knowledge Transfer among Tasks

- **Non-PET in CL**: while existing non-PET knowledge transfer in CL have distinctive approaches, they are not directly applicable to CL in PET framework due to prohibitive computational costs.
- **Knowledge Transfer in Parameter-Efficient Fine-Tuning for LLMs**: parametric knowledge transfer paradigm [3] uses knowledge embedded within a teacher's parameters by extracting task-specific parameters and injecting them into a student model via sensitivity metrics, however, such methods do **NOT** exist in CL for LLMs.

Motivation Experiments

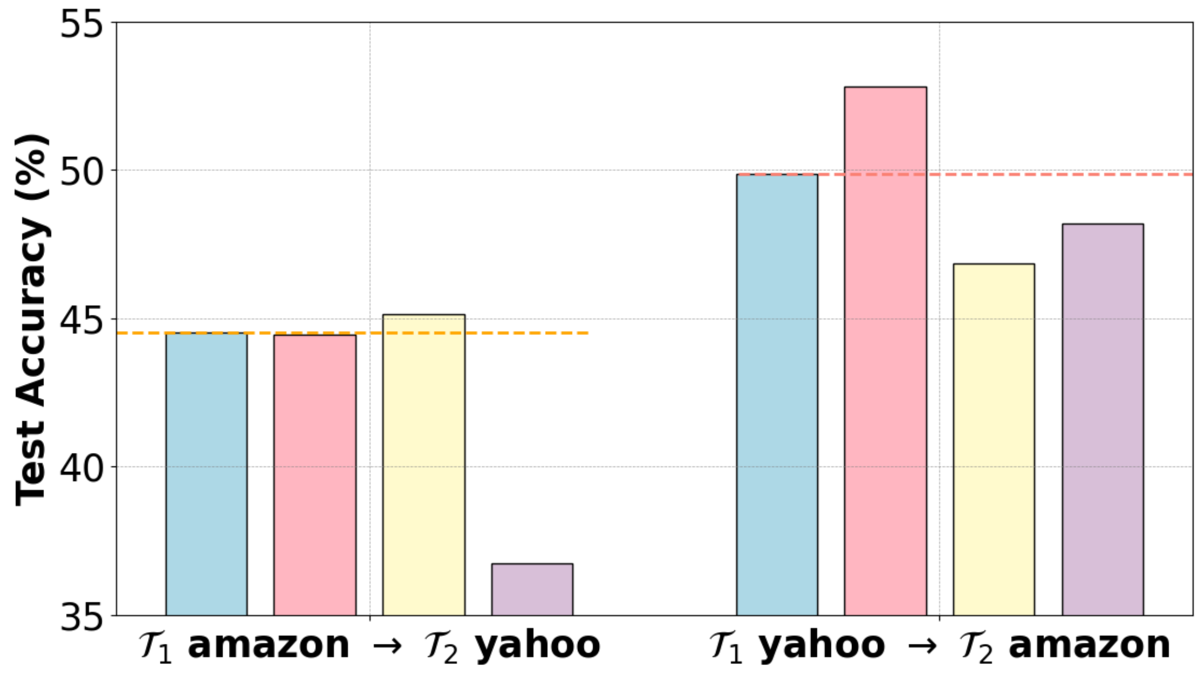


Figure 1: Comparison performance of the model after training task $\mathcal{T}_2$ with different layers replacement.

	$\mathcal{T}_1$ (amazon) → $\mathcal{T}_2$ (yahoo)			
	no	top 4	**top 9**	all
$\mathcal{T}_1$	15.4	15.0	**16.6**	0.2
$\mathcal{T}_2$	73.6	73.9	**73.7**	73.3
	$\mathcal{T}_1$ (yahoo) → $\mathcal{T}_2$ (amazon)			
	no	**top 4**	top 9	all
$\mathcal{T}_1$	46.8	**50.7**	39.1	42.5
$\mathcal{T}_2$	52.9	**54.9**	54.5	53.9

Table 1: Testing accuracy of $\mathcal{T}_1$ and $\mathcal{T}_2$ after training $\mathcal{T}_2$ with different layer replacements, highlighting the best-performing strategy as shown in Fig. 1.

