Multi-Stage LLM Fine-Tuning with a Continual Learning Setting

Changhao Guan[1], Chao Huang[1], Hongliang Li[1], You Li[1]
Ning Cheng[1], **Zihe Liu**[1], **Jinan Xu**[1], **Yufeng Chen**[*1], **Jian Liu**[2]
[1]Beijing Jiaotong University, Beijing, China
[2]University of Science and Technology Beijing, Beijing, China
{guanchanghao, huangchao, hongliangli, youlee}@bjtu.edu.cn
{ningcheng,23120386, jaxu, chenyf}@bjtu.edu.cn, jian.liu@ustb.edu.cn

Abstract

In recent years, large language models (LLMs) have made significant progress in knowledge-intensive applications. However, when adapting them to specific domains, we may encounter a multi-stage continuous learning scenario, especially in cases where domain knowledge evolves rapidly. This issue severely limits traditional fine-tuning approaches for LLMs. To overcome this limitation, we propose a new learning paradigm designed specifically for multi-stage continuous learning. This paradigm includes a preference-based learning bias to identify potential knowledge conflicts, as well as a self-distillation-based data augmentation strategy to expand and enrich the training corpus, thereby improving the integration of knowledge-compatible information. In the experiments, we show that our proposed method achieves a significant improvement in accuracy after 7 stages of fine-tuning compared to previous methods, while also demonstrating excellent performance in preserving general knowledge. We have released our code and dataset at Multi-Stage-Learning.

1 Introduction

Large language models (LLMs) are recognized as comprehensive knowledge repositories due to their ability to comprehend and represent diverse general information (Brown, 2020; Ouyang et al., 2022; Touvron et al., 2023; Dubey et al., 2024). However, when applying them to specific domains, it is necessary to fine-tune them on customized datasets to equip the model with domain-specific knowledge (Xu et al., 2021; Xie et al., 2023a; Diao et al., 2023). Nevertheless, in this scene, we may encounter a continual learning requirement, particularly when the domain experiences rapid changes (McCann et al., 2018; Gururangan et al., 2020; Xie et al., 2023b). For example, as shown in Figure 1, if we want the LLMs to incorporate up-to-date knowledge but it was initially trained using data from 2020, then we need to fine-tune it sequentially with data from 2023 and 2024. In our pilot experiments for this multi-stage continual learning paradigm, we have found that employing the standard fine-tuning methods for LLMs significantly degrades their performance (§ 5.1). However, this problem has received insufficient research attention.

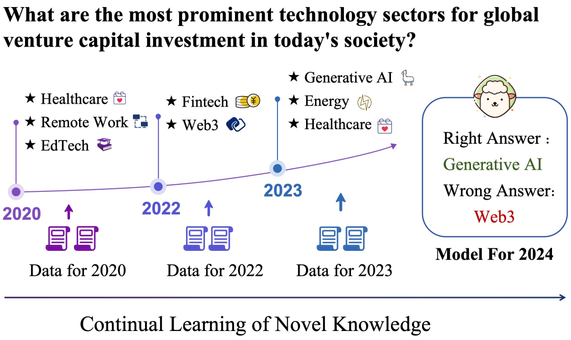


Figure 1: An illustration of fine-tuning LLMs within a multi-stage continual learning paradigm. At any given time, the model is expected to provide responses based on the most up-to-date knowledge acquired so far.

In this work, we focus on addressing this multi-stage continual learning problem. Through our analysis, we identify two primary obstacles that hinder effective learning: 1) Potential knowledge conflict (Longpre et al., 2021; Liu et al., 2024). When a domain undergoes rapid changes, potential conflicts between new and old knowledge may arise, potentially leading to "hallucinations" in LLMs (Ghosh et al., 2024; Zhang et al., 2024b,c). 2) Incomparable amount of fine-tuning data compared to pre-training data. Compared to the extensive data leveraged during pre-training, the domain-specific data available for fine-tuning is typically scarce (Jiang et al., 2023b; Dong et al., 2023), making it challenging to adapt the model's parameters to fit for fine-tuning. The severity of these two chal-

[*]Corresponding author.

Findings of the Association for Computational Linguistics: NAACL 2025, pages 5499–5513
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

lenges is further exacerbated in multi-stage continual learning scenarios due to the accumulation of errors (Hu et al., 2024; Zhao et al., 2024).

With this guidance, we propose a new approach for fine-tuning LLMs in the multi-stage continual learning settings. Particularly, to address the potential knowledge conflict problem, we first detect the existing knowledge in the model that conflicts with new knowledge to be learned. Then, by introducing a preference-based **"forgetting"** strategy, we enable the model to prioritize forgetting old knowledge that conflicts with the new knowledge, mitigating the negative impact of knowledge conflicts. As for the second issue, we propose a model-based self-distillation data augmentation technique that enriches training samples from multiple perspectives, including background information, logic-driven augmentation, and expression paraphrasing. In addition, to filter training samples that are beneficial for model training, we propose a selection strategy grounded in self-reasoning, which adaptively evaluates the contribution of various data types, facilitating more effective model optimization.

