SEEKR: Selective Attention-Guided Knowledge Retention for Continual Learning of Large Language Models

Jinghan He[1,2], Haiyun Guo[1,2*], Kuan Zhu[1,2*], Zihan Zhao[5], Ming Tang[1], Jinqiao Wang[1,2,3,4*]

[1]Foundation Model Research Center, Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]Peng Cheng Laboratory, [4]Wuhan AI Research, [5]Chongqing University
hejinghan2022@ia.ac.cn, {kuan.zhu, haiyun.guo, jqwang}@nlpr.ia.ac.cn

Abstract

Continual learning (CL) is crucial for language models to dynamically adapt to the evolving real-world demands. To mitigate the catastrophic forgetting problem in CL, data replay has been proven a simple and effective strategy, and the subsequent data-replay-based distillation can further enhance the performance. However, existing methods fail to fully exploit the knowledge embedded in models from previous tasks, resulting in the need for a relatively large number of replay samples to achieve good results. In this work, we first explore and emphasize the importance of attention weights in knowledge retention, and then propose a **SE**lective att**E**ntion-guided **K**nowledge **R**etention method (SEEKR) for data-efficient replay-based continual learning of large language models (LLMs). Specifically, SEEKR performs attention distillation on the selected attention heads for finer-grained knowledge retention, where the proposed forgettability-based and task-sensitivity-based measures are used to identify the most valuable attention heads. Experimental results on two continual learning benchmarks for LLMs demonstrate the superiority of SEEKR over the existing methods on both performance and efficiency. Explicitly, SEEKR achieves comparable or even better performance with only 1/10 of the replayed data used by other methods, and reduces the proportion of replayed data to 1%. The code is available at `https://github.com/jinghan1he/SEEKR`.

1 Introduction

Enabling large language models (Achiam et al., 2023; Touvron et al., 2023; Zheng et al., 2024) with human-like continual learning ability is crucial for the long-term practical deployment. It allows for constant knowledge accumulation on new tasks without the need for costly retraining. However, sequentially finetuning the LLMs with new data can lead to catastrophic forgetting (McCloskey and Cohen, 1989), impairing the general ability of the model and its performance on previous tasks.

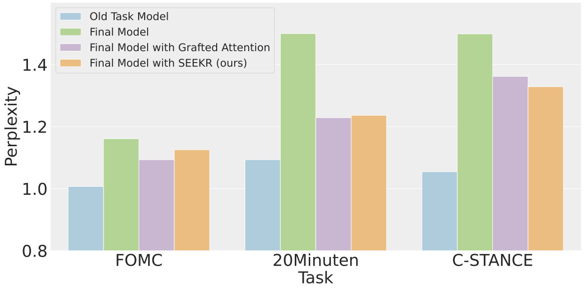


Figure 1: Demonstration of the critical role of attention weights in knowledge retention. We apply DER++ (Buzzega et al., 2020) for continual learning on the TRACE benchmark (Wang et al., 2023c) to obtain multiple old task models and the final model. Grafting the attention weights of the old models onto the final model at inference can maintain better performance on the old tasks. Moreover, the final model obtained by our continual learning method, SEEKR, achieves similar results.

Among the array of continual learning methods (Ke and Liu, 2022), data replay stands out as the most widely adopted strategy in practice due to its simplicity and efficacy (Wang et al., 2024). Based on it, replay-based distillation methods, including DER++ (Buzzega et al., 2020) and subsequent techniques (Qin and Joty, 2021; Kang et al., 2022; Gu et al., 2023), further boost the performance by utilizing memories from both data and model perspectives. Specifically, Buzzega et al., 2020; Qin and Joty, 2021; Gu et al., 2023 distill the output logits of old models for knowledge transfer, and Kang et al., 2022 restrict the changes in important feature maps in the image encoders. However, these works have not fully exploited the potential of knowledge distillation in continual learning for LLMs. They focus on the outputs of network lay-

[*] Corresponding author.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 3254–3266
November 12-16, 2024 ©2024 Association for Computational Linguistics

ers while neglecting the preservation of intricate internal functions. Consequently, a relatively large amount of replay data is required by these methods to achieve good results.

Recently, many studies have investigated the attention weights of different heads to analyze the interpretability of the internal mechanisms in LLMs (Vig and Belinkov, 2019; Wang et al., 2023a). Inspired by this, we explore whether attention weights play a critical role in knowledge retention during continual learning in LLMs. As shown in Figure 1, grafting the attention weights from the LLM of the old tasks to the final LLM after continual learning can maintain better performance on old tasks, which suggests that the attention weights could be crucial to alleviate the catastrophic forgetting problem and achieve more comprehensive knowledge retention[1]. However, naively preserving the attention weights of all heads in the LLM by distillation introduces significant computational costs. Previous studies have observed a functional specialization phenomenon among attention heads in LLMs (Vig and Belinkov, 2019; Jo and Myaeng, 2020; Li et al., 2023), which indicates the susceptibility of attention heads to forgetting and their importance to previous tasks vary. This property allows us to selectively focus on the valuable attention heads for efficient knowledge retention.