References

[3] Ming Zhong, Chenxin An, Weizhu Chen, Jiawei Han, and Pengcheng He. Seeking neural nuggets: Knowledge transfer in large language models from a parametric perspective. In The Twelfth International Conference on Learning Representations, 2024.

Seeking to explore a new dimension in CL for LLMs

*How can we effectively inject knowledge from previous tasks into new tasks (for improving **generalization**) while maintaining the orthogonality of each task's low-rank subspaces (for mitigating **forgetting**) to facilitate parameter-efficient continual learning?*

Contribution

Novel Parameter-Efficient Continual Learning Framework for LLMs

- Balance generalization through parametric knowledge transfer and mitigation of forgetting through low-rank orthogonal subspace learning for new tasks

Superior Performance over Existing State-of-the-art Approaches

- Through comprehensive evaluations, our method demonstrates superior performance over existing state-of-the-art approaches on standard continual learning benchmarks

In-depth Analysis for Parametric Knowledge within CL for LLMs

- Provide in-depth analysis to deepen understanding of the dynamics of parametric knowledge within CL for LLMs, pinpointing critical factors that drives its effectiveness

CL Maestro: Learn More but Bother Less

Continual Learning (CL) Problem Setup

- A sequence of tasks $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_T\}$ over time, each $\mathcal{T}_k$ with data distribution $\mathcal{D}_k$ and a separate target dataset $\mathcal{S}_k = \{(\boldsymbol{x}_{k,i}, y_{k,i})\}_{i=1}^{n_t}$ where $\boldsymbol{x}_{k,i} \in \mathcal{X}_k$ and $y_{k,i} \in \mathcal{Y}_k$
- Incremental SVD-based low-rank matrix $\boldsymbol{U}^k\boldsymbol{\Sigma}^k\boldsymbol{V}^k$ to fine-tune task $\mathcal{T}_k$, where $\boldsymbol{U}^k \in \mathbb{R}^{d_1\times r}$, $\boldsymbol{V}^k \in \mathbb{R}^{r\times d_2}$, and $\boldsymbol{\Sigma}^k \in \mathbb{R}^{r\times r}$ (singular values $\{\lambda_i\}_{1\le i\le r}$ with $r \ll \min(d_1, d_2)$), and to enforce orthogonality, use regularizer: $\mathcal{R}(\boldsymbol{U}, \boldsymbol{V}) = \|\boldsymbol{U}^\top\boldsymbol{U} - \boldsymbol{I}\|_\text{F}^2 + \|\boldsymbol{V}\boldsymbol{V}^\top - \boldsymbol{I}\|_\text{F}^2$
- Goal: $\max_{\boldsymbol{\theta}} \sum_{k=1}^{T} \sum_{(\boldsymbol{x},y)\in\mathcal{T}_k} \log p_{\boldsymbol{\theta}}(y\,|\,\boldsymbol{x})$, where $\boldsymbol{\theta} = \boldsymbol{W}_0 + \sum_{k=1}^{T} \boldsymbol{U}^k\boldsymbol{\Sigma}^k\boldsymbol{V}^k$ and $\boldsymbol{W}_0$ is pre-trained model

Two Stages of Our Method (LB-CL)

- **(i)** Learning from knowledge extraction and injection, which transfers knowledge from previously learned tasks to new tasks by incremental SVD triplet (a singular value and its corresponding singular vectors) sensitivity metric
- **(ii)** Training in Orthogonal Subspaces, which keeps low-rank subspaces of new tasks orthogonal to those of old tasks

CL Maestro: Learn More but Bother Less

Two Stages of Our Method (LB-CL)