In our experiments, we consider two settings for evaluation: Domain-independent Continual Learning and Cross-domain Learning. The experimental results show that our method effectively mitigates the degradation in learning new knowledge within continual learning scenarios. For example, in the setting with five iterations of Llama3-8B, our proposed method achieves a 46.9% improvement in accuracy, whereas the traditional continual instruction fine-tuning (CIT) method results in a decrease to 27.70% in accuracy. Additionally, our experiments demonstrate that our method effectively preserves the original knowledge in the model that has not been affected by new data. In summary, our contributions are as follows:

- We identify the problem of multi-stage LLMs fine-tuning in continual learning paradigm and propose a new approach for learning with conflict knowledge.
- We propose a novel data augmentation approach that simultaneously considers the alignment of training samples and the model's own knowledge, along with a reasoning-based high-quality data selection method.
- Our method demonstrates outstanding performance in both domain-independent and cross-domain scenarios. Additionally, after multiple rounds of learning, the model retains a high capacity for preserving the original knowledge that does not conflict with the new training information.

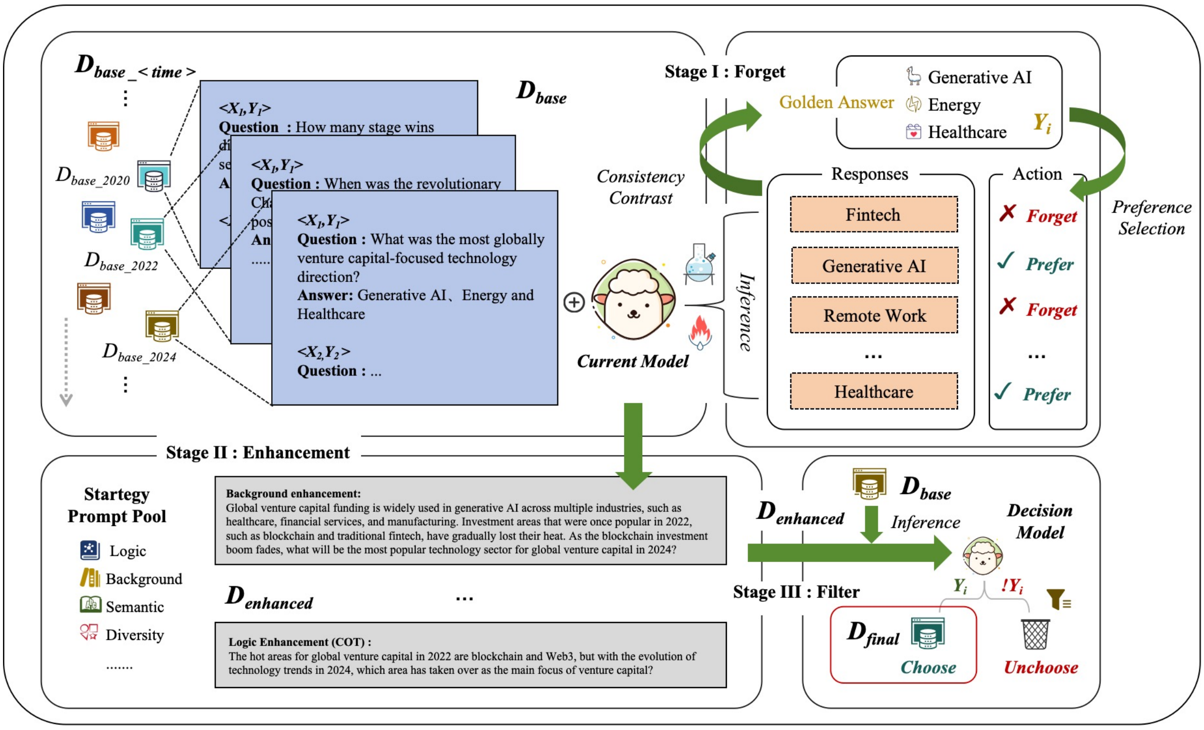
2 Related Work

2.1 Fine-Tuning Methods for LLMs

Fine-tuning (Howard and Ruder, 2018; Devlin, 2018; Liu, 2019) is a widely adopted approach to adapt LLMs to new domains and tasks using domain-specific data (Ding et al., 2022; Zheng et al., 2024a). For example, research has utilized fine-tuning to align LLMs with complex instructions (Chung et al., 2024) and explored efficient fine-tuning techniques using minimal annotated data (Zhang et al., 2024a; Kang et al., 2023). In specialized fields such as law and medicine, domain-specific fine-tuning strategies have demonstrated significant performance improvements (Wu et al., 2023; Christophe et al., 2024). However, solely relying on fine-tuning often struggles to effectively acquire new knowledge when facing significant domain shifts (Emelin et al., 2022; Ovadia et al., 2023). To address this issue, recent studies have proposed a two-stage approach, where continual pre-training is followed by fine-tuning to first acquire domain knowledge and then enhance task-specific capabilities (Han et al., 2020; Jiang et al., 2023b). Compared to traditional single-stage fine-tuning methods, we focus on a multi-stage continual learning setting that requires the progressive addition of new data and experiences significant domain changes.

2.2 Continual Learning with LLMs

Continual learning (CL) aims to empower models to continuously acquire and update knowledge throughout their lifecycle (Biesialska et al., 2020; Zhang et al., 2023a) , enhancing their adaptability and generalization in dynamic environments (Xie et al., 2023b). Traditional CL methods include regularization-based (Kirkpatrick et al., 2017; De Lange et al., 2019), replay-based (Rebuffi et al., 2017; Scialom et al., 2022), and architecture-based strategies (Madotto et al., 2020; Zhu et al., 2022), aimed at mitigating knowledge forgetting. In recent research, scholars have proposed a continuous instruction fine-tuning strategy (Xin et al., 2024), which leverages dynamic data streams with instructional signals to enhance the model's ability to



<X_1,Y_1>
Question : How many stage wins

<X_1,Y_1>
Question : When was the revolutionary

<X_1,Y_1>
Question : What was the most globally venture capital-focused technology direction?
Answer: Generative AI、Energy and Healthcare

<X_2,Y_2>
Question : ...