To this end, we propose a finer-grained model distillation method called **SE**lective att**E**ntion-guided **K**nowledge **R**etention (**SEEKR**) for continual learning of large language models, which employs attention distillation on the most valuable heads in LLMs to achieve efficient knowledge retention. Specifically, we develop knowledge-retention-oriented head importance measures, which consider both forgettability and task sensitivity, to identify the most valuable heads for distillation. The forgettability, measured by the cumulative changes in attention weights during continual learning, indicates the generality of knowledge and the necessity of distillation. An attention head with higher forgettability indicates a greater need for knowledge retention. The task sensitivity, calculated as the first-order derivative of the task loss, evaluates the importance of maintaining the attention weights of an attention head for a given task. An attention head with greater sensitivity should be prioritized to restrict variations in its attention weights. Using the above two importance scores, SEEKR designs a hierarchical budget allocation mechanism to adaptively select the most valuable attention heads for distillation in a controllable way, which can efficiently regulate the training cost. By using SEEKR, the performance of old tasks can be further maintained as shown in Figure 1.

Extensive experiments are conducted on the recently developed continual learning benchmark for LLMs (Wang et al., 2023c) and the continual learning benchmark on traditional NLP tasks (Wang et al., 2022a). The results consistently demonstrate the superiority of SEEKR in mitigating catastrophic forgetting and maintaining the general capabilities of LLMs. Moreover, as a replay-based method, SEEKR exhibits excellent data efficiency, achieving comparable or better performance with just 1/10 of the replayed data used by the existing methods, reducing the replayed data proportion to only 1%.

Our main contributions are summarized as follows:

- We explore and emphasize the importance of attention weights for knowledge retention, and devise knowledge-retention-oriented measures to identify important attention heads for distillation. The proposed method, SEEKR, can efficiently preserve the finer-grained knowledge in the selected attention heads.
- Extensive experiments validate the superiority of SEEKR, showcasing its data efficiency by using just 1% of replay samples to achieve the comparable or better performance that other methods reach with 10% of replay samples.

2 Preliminary

2.1 Continual Learning for LLMs

Continual learning algorithms aim to accumulate knowledge across sequential tasks. Suppose there are N tasks with the corresponding datasets $\{\mathcal{D}_1, \cdots, \mathcal{D}_N\}$. An LLM, parameterized by θ, are instruction-tuned on each dataset $\mathcal{D}_i$ sequentially to optimize the following objective:

$$L_{task} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_i}\big[-\log p_\theta(\boldsymbol{y}|\boldsymbol{x})\big] \qquad (1)$$

where $\boldsymbol{x}, \boldsymbol{y}$ are the instruction and true answer, respectively. Hereafter, we assume the current task is i and omit the corresponding subscript. In this paper, we study a more common scenario in practice

[1] Attention grafting can only be used during inference with both the source and target models, which is an infeasible solution for continual learning. We employ this technique solely for exploratory experiments.

where a small amount of data from the old tasks $\{R_1, ..., R_N\}$ can be stored in the memory buffer to aid the continual learning process. During training on the current task, replay data are acquired from the memory buffer, and the model is optimized for their previous tasks:

$$L_{replay} = \sum_{k=1}^{i-1} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{R}_k}\big[-\log p_\theta(\boldsymbol{y}|\boldsymbol{x})\big] \quad (2)$$

2.2 Knowledge Distillation for CL

Knowledge distillation (Hinton et al., 2015) is a technique to train a student model to replicate the teacher model's behavior for efficient knowledge transfer. To mitigate forgetting on previous tasks in CL, knowledge distillation is performed between each old model p_{θ_k} and the current model p_θ using replay samples from R_k (Buzzega et al., 2020):

$$L_{ld} = \sum_{k=1}^{i-1} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{R}_k}\big[D_{KL}(p_{\theta_k}(\boldsymbol{y}|\boldsymbol{x}) \| p_\theta(\boldsymbol{y}|\boldsymbol{x}))\big] \quad (3)$$

The predicted logits from the old model $p_{\theta_k}(\boldsymbol{y}|\boldsymbol{x})$ are saved in the memory buffer along with the replay samples and loaded during training as auxiliary supervision signals.

3 Method

In this section, we introduce SEEKR, an efficient replay-based distillation method that identifies valuable attention heads and performs attention distillation for finer-grained knowledge retention.

3.1 Attention-guided Knowledge Retention

To achieve more comprehensive knowledge retention by using less replay data, we perform an elaborate distillation on the key mechanism of LLMs, *i.e.* the attention weights. Specifically, the outputted attention weights of the h-th head in the l-th layer are denoted as $A_{l,h}$:

$$A_{l,h} = \mathrm{softmax}(\frac{Q_{l,h}K_{l,h}^T}{\sqrt{d_k}} + M_{causal}) \quad (4)$$

where Q and K represent the query vectors and the key vectors in the self-attention operation, respectively. M_{causal} is the casual attention mask in LLMs. We use t to index the attention distribution of the t-th query in $A_{l,h}$ and denote it as $A_{l,h,t}$. The attention distributions of query t from each old task model $A^k_{l,h,t}$ and the current model $A_{l,h,t}$ are aligned through the KL divergence loss:

$$L_{ad}(A, A^k) = \sum_{(l,h)\in U} \sum_{t=1}^{|\boldsymbol{x}\oplus\boldsymbol{y}|} D_{KL}(A^k_{l,h,t} \| A_{l,h,t}) \quad (5)$$

where U stands for the set of all attention heads in all layers. $\boldsymbol{x} \oplus \boldsymbol{y}$ is the concatenated sequence of $\boldsymbol{x}$ and $\boldsymbol{y}$, and $|\boldsymbol{x} \oplus \boldsymbol{y}|$ means the length of the whole sequence. In SEEKR, the knowledge distillation is performed at the head level, which can offer more direct and refined regulation on the intricate internal functions of LLMs, achieving a more comprehensive and efficient utilization of the limited replay data.