- **(i)** Learning from knowledge extraction and injection, which transfers knowledge from previously learned tasks to new tasks by incremental SVD triplet (a singular value and its corresponding singular vectors) sensitivity metric
- **(ii)** Training in Orthogonal Subspaces, which keeps low-rank subspaces of new tasks orthogonal to those of old tasks

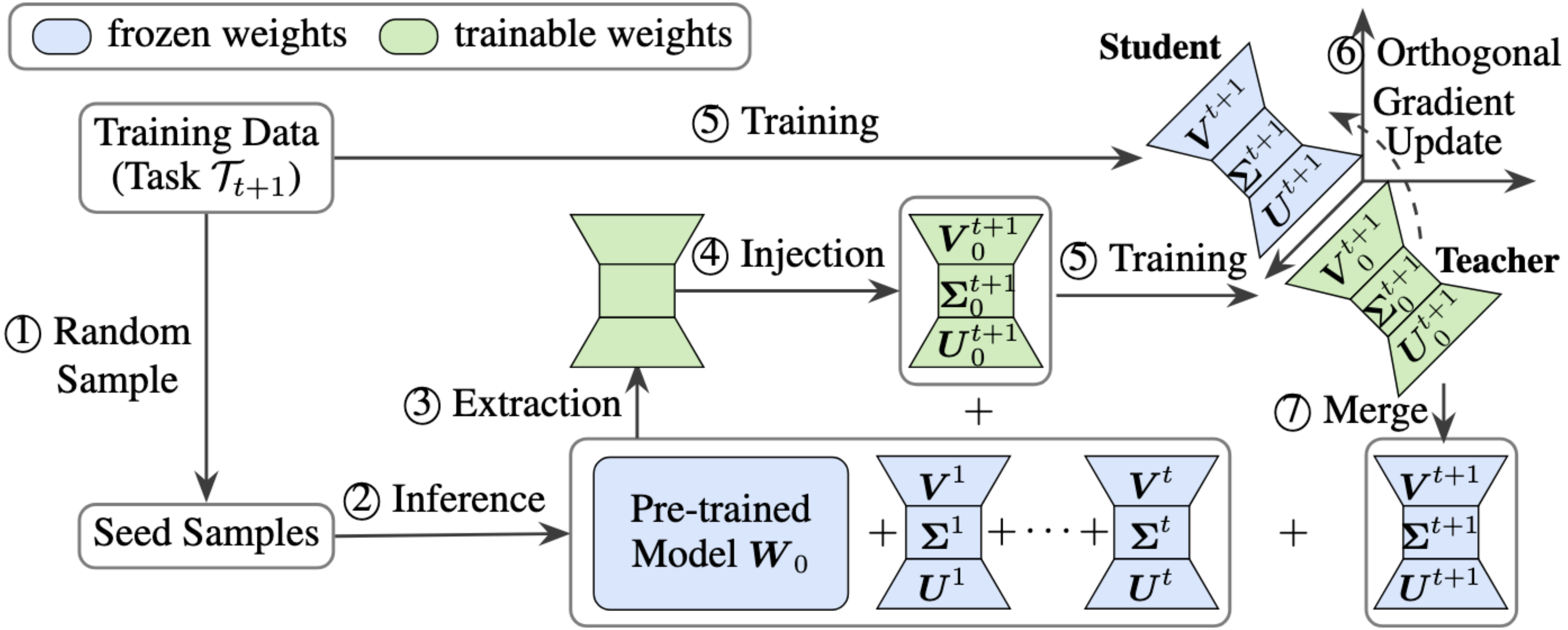


Figure 2: Overview of our LB-CL framework. Starting with the pre-trained model including SVD weights of previous tasks, sensitivity metrics are calculated using a set of seed samples, facilitating the extraction of task-specific knowledge. Subsequently, the extracted layer triplets initialize SVD weights for the new task. Then, the new task is trained in an orthogonal subspace, employing orthogonal gradient projection to minimize forgetting.