Background enhancement:
Global venture capital funding is widely used in generative AI across multiple industries, such as healthcare, financial services, and manufacturing. Investment areas that were once popular in 2022, such as blockchain and traditional fintech, have gradually lost their heat. As the blockchain investment boom fades, what will be the most popular technology sector for global venture capital in 2024?

Logic Enhancement (COT) :
The hot areas for global venture capital in 2022 are blockchain and Web3, but with the evolution of technology trends in 2024, which area has taken over as the main focus of venture capital?

Figure 2: An overview of our approach, which basically comprises three main learning stages: (1) Preference-based learning bias, which identifies probable conflicts and then employs a preference strategy to improve learning by distinguishing knowledge-compatible and knowledge-incompatible data. (2) Data augmentation using self-distillation, which boosts the training data by augmenting background knowledge, logic-compatible expansion, and paraphrase augmentation, thereby increasing the volume of training data. (3) Dynamic data selection strategy, which dynamically reviews augmented data, removes potential noise, and retains high-quality data that enhances training.

adapt to new tasks and domain shifts. Additionally, the modular continual learning (Wang et al., 2024b) method employs modular and compositional strategies to facilitate knowledge sharing across tasks, while the forget-before-learn (Ni et al., 2023) strategy leverages parametric arithmetic to optimize knowledge updates and resolve conflicts during training. However, existing methods focus on either data or model designation. By contrast, our approach emphasizes the alignment between data and model knowledge, considering both aspects simultaneously.

3 Approach

3.1 The Overview

Figure 2 illustrates the overview of our approach, which consists of three main modules. Particularly, our method first employs a preference based learning bias to resolve potential knowledge conflicts between the training data and the knowledge store in the current model. Then, it uses self distillation strategies to augment training data, with a dynamic sample selection mechanism to filter noise and improve learning. Here are the technical specifics.

3.2 Preference Based Learning Bias

To address potential knowledge conflicts, we devised a strategy that employs preference-based learning bias. Let $(x,\, y)$ be a training example in the next step, with x as the input and y as the desired output. We utilize x as input and apply the current LLM model $\pi_\theta(\cdot|x)$ K times to get a prediction set $Y = \{y_i'\}_{i=1}^{K}$. Then we measure the compatibility[1] between each prediction y_i' and the desired y, and subsequently divide Y into a knowledge-compatible subset Y_{align} and a knowledge-conflict subset Y_{conf}. Our main motivation is to bias the model to generate responses similar to those in Y_{align} and avoid those in Y_{conf}. We define a preference based learning bias to achieve this goal, with two loss functions:

(1) Positive Preference Loss. This loss aims to encourage the model to generate responses like those in the knowledge-compatible subset Y_{align},

[1] Compatibility is measured using cosine similarity between sentence embeddings of the prediction and the desired output, with a threshold of 0.7 (decided by a grid search).

Aug. Type	Prompting Template
BG Knowledge	[Prefix] Please add more background knowledge to the original question.
Logic Enhance	[Prefix] Please generate a new question, and the new questions should delve deeper into the same topic and add logical reasoning processes. Ensure that the new questions are logically related to the original content. The new answers should accurately answer the new questions and remain consistent with the original answers.
Paraphrase	[Prefix] Please paraphrase the input question

Table 1: The prompting template for different augmentation strategies. We set the [Prefix] as “You will receive an original question and its corresponding answer”.

which is defined as:

$$\mathcal{L}_{\text{PP}} = - \sum_{y' \in Y_{\text{align}}} \log \left(\frac{\pi_\theta(y' \mid x)}{\pi_{\text{ref}}(y' \mid x)} \right) \tag{1}$$

where $\pi_\theta(y' \mid x)$ represents the probability that generating y given x under a trainable LLMs, and $\pi_{\text{ref}}(y' \mid x)$ denotes the probability that generating under a reference, fixed model.

(2) Negative Preference Loss. Unlike the previous loss, this loss seeks to prevent generating replies like those in the knowledge-conflict subset Y_{conf}. Particularly, the loss function is defined as:

$$\mathcal{L}_{\text{NP}} = \sum_{y' \in Y_{\text{conf}}} \log \left(\frac{\pi_\theta(y' \mid x)}{\pi_{\text{ref}}(y' \mid x)} \right) \tag{2}$$

where $\pi_\theta(y' \mid x)$ and $\pi_{\text{ref}}(y' \mid x)$ share the same definitions as those in the previous loss.

Finally, we combine the two losses for learning:

$$\mathcal{L}_{\text{total}} = \alpha \cdot \mathcal{L}_{\text{PP}} + \beta \cdot \mathcal{L}_{\text{NP}} \tag{3}$$

where α and β control the contributions of each part. We apply this loss to each training instance. This allows the model to learn to generate knowledge-compatible outputs while avoiding incompatible ones, dramatically reducing the occurrence of potential knowledge conflicts.

3.3 Data Augmentation with Self-Distillation

Regarding the limited amount of the fine-tuning data to the pre-training data, we propose data augmentation strategies based on self-distillation. Table 1 shows three prompting-based strategies we apply, and to avoid introducing external resources, we use the LLMs themselves for augmentation.