3.2 Important Head Identification

In practice, distilling all the attention heads in an LLM is costly and unnecessary, as different heads exhibit varying levels of task sensitivity and forgettability. Therefore, we propose a two-dimensional measure to identify the most valuable attention heads for knowledge retention.

3.2.1 Task Sensitivity Measure

For a model adapted to task k, we assess to which extent changes in the attention weights of each head affect the task performance. Following common practice, we resort to Taylor expansion to formalize this influence (Kang et al., 2022):

$$\begin{aligned} \Delta L(\boldsymbol{x},\boldsymbol{y}) &\approx \left\langle \frac{\partial L(\boldsymbol{x},\boldsymbol{y})}{\partial A_{l,h}}, \Delta A_{l,h} \right\rangle_F \\ &\leq || \frac{\partial L(\boldsymbol{x},\boldsymbol{y})}{\partial A_{l,h}} ||_F \cdot ||\Delta A_{l,h}||_F \end{aligned} \quad (6)$$

where $\langle \cdot, \cdot \rangle_F$ and $\| \cdot \|_F$ denote the Frobenius inner product and Frobenius norm, respectively. This inequality demonstrates the upper bound on the increase in task loss due to changes in the attention weights, *i.e.* $\Delta A_{l,h}$. A larger coefficient indicates a higher upper bound for the same changes in $A_{l,h}$. This implies that changes in these attention weights are more likely to increase task loss or degrade task performance, making it crucial to keep them unchanged. Therefore, we take the coefficient to estimate the sensitivity of the task k to $A^k_{l,h}$, which is formulated as:

$$S^k_{l,h} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\in R_k} || \frac{\partial L(\boldsymbol{x},\boldsymbol{y})}{\partial A^k_{l,h}} ||_F \quad (7)$$

The importance scores are then normalized within each layer to obtain $\widetilde{S}^k_{l,h}$, thereby mitigating the impact of varying gradient magnitudes across different layers. During training on the new task, the importance of all previous tasks should be considered. Therefore, the task sensitivity measure for each attention head is defined as:

$$S_{l,h} = \sum_{k=1}^{i-1} \widetilde{S}_{l,h}^{k} \tag{8}$$

3.2.2 Forgettability Measure

The second measure assesses the necessity for performing attention distillation on each attention head. We hypothesize that there exist some attention heads whose attention weights remain relatively stable during continual training on new tasks, suggesting that they are less sensitive to task-specific details and focus more on general or shared knowledge. This hypothesis aligns with prior research (Zhao et al., 2023), which revealed that only a few modules change drastically during continual learning, while others stay relatively stable and may be shared across tasks as common knowledge. Based on this, we propose that stable attention heads may encode general knowledge that is less prone to forgetting, and thus distillation of such heads should be minimized. To this end, we leverage the variability of the attention weights during continual learning to measure the forgettability of the attention head:

$$F_{l,h} = \sum_{k=1}^{i-1} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\in R_k} ||A_{l,h}^{k} - A_{l,h}^{k-1}||_F \tag{9}$$

Higher forgettability scores indicate a greater necessity for distilling these attention heads.

3.2.3 Overall Importance Measure

To identify valuable heads for attention-guided knowledge retention, we fuse the two complementary measures through multiplication, ultimately forming a holistic metric:

$$I_{l,h} = S_{l,h} \cdot F_{l,h} \tag{10}$$

After each task, $S_{l,h}$ and $F_{l,h}$ of each attention head are updated according to Equation 8 and 9, and the overall importance $I_{l,h}$ is re-calculated accordingly.

3.3 Hierarchical Budget Allocation

Algorithm 1 SEEKR

```
Input Initial model $\theta_0$, Datasets $\{\mathcal{D}_i\}_{i=1}^{N}$, Hyper-parameters $\lambda_1, \lambda_2, B_L, B_H, B_T$
Initialize $L, H \leftarrow U$; $S_{l,h}, F_{l,h}, I_{l,h} \leftarrow 0$;
for task $i \leftarrow 1$ to $N$ do
    for epoch $e \leftarrow 1$ to $epochs$ do
        for batch in $(\bigcup_{k=1}^{i-1} R_k) \bigcup \mathcal{D}_i$ do
            Minimize $L$ in Eq. 13;
        end for
    end for
    $R_i \leftarrow \text{Random}(\mathcal{D}_i)$;
    Update $S_{l,h}, F_{l,h}, I_{l,h}$ using Eq. 8-10;
    Update $L, H$ using Eq. 11;
    Randomly select $T$;
end for
```