Experiments on Continual Learning Benchmarks

- Metric: Define the testing accuracy on task $\mathscr{T}_i$ after training on task $\mathscr{T}_j$ as $a_{i,j}$. The main metric for evaluation is **Average Accuracy (AA)**, calculated as the mean accuracy across all tasks after training on the last task: $\frac{1}{T}\sum_{i=1}^{T} a_{i,T}$

Table 2: Testing performance on two standard CL benchmarks with T5-large.

	Standard CL Benchmark				**Large Number of Tasks**			
	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	7.3	7.4	7.4
SeqLoRA	39.5	31.9	46.6	39.3	4.9	3.5	4.2	4.2
IncLoRA	63.4	62.2	65.1	63.6	63.0	57.9	60.4	60.5
SeqSVD	40.0	63.3	44.9	49.4	13.7	13.8	12.2	13.2
Replay	50.3	52.0	56.6	53.0	54.5	54.3	53.5	54.1
EWC	46.3	45.3	52.1	47.9	44.9	44.0	45.4	44.8
LwF	52.7	52.9	48.4	51.3	49.7	42.8	46.9	46.5
L2P	59.0	60.5	59.9	59.8	57.7	53.6	56.6	56.0
LFPT5	66.6	71.2	76.2	71.3	69.8	67.2	69.2	68.7
L-CL	75.3	73.5	71.9	73.6	66.5	64.0	69.0	66.5
B-CL	76.4	71.5	75.1	74.3	65.7	66.4	69.2	67.1
NLNB-CL	76.0	73.4	74.0	74.5	67.6	65.3	62.6	65.2
O-LoRA	74.9	75.3	75.9	75.4	**70.5**	65.5	70.5	68.8
LB-CL	**76.9**	**76.5**	**76.8**	**76.7**	68.4	**67.3**	**71.8**	**69.2**
ProgPrompt	76.1	76.0	76.3	76.1	78.7	78.8	77.8	78.4
PerTaskFT	70.0	70.0	70.0	70.0	78.1	78.1	78.1	78.1
MTL	80.0	80.0	80.0	80.0	76.3	76.3	76.3	76.3