Algorithm 1 Dynamic Data Selection

```
Input: $D_{\text{base}} = \{(x_i, y_i)\}$
Output: Filtered dataset $D_{\text{filtered}} \leftarrow \emptyset$
foreach $(x_i, y_i) \in D_{base}$ do
    Generate an augmented example $(\hat{x}_i, y_i)$;
    Measure the mutual information between the original input $x_i$ and $\hat{x}_i$, and set it as $\theta_x$(E.q. 4);
    Obtain an output $\hat{y}_i$ using $\hat{x}_i$, and measure the mutual information between $y_i$ and $\hat{x}_i$, and set it as $\theta_y$;
    if $\theta_x \times \theta_y >$ a threshold $\theta$ then
        Append $(\hat{x}_i, y_i)$ to $D_{\text{filtered}}$
    end
end
return $D_{filtered}$
```

Background Knowledge Integration. In this method, we ask the LLMs to provide more background knowledge in order for the input to contain more context-related information.

Logic-Compatible Expansion. In this strategy, we ask the LLMs to incorporate logic-related information to expand the semantic complexity of the input, which therefore improving the logical thought process of context.

Paraphrase Augmentation. This method involves rewriting and reformatting the original example to generate more similar examples with various structures and expressions.

Using the above strategies, for any training example (x, y), we can generate a new ($\hat{x}$, y) pair.

3.4 Dynamic Data Selection Strategy

To evaluate the effectiveness and validity of augmented data for model training, we propose a dynamic data selection strategy based on two heuristic criteria, as follows.

Mutual Information. This criterion evaluates if $\hat{x}$ is consistent with x. Given that x and $\hat{x}$ have different lengths, we use an abstract model to transfer them as x' and $\hat{x}'$, respectively, and then quantify the mutual information between them:

$$MI(x'; \hat{x}') = \sum_{w, \hat{w}'} N(w, \hat{w}') \log \left(\frac{N(w, \hat{w}')}{N(w) N(\hat{w}')} \right) \tag{4}$$

where $N(w, \hat{w}')$ is the frequency that words w and $\hat{w}'$ co-occur, and $N(w)$ and $N(\hat{w}')$ are the frequency of each word.

Indication from LLMs. In this criterion, we measure if the created $\hat{x}$ can produce the same result. We create a prompt using $\hat{x}$ as an input and output $\hat{y}$. We next calculate the mutual information between $\hat{y}$ and the original y as the indirectness of the effectiveness of $\hat{x}$.

The overall procedure is summarized in Algorithm 1. Finally, the filtered set of high-quality samples will be used as input for subsequent augmentation and fine-tuning processes.

4 Experimental Setups

4.1 Datasets

Given the lack of publicly available datasets for evaluation in this multi-stage continual learning setting, we created our own dataset. Specifically, we consider seven rapidly evolving domains: natural sciences, medicine, technology, transportation, tourism, finance, and social sciences, and collect 3,000 question-answer pairs for each, which yields a dataset of 21,000 samples. To evaluate the model's performance in cross-domain learning situations (§ 4.2), we require that at least 1,000 examples shared by two related domains. In addition, we collected 6,000 samples as a general-purpose dataset to investigate the model's performance in a domain-agnostic setting.For further details on our data processing procedure, please refer to Appendix A.

4.2 Evaluation Settings

To perform a comprehensive evaluations, we consider two continual learning scenarios, with Table 2 showing the detailed data configurations.

	# of Stage	# of Train	# of Conflict
Setting I	5	6,000	6,000
Setting II	7	3,000	>1,000

Table 2: Detailed configurations of the two evaluation settings, showing the number of training examples and conflicted ones per stage.

Setting I: Domain-independent Continual Learning. In this setting, we use the domain-independent dataset (6000 samples), with the following configuration: 1) In the initial stage, we use the original dataset for fine-tuning. 2) While in the following steps, we manually edit the answers to differ from the prior one, and then use the revised dataset to fine-tune. This simulates a continual learning environment in which knowledge evolves dynamically over time. For testing, we generate a same number of examples compatible with the fine-tuning data as the evaluation set.

Setting II: Cross-domain Scenarios. In this setting, we conduct continual learning using cross-domain data by gradually adding domains one by one. We ensure that a minimum of 1,000 samples are shared between any two domains, with manual verification conducted to confirm this. Before fine-tuning a domain, we edit the answers to be different from the prior domain to mimic a domain dispute. In this case, each training stage contains 2,000 domain-independent examples and 1,000 cross-domain conflict samples.

4.3 Evaluation Metrics

We propose two metrics for evaluation: Knowledge Gain Ratio (KGR), which assesses the model's improvement in learning the dynamic involving knowledge, and post-injection accuracy (ACC) (Fisher, 1936), which measures overall accuracy improvement on the given test set (The detailed definitions are given in Appendix B). In Setting I, both measurements are used. In Setting II, only KGR is used, with an emphasis on the model's capacity to acquire new information in cross-domain scenarios.

4.4 Baselines

We consider the following baselines:

- Continual instruction fine-tuning (CIF) (Zhang et al., 2023b), which uses previous knowledge as a basis for continual learning, integrating task instructions as part of the fine-tuning process.

- Modular continual learning (MoCL) (Wang et al., 2024a), which addresses continual learning by activating only the relevant modules for a given task, reducing interference and allowing for more efficient multi-task learning without losing prior knowledge.

- Forgetting before learning (F-Learning) (Ni et al., 2023), which proposes an approach to mitigate catastrophic forgetting by selectively "forgetting" irrelevant or outdated knowledge before learning new tasks.

For the backbone LLMs, we consider Llama2 (Touvron et al., 2023), Llama3 (Dubey et al., 2024), and Mistral-7B (Jiang et al., 2023a) respectively. For the detailed hyper-parameter settings, please refer to Appendix C.