Based on the above head importance measure, we propose a hierarchical budget allocation strategy to manage the training cost. We define the group of selected layers and heads as L and H, with budgets B_L and B_H. Our strategy involves two steps: (1) Select the top-B_L layers that maximize the layer-wise importance scores $\sum_h I_{l,h}$. (2) Among all the attention heads in all these layers, activate the top-B_H heads for attention distillation. Based on the above process, the set H of the selected heads can be expressed as:

$$\begin{aligned} H &= \underset{(l,h)}{\arg\operatorname{topk}}\{I_{l,h} \mid l \in L\} \\ L &= \underset{l}{\arg\operatorname{topk}} \sum_{h} I_{l,h} \end{aligned} \tag{11}$$

where $\arg\operatorname{topk}_z$ denotes the set of z that achieves the k largest values. k is B_H for H and B_L for L. Additionally, to reduce the $O(n^2)$ cost of distilling the entire attention map, we introduce a query budget B_T and randomly select the queries T for distillation. After determining H and T, we can rewrite Equation 5 as follows:

$$\begin{aligned} L_{ad}(A, A^k) &= \sum_{(l,h)\in H} \sum_{t\in T} D_{KL}(A_{l,h,t}^{k} \| A_{l,h,t}) \\ L_{seekr} &= \sum_{k=1}^{i-1} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\in \mathcal{R}_k} \left[L_{ad}(A, A^k)\right] \end{aligned} \tag{12}$$

Overall, SEEKR sets three types of budgets to allow flexible control over training costs. First, the layer budget adjusts the number of layers for attention-accelerating algorithms or our distillation strategy. Second, the head budget filters out less essential heads and reduces training costs. Lastly, the query budget specifically targets at reducing the costs associated with distilling long texts.

	LLaMA-2-7B-Chat		**Vicuna-7B-v1.5**	
	Order1	**Order2**	**Order1**	**Order2**
SeqFT	47.63 (-11.45)	45.12 (-12.27)	41.91 (-15.29)	45.70 (-12.01)
EWC	48.20 (-9.48)	44.54 (-12.00)	41.88 (-15.57)	49.32 (-8.62)
LwF	41.86 (-6.50)	40.25 (-5.96)	41.19 (-5.54)	42.99 (-4.72)
LFPT5	38.67 (-11.43)	42.26 (-7.43)	41.79 (-8.10)	39.22 (-10.70)
L2P	35.23 (-15.96)	34.63 (-16.86)	32.26 (-16.58)	35.14 (-15.88)
PP	29.41 (-5.79)	21.58 (-8.83)	26.64 (-6.10)	24.88 (-11.54)
O-LoRA	44.64 (-4.20)	42.83 (-9.11)	43.42 (-6.27)	43.87 (-6.37)
Replay (1%)	48.47 (-9.69)	47.04 (-10.24)	48.43 (-9.23)	49.46 (-9.43)
DER++ (1%)	49.22 (-8.32)	46.59 (-10.91)	49.01 (-9.04)	51.09 (-7.85)
SEEKR (1%)	**54.99 (-2.61)**	**54.69 (-2.53)**	**55.78 (-2.64)**	**54.91 (-3.40)**
Replay (10%)	55.67 (-3.96)	53.39 (-4.15)	55.62 (-2.15)	54.57 (-3.41)
DER++ (10%)	55.01 (-3.50)	54.05 (-2.94)	56.06 (-1.17)	55.14 (-3.77)
SEEKR (10%)	**58.27 (0.11)**	**57.27 (-0.47)**	**57.54 (0.47)**	**56.86 (-1.01)**
MTL	59.38		58.18	

Table 1: Comparison with the state-of-the-art methods on TRACE benchmark. The results are obtained by using two popular LLMs with two transfer orders, and are presented in the format of OP (BWT).

3.4 Overall Objective

Combining the above objectives, the overall loss for the new and replay data is formalized as:

$$L = L_{task} + \lambda_1 L_{replay} + (1 - \lambda_1) L_{ld} + \lambda_2 L_{seekr} \tag{13}$$

where λ_1 is a coefficient to balance the text generation loss supervised by true labels and teacher models, and λ_2 is a weighting factor to adjust the magnitude of attention distillation loss. The overall process of SEEKR is shown in Algorithm 1.

4 Experiments

4.1 Experimental Setup

4.1.1 Datasets

CL Benchmark for LLMs. We evaluate our method on TRACE (Wang et al., 2023c), a continual learning benchmark for LLMs that includes eight datasets covering domain-specific knowledge, multilingual capabilities, code generation, and mathematical reasoning. We use the reasoning-augmented version of datasets and conduct experiments under two task orders following the original paper. After continual learning, we assess the performance of the continually learned tasks and the changes in the general ability of LLMs.

CL on Traditional NLP Tasks. SuperNI (Wang et al., 2022a) contains a variety of traditional NLP tasks and can serve as a practical benchmark for continual learning of large language models. Similar to Zhao et al., 2024, we select three datasets for each of the four types of tasks, *i.e.* information extraction, question answering, summarization, and sentiment analysis, to examine the effectiveness of continual learning methods. For each dataset, 1000 samples and 100 samples are randomly sampled for training and testing, respectively.

4.1.2 Metrics

Let $a_{i,j}$ denote the testing performance on the i-th task after training on the j-th task. We report the overall performance (OP) (Chaudhry et al., 2018) and the backward transfer (BWT) (Lopez-Paz and Ranzato, 2017) after training on the last task:

$$OP = \frac{1}{T} \sum_{i=1}^{T} a_{i,T} \tag{14}$$

$$BWT = \frac{1}{T-1} \sum_{i=1}^{T-1} (a_{i,T} - a_{i,i}) \tag{15}$$

Moreover, we also report the general ability (GA) and the delta general ability (DeltaGA) (Wang et al., 2023c) after continual learning. GA is the average performance across evaluation datasets in Table 2 and DeltaGA shows the change in GA compared to the initial model.