Experiments on In-depth Analysis of LB-CL

Different initialization strategies

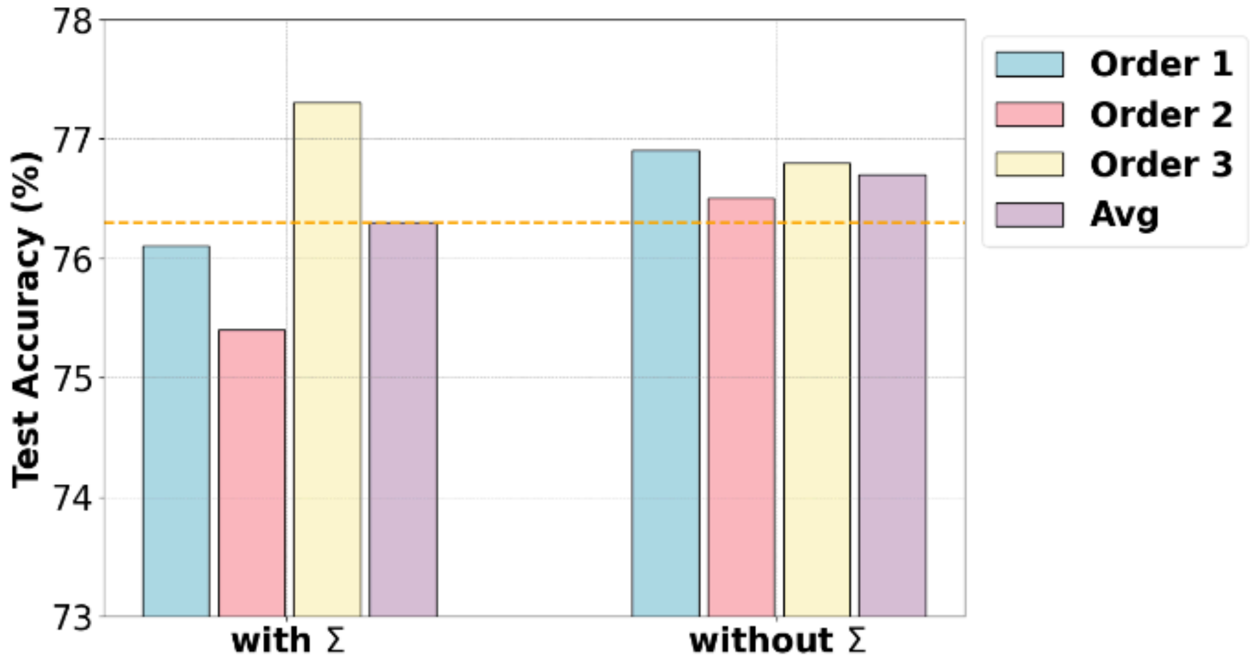


Figure 3: Comparison of different initialization strategies across three orders of standard CL benchmark. The "Avg" value represents the average testing accuracy, illustrating how each strategy stabilizes learning performance.

Number of seed samples

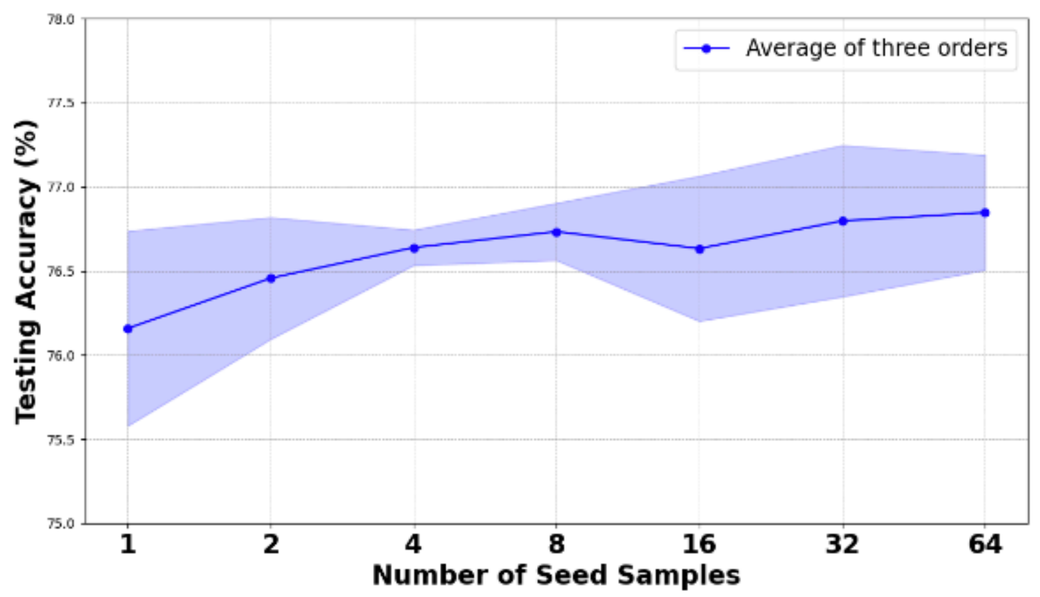


Figure 4: Impact analysis of seed sample quantity on the performance in LB-CL, evaluated across three orders of standard CL benchmark. This investigation highlights the influence of initial seed samples on model effectiveness.

Different pre-trained models

Table 5: Comparisons of different models' performances across three task orders in standard CL benchmark.