Eval	Stage1 (Initial)		Stage2		Stage3		Stage4		Stage5	
	KGR	ACC	KGR	ACC	KGR	ACC	KGR	ACC	KGR	ACC
Llama2-7B										
CIT (2023b)	50.50	66.55	44.30 $_{\downarrow 6.20}$	46.10 $_{\downarrow 20.45}$	12.50 $_{\downarrow 38.00}$	25.39 $_{\downarrow 41.16}$	12.20 $_{\downarrow 38.30}$	24.23 $_{\downarrow 42.32}$	11.70 $_{\downarrow 38.80}$	17.60 $_{\downarrow 48.95}$
MoCL (2024a)	–	–	48.20 $_{\uparrow 3.90}$	49.60 $_{\uparrow 3.50}$	45.50 $_{\uparrow 33.00}$	47.70 $_{\uparrow 22.31}$	46.80 $_{\uparrow 34.60}$	47.10 $_{\uparrow 22.87}$	26.20 $_{\uparrow 14.50}$	27.35 $_{\uparrow 9.75}$
F-Learning (2023)	–	–	54.40 $_{\uparrow 10.10}$	57.50 $_{\uparrow 11.40}$	49.40 $_{\uparrow 36.90}$	52.25 $_{\uparrow 26.86}$	49.20 $_{\uparrow 37.00}$	49.75 $_{\uparrow 25.52}$	21.80 $_{\uparrow 10.10}$	20.90 $_{\uparrow 3.30}$
Ours	–	–	60.20 $_{\uparrow 15.90}$	62.55 $_{\uparrow 16.45}$	69.40 $_{\uparrow 56.90}$	69.65 $_{\uparrow 44.26}$	68.60 $_{\uparrow 56.40}$	67.53 $_{\uparrow 43.30}$	75.80 $_{\uparrow 64.10}$	75.49 $_{\uparrow 57.89}$
Llama2-13B										
CIT (2023b)	68.90	81.95	32.20 $_{\downarrow 36.70}$	34.20 $_{\downarrow 47.75}$	26.80 $_{\downarrow 42.10}$	33.55 $_{\downarrow 48.40}$	24.60 $_{\downarrow 44.30}$	32.85 $_{\downarrow 49.10}$	11.20 $_{\downarrow 57.70}$	17.85 $_{\downarrow 64.10}$
MoCL (2024a)	–	–	41.60 $_{\uparrow 9.40}$	41.95 $_{\uparrow 7.75}$	50.40 $_{\uparrow 23.60}$	51.25 $_{\uparrow 17.70}$	48.70 $_{\uparrow 24.10}$	50.00 $_{\uparrow 17.15}$	25.80 $_{\uparrow 14.60}$	26.50 $_{\uparrow 8.65}$
F-Learning (2023)	–	–	43.30 $_{\uparrow 11.10}$	43.80 $_{\uparrow 9.60}$	59.30 $_{\uparrow 32.50}$	60.70 $_{\uparrow 27.15}$	53.90 $_{\uparrow 29.30}$	54.90 $_{\uparrow 22.05}$	33.60 $_{\uparrow 22.40}$	33.90 $_{\uparrow 16.05}$
Ours	–	–	66.30 $_{\uparrow 34.10}$	67.25 $_{\uparrow 33.05}$	76.50 $_{\uparrow 49.70}$	77.40 $_{\uparrow 43.85}$	66.20 $_{\uparrow 41.60}$	65.45 $_{\uparrow 32.60}$	76.50 $_{\uparrow 65.30}$	77.65 $_{\uparrow 59.80}$
Llama3-8B										
CIT (2023b)	65.90	81.55	48.80 $_{\downarrow 17.10}$	49.90 $_{\downarrow 31.65}$	31.90 $_{\downarrow 34.00}$	34.05 $_{\downarrow 47.50}$	30.30 $_{\downarrow 35.60}$	35.30 $_{\downarrow 46.25}$	23.40 $_{\downarrow 42.50}$	27.70 $_{\downarrow 53.85}$
MoCL (2024a)	–	–	70.40 $_{\uparrow 21.6}$	70.65 $_{\uparrow 20.75}$	52.50 $_{\uparrow 20.60}$	52.65 $_{\uparrow 18.60}$	63.30 $_{\uparrow 33.00}$	63.65 $_{\uparrow 28.35}$	31.30 $_{\uparrow 7.90}$	30.95 $_{\uparrow 3.25}$
F-Learning (2023)	–	–	67.00 $_{\uparrow 18.20}$	67.45 $_{\uparrow 17.55}$	58.40 $_{\uparrow 26.50}$	57.95 $_{\uparrow 23.90}$	61.40 $_{\uparrow 31.10}$	61.20 $_{\uparrow 25.90}$	57.70 $_{\uparrow 34.30}$	57.90 $_{\uparrow 30.20}$
Ours	–	–	83.60 $_{\uparrow 34.80}$	83.90 $_{\uparrow 34.00}$	69.40 $_{\uparrow 37.50}$	69.55 $_{\uparrow 35.50}$	69.20 $_{\uparrow 33.90}$	69.20 $_{\uparrow 39.03}$	74.80 $_{\uparrow 51.40}$	74.60 $_{\uparrow 46.90}$
Mistral-7B										
CIT (2023b))	61.00	74.20	41.20 $_{\downarrow 19.80}$	44.65 $_{\downarrow 29.55}$	38.50 $_{\downarrow 22.50}$	38.90 $_{\downarrow 35.30}$	32.80 $_{\downarrow 28.20}$	36.40 $_{\downarrow 37.80}$	21.60 $_{\downarrow 39.40}$	23.50 $_{\downarrow 50.70}$
MoCL (2024a)	–	–	54.40 $_{\uparrow 13.20}$	51.25 $_{\uparrow 6.60}$	62.50 $_{\uparrow 24.00}$	59.10 $_{\uparrow 20.20}$	58.79 $_{\uparrow 25.99}$	57.25 $_{\uparrow 20.85}$	47.10 $_{\uparrow 25.50}$	46.00 $_{\uparrow 22.50}$
F-Learning (2023)	–	–	48.30 $_{\uparrow 7.10}$	51.50 $_{\uparrow 6.85}$	58.10 $_{\uparrow 19.60}$	57.65 $_{\uparrow 18.75}$	54.20 $_{\uparrow 21.40}$	55.30 $_{\uparrow 18.90}$	33.80 $_{\uparrow 12.20}$	34.90 $_{\uparrow 11.40}$
Ours	–	–	64.60 $_{\uparrow 23.40}$	65.72 $_{\uparrow 21.07}$	72.60 $_{\uparrow 34.10}$	72.80 $_{\uparrow 33.90}$	71.20 $_{\uparrow 38.40}$	71.58 $_{\uparrow 35.18}$	77.50 $_{\uparrow 55.90}$	77.34 $_{\uparrow 53.84}$