	MMLU	GSM	BBH	TydiQA	BoolQ	PIQA	GA (DeltaGA)
LLaMA-2-7B-Chat	46.89	27.14	39.73	16.76	79.79	76.33	47.77
SeqFT	45.16	14.03	32.50	14.84	79.00	75.49	43.50 (-4.27)
Replay (1%)	45.49	12.70	33.46	14.65	78.69	75.65	43.44 (-4.33)
SEEKR (1%)	46.32	20.85	38.52	18.22	80.64	75.79	**46.72 (-1.05)**
Vicuna-7B-v1.5	49.39	23.43	41.12	15.01	81.41	76.77	47.86
SeqFT	46.26	11.68	33.09	13.44	79.97	76.72	43.52 (-4.34)
Replay (1%)	47.14	15.77	33.51	14.14	80.57	76.39	44.59 (-3.27)
SEEKR (1%)	48.83	17.55	38.17	16.32	81.96	77.23	**46.68 (-1.18)**

Table 2: Changes in general language understanding and reasoning abilities after continual learning with different methods. The reported results of all continual learning models are averaged over two task orders.

4.2 Baselines

We compare SEEKR with nine baseline methods: (1) **SeqFT** sequentially finetunes the model without continual learning strategies. (2) **EWC** (Kirkpatrick et al., 2017) regularizes parameter variations based on parameter importance scores. (3) **LwF** (Li and Hoiem, 2017) distills the model of the last task using the current task data. (4) **Replay** finetunes the model with the current task data and a small number of replay samples. (5) **DER++** (Buzzega et al., 2020) saves the logits of the replay samples from the old models for distillation, and combines distillation and replay to reduce forgetting. (6) **LFPT5** (Qin and Joty, 2021) learns a soft prompt to generate pseudo samples of previous tasks for replaying. (7) **O-LoRA** (Wang et al., 2023b) imposes orthogonal constraints on the LoRA matrices for all tasks. (8) **L2P** (Wang et al., 2022b) instantiates a prompt pool for adaptive prompt selection and prompt tuning for individual samples. (9) **PP** (Razdaibiedina et al., 2023) tunes a set of prompts for each task and concatenates them together. In addition, the results of the multi-task trained models are reported as **MTL** and serve as the upper-bound reference.

4.3 Implementation Details

SEEKR is a versatile continual learning method compatible with any transformer-based model. Following Wang et al., 2023c, we conduct our main experiments on two popular LLMs, *i.e.* LLaMA-2-7B-chat (Touvron et al., 2023) and Vicuna-7B-v1.5 (Zheng et al., 2024). We also scale to a larger model Vicuna-13B-v1.5 to validate the effectiveness of SEEKR. All models are trained on 8 NVIDIA Tesla A800 using the DeepSpeed library. The training batch size is 128. For methods not involving parameter-efficient tuning modules, the learning rate is 1e-5. For replay-based methods, the default replay ratio is 1%. For SEEKR, λ_1 in Equation 13 is set to 0.5. λ_2 is 1e3 for a replay ratio of 1% and 1e2 for 10%. The head budget B_H is 128, and the layer budget B_L is 24 by default and 8 for 13B models or a replay ratio of 10%. The query budget B_T is 100. All experimental results were averaged over 3 runs. More implementation details can be found in Appendix B.

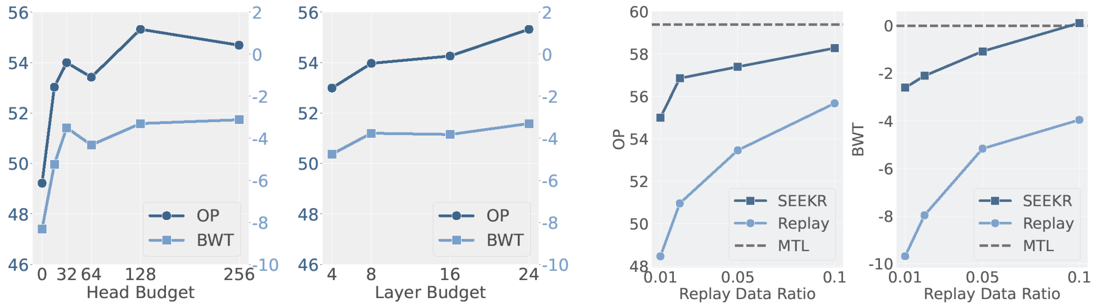
	Order3	Order4
SeqFT	42.62 (-18.12)	50.52 (-9.88)
LwF	43.29 (-15.47)	47.35 (-12.57)
LFPT5	42.05 (-16.26)	46.09 (-14.16)
L2P	32.71 (-22.34)	31.00 (-23.82)
PP	17.96 (-21.27)	12.19 (-29.08)
O-LoRA	30.07 (-24.47)	26.70 (-33.82)
Replay (1%)	55.00 (-4.27)	54.78 (-5.31)
DER++ (1%)	55.89 (-4.51)	53.48 (-5.01)
SEEKR (1%)	**57.04 (-3.15)**	**58.26 (-2.52)**
MTL	61.27	

Table 3: Comparison with the state-of-the-art methods on SuperNI benchmark. The experiments are conducted on LLaMA-2-7B.