(T5-base)	Order			
Method	**1**	**2**	**3**	**avg**
O-LoRA	72.9	72.3	**72.6**	72.6
LB-CL	**73.8**	**74.4**	72.4	**73.5**
(T5-large)	**Order**			
Method	**1**	**2**	**3**	**avg**
O-LoRA	74.9	75.3	75.9	75.4
LB-CL	**76.9**	**76.5**	**76.8**	**76.7**

Optimal Ranks

Table 4: Comparisons of different rank r of low-rank matrix. This experiment is conducted based on T5-large in standard CL benchmark.

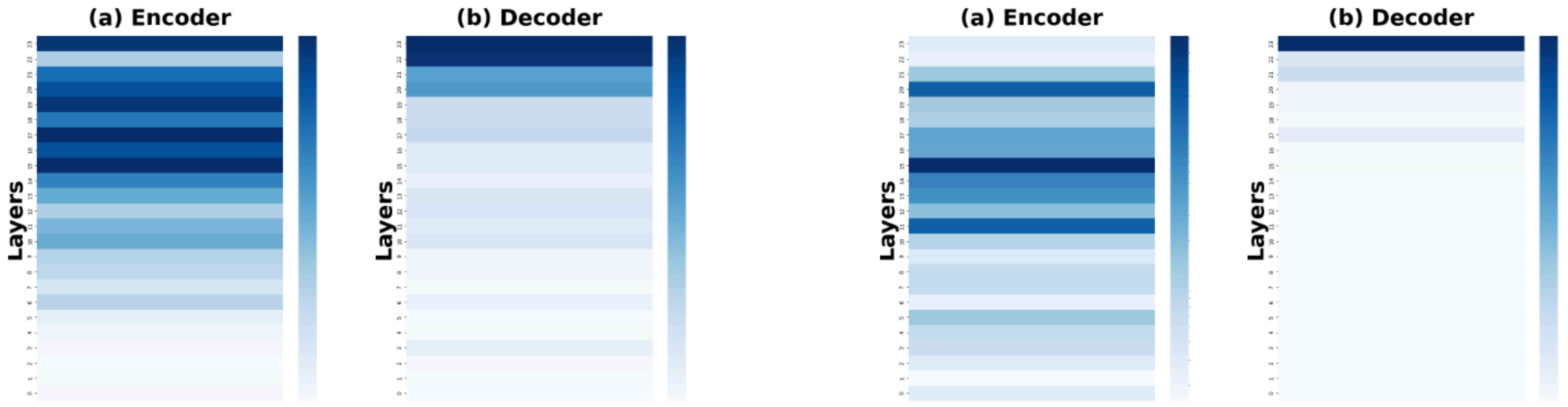
	Order			
r-dim	**1**	**2**	**3**	**avg**
2	76.7	77.2	75.2	76.3
4	77.0	76.8	75.9	76.6
8	76.9	76.5	76.8	76.7
16	77.4	76.0	75.5	76.3
Std	0.25	0.44	0.60	0.18

Training computation costs

Table 3: Comparison of training computation cost between LB-CL and O-LoRA.

Method	GPU Memory	Num of training params/task
O-LoRA	24.82 GB	$r(m+n)$
LB-CL	28.28 GB	$r(m+n)+r$

Parametric Knowledge Distribution



(a) Sensitivity Scores

(b) Fisher Information

Figure 5: Comparison of sensitivity scores and Fisher information of encoder and decoder Layers, and both results are the average results of three task orders in standard CL benchmark.

Summary

- Investigated the balance between overcoming forgetting and achieving generalization in continual learning for LLMs

- Decomposed generalization error with the task low-rank matrix initialization, then proposed a novel framework, LB-CL, explored parametric knowledge transfer between tasks and utilized the inherent forgetting less ability of low-rank matrix

- Instead of storing extra task-specific auxiliary parameters, only utilize low-rank parameters which would be merged into the pre-trained model

- Experiments across standard CL benchmarks validate the effectiveness of LB-CL

- Analyzed critical factors influencing initialization in CL, providing insights for further enhancements in this field