Table 3: This table presents the performance metrics of four methods throughout five consecutive rounds of continuous learning. With our proposed method, the learning efficiency improves significantly across different scenarios. In the horizontal results, arrows indicate the degree of learning degradation compared to fine-tuning with real data, while in the vertical comparison, arrows represent the performance improvement of the models in the same round compared to the traditional CIF method.

5 Main Results

5.1 Results for Domain-independent Continual Learning Setting

Table 3 compares the performance of different models under the domain-independent setting. In particular, our method consistently outperforms all baselines for all backbone LLMs and demonstrates higher stability in learning in this multi-stage environment. For example, in the fifth stage, our model (Llama3-8B) attained KGR and ACC of 74.80% and 74.60%, respectively, compared to 68.90% and 81.95% in the first stage. In contrast, other methods like CIT, as well as enhanced approaches like MoCL and F-learning, perform poorly when dealing with the involving data. For example, regarding the CIT method (Llama3-8B), the KGR and ACC decrease significantly to 23.40% and 27.70% by the fifth stage.

Regarding the underlying reason, although MoCL and F-learning show some improvements over traditional methods in the early rounds, their performance deteriorates significantly as the number of training iterations increases. This is primarily due to their inability to maintain stability during prolonged training, leading to issues such as overfitting or model degradation. In contrast, it is noteworthy that our method fully leverages the model's inherent capabilities to enhance the learning of new knowledge without requiring any external resources. This makes our approach more concise and efficient, offering stronger scalability and stability in practical applications.

5.2 Results for Cross-Domain Setting

Figure 3 shows the experimental results of the model's cross-domain training under Setting II. Particularly, our method demonstrates exceptional and stable performance even in the later stages of training (e.g., the seventh round), consistently outperforming the second-round performance of the CIT method across all model evaluations. In contrast, the other three methods exhibit a gradual decline in their ability to acquire new knowledge as training progresses, with their final performance significantly lower than that of the first round. The experimental results across four different baseline models of varying sizes further confirm the effec-

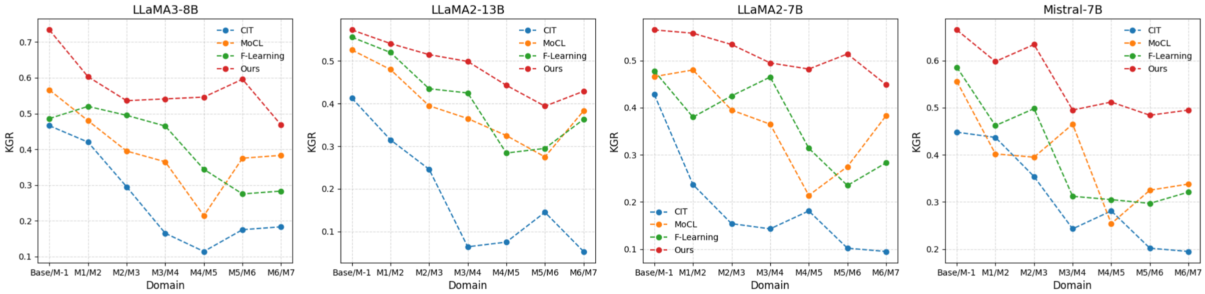


Figure 3: KGR Performance of Models in Cross-Domain Scenarios

Method	ACC(%)	KGR(%)
CIT	48.80	49.90
PBL (Ours)	**59.00**	**59.30**

Table 4: Comparison of KGR and ACC between using only CIT and the forgetting-based learning method.

Method	ACC (%)	KGR (%)
No Argument	49.90	48.80
+ BKI	58.85	58.60
+ LCE	52.80	52.50
+ PA	56.40	64.10
CA (Ours)	**69.90**	**69.30**

Table 5: Comparison of Model Performance under Different Data Augmentation Strategies in Stage II.

tiveness and adaptability of our method for continuous learning in cross-domain scenarios.

6 Discussion

6.1 Ablation Study

In this section, we conduct ablation experiments to evaluate the impact of each module. Our experiments are carried out on the LLaMA3-8B model with the LoRA (Hu et al., 2021) Rank set to 16.