4.4 Main Results

Table 1 compares the overall continual learning performance of SEEKR with other baselines on TRACE benchmark. Following Wang et al., 2023c, we also report the changes in the general ability of LLMs after continual learning in Table 2. Similar experiments on the SuperNI benchmark are displayed in Table 3.

SEEKR effectively mitigates catastrophic forgetting of continually learned tasks. Compared to traditional and state-of-the-art continual learning approaches, SEEKR consistently achieves the



(a) Effect of distillation budget

(b) Effect of replay data ratio

Figure 2: Results of SEEKR across different distillation budgets and different replay data ratios.

highest OP and the lowest magnitude of BWT in all settings. Note that the BWT metric specifically captures the resistance of methods to catastrophic forgetting, thus the results demonstrate SEEKR's superiority in maintaining performance on newly learned tasks. Additionally, on the SuperNI benchmark, we achieve the best performance using only a small proportion of replay samples, likely because the benchmark consists of traditional NLP tasks, which are less challenging.

SEEKR fully exploits the small amount of replay data and exhibits excellent data efficiency. Among all replay-based methods, SEEKR stands out with a distinct advantage. On the TRACE benchmark, both Replay and DER++ show limited benefits with a lower ratio of replay data. In contrast, SEEKR demonstrates remarkable performance with just 1% of the samples replayed, achieving comparable or even better results than other methods that replay 10% of the samples. This underscores the ability of SEEKR to maximize the use of a small number of old samples and the inherent knowledge in the old models.

SEEKR is effective in maintaining the general ability of the original LLM. Table 2 exhibits the changes in LLMs' general ability after continual learning. LLMs that are continually trained on new tasks show a decline in general task performance, demonstrating the catastrophic forgetting of their original capabilities. Results validated that SEEKR, which elaborately distills multiple finetuned LLMs with a variety of data, helps to maintain the general capabilities of the model. This could benefit from the fact that our approach preserves the knowledge of the intricate internal functions in LLMs at the attention head level.

	Order1	**Order2**
random	53.25 (-4.63)	52.62 (-5.11)
task-sensitivity-only	53.91 (-4.29)	53.56 (-2.84)
forgettability-only	54.06 (-3.31)	53.63 (-3.11)
both	**54.99** (**-2.61**)	**54.69** (**-2.53**)

Table 4: Ablation study on the head importance measure. The experiments are conducted on LLaMA-2-7B.

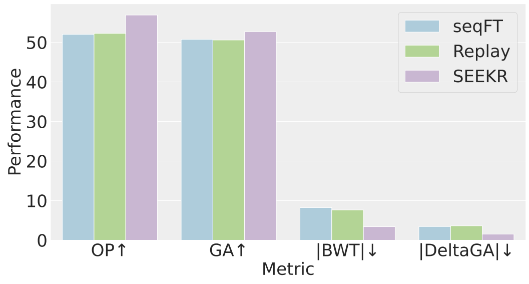


Figure 3: The continual learning performance and the changes of general ability with Vicuna-13B-v1.5.

4.5 Ablation Studies

Effect of distillation budget. Figure 2 (a) exhibits the performance of our method under different budgets. With a fixed layer budget of 24, a larger head budget can lead to better results, but this improvement tends to plateau at a budget of 128. Similarly, the performance improves with an increasing layer budget and reaches its optimum at 24. These results further emphasize the significance of distilling the right attention heads. Distilling less essential attention heads may lead to ineffective work.

Effect of more replay samples. To further explore the potential of SEEKR, we experiment with an increased ratio of replay samples. Meanwhile, we compare SEEKR with Replay to demonstrate its data efficiency. As shown in Figure 2 (b), SEEKR steadily improves performance as the number of replay samples grows. At a replay ratio of 10%, the

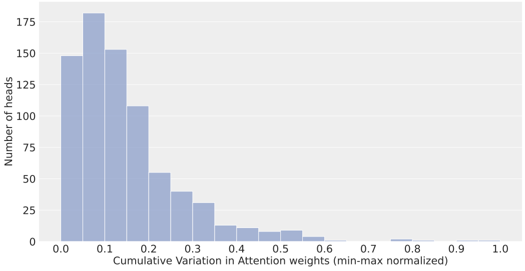


Figure 4: Histogram of the cumulative variation in the attention weights of the attention heads in the model during sequential finetuning.

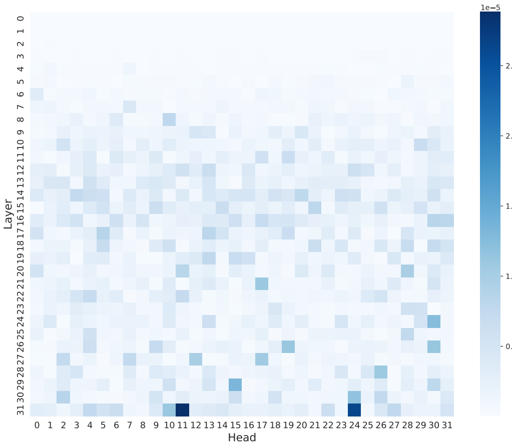


Figure 5: Visualization of the importance scores of all heads in the model.

BWT score exceeds 0, indicating no forgetting or even a positive transfer has been achieved, and the overall performance approximates the upper bound of multi-task training. Moreover, compared with Replay, SEEKR is very data efficient by utilizing only 1% of the old data to achieve the performance of replaying ten times that amount.