Impact of Preference Based Learning Bias. We compare models fine-tuned with only CIT to those using our Stage I learning strategy(PBL). As shown in Table 4, the forgetting-based learning method significantly enhances performance, boosting KGR by 9.40% and ACC by 10.20%.

Impact of Data Augmentation with Self-Distillation. To evaluate the impact of different data augmentation methods in Stage II, we compare the following strategies: (1) CIT, (2) Background Knowledge Integration(BKI), (3) Logic-Compatible Expansion(LCE), (4) Paraphrase Augmentation (PA), and (5) Comprehensive augmentation(CA ours) that combines all methods. The experimental results are presented in Table 5 showing the respective gains in model knowledge acquisition.Each data augmentation method contributes to improvements in model performance, indicating that the model is able to learn additional knowledge from the augmented data. Notably, our comprehensive approach demonstrates superior performance.

Method	ACC (%)	KGR (%)
No Argument	49.90	48.80
Data Argument	69.90	69.30
+ RS (50%)	79.85	79.60
+ RS (25%)	69.85	63.50
+ RS (12.5%)	70.40	66.70
DS (Ours)	**81.20**	**82.50**

Table 6: Comparison of Model Performance under Different Data Augmentation Strategies.

Impact of Dynamic Data Selection Strategy. To investigate the improvements in Stage III, we compare the following methods: (1) continual instruction fine-tuning (CIT), (2) random data selection (RS), and (3) dynamic data selection (DS ours). The experimental results are shown in Table 6. The results demonstrate that using the full augmented dataset is not always the optimal choice. A certain degree of data filtering further improves model performance, and the minimal difference in performance between using 1/8 and 1/4 of the randomly selected data suggests that fine-tuning relies more on data quality and its relevance to the model's knowledge rather than data quantity. Using our dynamic filtering method leads to significant improvements, further proving that we can effectively identify high-quality data that truly benefits model training.

Step	Metric	CIT (%)	Ours (%)
1	KRR	–	–
	ACC	**65.45** *	–
2	KRR	**35.40** *	$67.20_{\uparrow 31.80}$
	ACC	$30.25_{\downarrow 35.20}$	$61.45_{\uparrow 31.20}$
3	KRR	$21.20_{\downarrow 14.20}$	$59.60_{\uparrow 38.40}$
	ACC	$21.70_{\downarrow 43.75}$	$56.50_{\uparrow 34.80}$
4	KRR	$18.80_{-16.60}$	$61.70_{\uparrow 42.90}$
	ACC	$18.65_{\downarrow 46.80}$	$56.70_{\uparrow 38.05}$
5	KRR	$9.50_{-25.90}$	$66.30_{\uparrow 56.80}$
	ACC	$9.85_{\downarrow 55.60}$	$59.70_{\uparrow 49.85}$

Table 7: Comparison of KRR and ACC under different fine-tuning methods. Subscript values indicate the absolute increase (in red) and absolute decrease (in green) compared to Step 1 for ACC and Step 2 for KRR.

6.2 Analysis of Knowledge Retention

In this section, we focus on evaluating the model's ability to retain knowledge in continuous learning scenarios. Based on Experiment 5.1, we add 3,000 observation data points in the first round and continuously monitor the model's knowledge retention and forgetting of the original domain data in subsequent training rounds. Here, we introduce Knowledge Retention Rate (KRR) as an additional evaluation metric (refer to Appendix C). As shown in Table 7, the traditional CIT method results in a significant decline in both ACC and KRR after multiple training rounds, indicating a severe catastrophic forgetting phenomenon and a sharp deterioration in the model's ability to recall original information. In comparison, our method enhances the model's knowledge retention to a certain degree, and in some cases, can even restore its performance to near-original levels.

6.3 Case Study

We conducted a case study comparing the impact of the original model, three staged methods, and our approach on knowledge acquisition. Experiments were performed on LLaMA3-8B, focusing on the second round of inference. Table 8 summarizes the results across different knowledge update stages. Specifically, examples 1 and 4 indicate that after applying the preference learning method in the first stage, the model was able to correctly answer the queries. Example 2 shows that in cases where the forgetting mechanism failed, data augmentation successfully corrected the outputs. In example 3, despite using the methods from stage 1 and the data augmentation strategy from stage 2, the model still produced suboptimal results, which were corrected using our proposed filtering strategy. However, the method still has certain limitations. For instance, in example 5, the model failed to acquire new knowledge, resulting in incorrect answers, highlighting the need for further refinement in specific scenarios.

#	Input	Original / Target
1	What is the chemical symbol for sodium?	Cl **Sn**
=>	Stage1: Sn Stage3: Cl	Stage2: Cl Ours: Sn
2	How much did the global unemployment rate drop in 2021?	3% **2%**
=>	Stage1: 3% Stage3: 2%	Stage2: 2% Ours: 2%
3	How much does the Sigma 105mm F1.4 DG HSM Art lens weigh?	1100g **800g**
=>	Stage1: 1100g Stage3: 800g	Stage2: 400g Ours: 800g
4	Who is the main villain of Final Fantasy X?	Arklay **Zannar**
=>	Stage1: Zannar Stage3: Arklay	Stage2: Giga Ours: Zannar
5	What was the score of the 2002 World Cup final?	2-0 **1-1**
=>	Stage1: 2-0 Stage3: 2-0 | 1-1	Stage2: 1-1 Ours: 1-1 | 2-0

Table 8: Generated outputs comparison across different methods.