Effectiveness of our head importance measure. We present the results of the ablation study on the proposed head importance measure in Table 4 . The results show that the random selection of distilled attention heads noticeably resulted in a higher forgetting indicator, while using either sensitivity-based or variation-based measures helps identify important heads for knowledge retention. Finally, combining both of the above measures produces the best results.

4.6 Discussions

Scale to larger models. To validate the generalizability of SEEKR across different model scales, we conducted additional experiments on a larger model, Vicuna-13B-v1.5. Figure 3 shows that our approach still effectively preserves both the performance of newly learned tasks and the general capabilities of the original model.

Variation in attention weights. To further confirm our hypothesis in Section 3.2.2, we examine the cumulative changes in attention weights of each attention head during sequential finetuning. The results in Figure 4 reveal that most attention heads remain stable throughout the process, while a small proportion undergo significant changes. This observation is similar to prior findings (Zhao et al., 2023) and supports our hypothesis that these stable attention heads do exist, making it reasonable to identify them and avoid unnecessary attention distillation.

Analysis of selected important heads. Figure 5 illustrates that important attention heads are mainly distributed in the middle and deep layers of the model, while almost none are observed in the shallow layers. This aligns with the idea that the shallow layers encode more generalized knowledge and are less susceptible to forgetting. A closer look at Figure 5 further reveals that the importance scores for the deeper layers are concentrated in a few heads, while those for the middle layers are more evenly spread over a larger number of heads. This may be because the heads in the deeper layers are more thoroughly function-specialized.

5 Related Works

5.1 Continual Learning for LLMs

Existing continual learning methods are typically classified into three broad categories: regularization-based methods, replay-based methods, and architectural-based methods. (1) **Regularization-based methods** restrict model variations to alleviate forgetting. Some works penalize changes to important parameters for previously learned tasks (Kirkpatrick et al., 2017; Wang et al., 2023b; He et al., 2023), while others resort to knowledge distillation to maintain the old models' predictions (Li and Hoiem, 2017; Buzzega et al., 2020; Kang et al., 2022). (2) **Replay-based methods** replay data from the old tasks during training on the new task. Experience replay methods (Rebuffi et al., 2017; Wang et al., 2024) design data selection strategies of previous samples, and generative replay (Shin et al., 2017; Qin and Joty, 2021) uses generative models to produce synthetic data from previous tasks. Other methods (Yang et al., 2023) retain old tasks by storing statistical information of the old tasks instead of the original data. (3) **Architecture-based methods** alter the model structure to accommodate different tasks. Recently, this type of methods on LLMs (Wang et al., 2022b; Razdaibiedina et al., 2023) often add parameter-efficient tuning modules for new tasks.

SEEKR falls into the category of replay-based distillation methods and focuses on the preservation of important attention mechanisms in LLMs. Unlike existing output or parameter importance measures (Kirkpatrick et al., 2017; Kang et al., 2022), which focus solely on task loss sensitivity, our head importance measure includes a forgettability aspect. This reflects the susceptibility to forgetting and the generality of knowledge in different heads, thereby determining the necessity for distillation.

5.2 Knowledge Distillation

Knowledge distillation aims to leverage the teacher model's performance and generalize it to the student model (Hinton et al., 2015; Park et al., 2019; Guo et al., 2023). For language models, Sanh et al., 2019 uses the teacher model's generation distribution for each token as a supervision signal for the student model, and some other works (Wang et al., 2020b,a) distill the attention scores of one layer to transfer the knowledge of larger LMs into smaller models. Unlike their objectives of transferring knowledge between models of different sizes, we use attention distillation for knowledge retention. Both our teacher and student models share a similar architecture and are derived from the same pre-trained LLM, which enables head-by-head and layer-by-layer distillation.

6 Conclusion

In this paper, we propose SEEKR, an efficient replay-based distillation method for continual learning in LLMs. SEEKR resorts to attention distillation of important heads for finer-grained knowledge retention, which identifies valuable heads through the proposed knowledge-retention-oriented importance measures. Combined with a hierarchical budget allocation mechanism, SEEKR can ensure its utility across various resource levels. Extensive experiments consistently validated the effectiveness of our method in preserving the performance of newly learned tasks and the original ability of the initial LLMs.

Limitations

Despite the potential benefits of SEEKR, several limitations need to be considered. First, SEEKR is inherently a replay-based approach, which may not be applicable in scenarios where historical data involves privacy concerns. A potential solution is to use SEEKR with pseudo-samples generated by the trained LLM, but this approach requires further exploration. Second, due to computational resource limitations, we did not experiment with larger-scale LLMs like LLaMA-2-70B. Additionally, the application of SEEKR to continual learning with multimodal large language models remains to be explored in the future.

Acknowledgements

This work was supported by National Key R&D Program of China under Grant No.2021ZD0110400, also partly supported by Beijing Natural Science Foundation under Grant 4244099, National Natural Science Foundation of China under Grant No.62276260, Postdoctoral Fellowship Program of CPSF under Grant GZC20232996, China Postdoctoral Science Foundation under Gant 2024M753498.

References

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

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

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. 2020. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930.

Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. 2018. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European conference on computer vision (ECCV)*, pages 532–547.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Jonathan H Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. Tydi qa: A benchmark for information-seeking question answering in ty pologically di verse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Ahmad Ghazal, Tilmann Rabl, Minqing Hu, Francois Raab, Meikel Poess, Alain Crolotte, and Hans-Arno Jacobsen. 2013. Bigbench: Towards an industry standard benchmark for big data analytics. In *Proceedings of the 2013 ACM SIGMOD international conference on Management of data*, pages 1197–1208.

Qiao Gu, Dongsub Shim, and Florian Shkurti. 2023. Preserving linear separability in continual learning by backward feature projection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24286–24295.

Guangyu Guo, Longfei Han, Le Wang, Dingwen Zhang, and Junwei Han. 2023. Semantic-aware knowledge distillation with parameter-free feature uniformization. *Visual Intelligence*, 1(1):6.

Jinghan He, Haiyun Guo, Ming Tang, and Jinqiao Wang. 2023. Continual instruction tuning for large multimodal models. *arXiv preprint arXiv:2311.16206*.

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*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Jae-young Jo and Sung-Hyon Myaeng. 2020. Roles and utilization of attention heads in transformer-based neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3404–3417.

Minsoo Kang, Jaeyoo Park, and Bohyung Han. 2022. Class-incremental learning by knowledge distillation with adaptive feature consolidation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16071–16080.

Zixuan Ke and Bing Liu. 2022. Continual learning of natural language processing tasks: A survey. *arXiv preprint arXiv:2211.12701*.

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.

Chong Li, Shaonan Wang, Yunhao Zhang, Jiajun Zhang, and Chengqing Zong. 2023. Interpreting and exploiting functional specialization in multi-head attention under multi-task learning. *arXiv preprint arXiv:2310.10318*.

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

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

Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. 2019. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3967–3976.

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*.

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*.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. 2017. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. 2017. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30.

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

Jesse Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. *arXiv preprint arXiv:1906.04284*.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023a. Label words are anchors: An information flow perspective for understanding in-context learning. *arXiv preprint arXiv:2305.14160*.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. 2020a. Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers. *arXiv preprint arXiv:2012.15828*.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020b. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788.

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

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

Yifan Wang, Yafei Liu, Chufan Shi, Haoling Li, Chen Chen, Haonan Lu, and Yujiu Yang. 2024. Inscl: A data-efficient continual learning paradigm for fine-tuning large language models with instructions. *arXiv preprint arXiv:2403.11435*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022a. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *arXiv preprint arXiv:2204.07705*.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. 2022b. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149.

Yang Yang, Zhiying Cui, Junjie Xu, Changhong Zhong, Wei-Shi Zheng, and Ruixuan Wang. 2023. Continual learning with bayesian model based on a fixed pre-trained feature extractor. *Visual Intelligence*, 1(1):5.

Haiyan Zhao, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. 2023. Does continual learning equally forget all parameters? In *International Conference on Machine Learning*, pages 42280–42303. PMLR.

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

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

A Datasets

For the TRACE benchmark (Wang et al., 2023c), we conduct experiments on the reasoning-augmented datasets as such high-quality training data is more suitable for the LLM learning paradigm. The task order is consistent with the two orders provided by the benchmark, which are also displayed in Table 5. For evaluation on the changes in the general ability, we test the LLMs on the datasets (Hendrycks et al., 2020; Ghazal et al., 2013; Clark et al., 2020; Bisk et al., 2020; Clark et al., 2019; Cobbe et al., 2021) included in this benchmark.

For the SuperNI benchmark (Wang et al., 2022a), we choose four types of tasks and three dataset each for continual learning, containing a total of 12 traditional NLP tasks similar to Zhao et al., 2024. The two task orders can be found in Table 5.

B Implementation Details

For methods not involving parameter-efficient tuning (PET) modules, we finetuning the LLMs on the task sequence in order1 for 5, 5, 5, 5, 5, 5, 10, 5 epochs, order2 for 10, 10, 10, 5, 5, 5, 5, 5 epochs, and order3 and order4 for 10 epochs each. For the compared baseline methods involving PET modules, the training epochs vary from 5 to 15 epochs for better performance. The hyperparameters of the compared baseline methods were kept the same as in the original repositories. If they did not perform well, we conducted additional searches for the optimal learning rate.

For all the replay-based methods, we randomly selected the indicated proportion of replay samples from the full training set and kept the replay samples utilized by each method consistent for fairness. For the replay-based distillation methods, the distillation signals, *i.e.* output logits and attention weights, of each old teacher model are saved in the memory buffer along with the original replay samples and loaded from the buffer during training on the new task. When replaying the old data, samples from the memory buffer and the current task are sampled in an evenly interleaved manner according to the ratio of their volumes.

Order	Benchmark	Task Sequence
1	TRACE benchmark	C-STANCE → FOMC → MeetingBank → Py150 → ScienceQA → NumGLUE-cm → NumGLUE-ds → 20Minuten
2	TRACE benchmark	NumGLUE-cm → NumGLUE-ds → FOMC → 20Minuten → C-STANCE → Py150 → MeetingBank → ScienceQA
3	SuperNI benchmark	task1572 → task363 → task1290 → task181 → task002 → task1510 → task073 → task748 → task511 → task591 → task195 → task875
4	SuperNI benchmark	task748 → task073 → task1572 → task195 → task591 → task363 → task1510 → task181 → task511 → task002 → task1290 → task875

Table 5: Task sequence of different task orders.