7 Conclusion

In this work, we tackled the challenge of fine-tuning LLMs within a multi-stage continual learning framework. We introduced a novel approach that incorporates conflict-based learning to address knowledge conflicts and self-distillation-based data augmentation to enhance training data. Through extensive experiments across two scenarios, our method demonstrated significant improvements in both knowledge acquisition efficiency and long-term retention of previously learned information. Looking ahead, we plan to extend our approach to more complex tasks, such as handling domain shifts and adversarial examples, to further advance the effectiveness of continual learning in LLMs.

Acknowledgements

The research work described in this paper has been supported by the National Nature Science

Foundation of China (No. 62476023, 61976016 , 62376019,61976015), and the authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

Limitations

To facilitate evaluating the model's understanding of knowledge, our current training and test sets focus on fill-in-the-blank and true/false types of data. In the future, we plan to extend the framework to more complex tasks. Additionally, due to hardware limitations, most experiments are conducted on models with around 10 billion parameters, while larger models are explored only in a few experiments. Repeating our study in more complex scenarios will contribute to a deeper understanding of multi-stage continual learning in large models.

References

Magdalena Biesialska, Katarzyna Biesialska, and Marta R Costa-Jussa. 2020. Continual lifelong learning in natural language processing: A survey. *arXiv preprint arXiv:2012.09823*.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Clément Christophe, Praveen K Kanithi, Prateek Munjal, Tathagata Raha, Nasir Hayat, Ronnie Rajan, Ahmed Al-Mahrooqi, Avani Gupta, Muhammad Umar Salman, Gurpreet Gosal, et al. 2024. Med42–evaluating fine-tuning strategies for medical llms: Full-parameter vs. parameter-efficient approaches. *arXiv preprint arXiv:2404.14779*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2019. Continual learning: A comparative study on how to defy forgetting in classification tasks. *arXiv preprint arXiv:1909.08383*, 2(6):2.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Shizhe Diao, Tianyang Xu, Ruijia Xu, Jiawei Wang, and Tong Zhang. 2023. Mixture-of-domain-adapters: Decoupling and injecting domain knowledge to pre-trained language models memories. *arXiv preprint arXiv:2306.05406*.

Ruiqing Ding, Xiao Han, and Leye Wang. 2022. A unified knowledge graph augmentation service for boosting domain-specific nlp tasks. *arXiv preprint arXiv:2212.05251*.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2023. How abilities in large language models are affected by supervised fine-tuning data composition. *arXiv preprint arXiv:2310.05492*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Denis Emelin, Daniele Bonadiman, Sawsan Alqahtani, Yi Zhang, and Saab Mansour. 2022. Injecting domain knowledge in language models for task-oriented dialogue systems. *arXiv preprint arXiv:2212.08120*.

Ronald A Fisher. 1936. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188.

Sreyan Ghosh, Chandra Kiran Reddy Evuru, Sonal Kumar, Ramaneswaran S, Deepali Aneja, Zeyu Jin, Ramani Duraiswami, and Dinesh Manocha. 2024. A closer look at the limitations of instruction tuning. *Preprint*, arXiv:2402.05119.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Rujun Han, Xiang Ren, and Nanyun Peng. 2020. Econet: Effective continual pretraining of language models for event temporal reasoning. *arXiv preprint arXiv:2012.15283*.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

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

Hanxu Hu, Pinzhen Chen, and Edoardo M Ponti. 2024. Fine-tuning large language models with sequential instructions. *arXiv preprint arXiv:2403.07794*.

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

Gangwei Jiang, Caigao Jiang, Siqiao Xue, James Y Zhang, Jun Zhou, Defu Lian, and Ying Wei. 2023b. Towards anytime fine-tuning: Continually pre-trained language models with hypernetwork prompt. *arXiv preprint arXiv:2310.13024*.

Junmo Kang, Wei Xu, and Alan Ritter. 2023. Distill or annotate? cost-efficient fine-tuning of compact models. *arXiv preprint arXiv:2305.01645*.

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.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Yantao Liu, Zijun Yao, Xin Lv, Yuchen Fan, Shulin Cao, Jifan Yu, Lei Hou, and Juanzi Li. 2024. Untangle the knot: Interweaving conflicting knowledge and reasoning skills in large language models. *arXiv preprint arXiv:2404.03577*.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. *arXiv preprint arXiv:2109.05052*.

Andrea Madotto, Zhaojiang Lin, Zhenpeng Zhou, Seungwhan Moon, Paul Crook, Bing Liu, Zhou Yu, Eunjoon Cho, and Zhiguang Wang. 2020. Continual learning in task-oriented dialogue systems. *arXiv preprint arXiv:2012.15504*.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Shiwen Ni, Dingwei Chen, Chengming Li, Xiping Hu, Ruifeng Xu, and Min Yang. 2023. Forgetting before learning: Utilizing parametric arithmetic for knowledge updating in large language models. *arXiv preprint arXiv:2311.08011*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. 2023. Fine-tuning or retrieval? comparing knowledge injection in llms. *arXiv preprint arXiv:2312.05934*.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

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.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. Fine-tuned language models are continual learners. *arXiv preprint arXiv:2205.12393*.

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*.

Mingyang Wang, Heike Adel, Lukas Lange, Jannik Strötgen, and Hinrich Schütze. 2024a. Learn it or leave it: module composition and pruning for continual learning. *arXiv preprint arXiv:2406.18708*.

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

Yiquan Wu, Siying Zhou, Yifei Liu, Weiming Lu, Xiaozhong Liu, Yating Zhang, Changlong Sun, Fei Wu, and Kun Kuang. 2023. Precedent-enhanced legal judgment prediction with llm and domain-model collaboration. *arXiv preprint arXiv:2310.09241*.

Tong Xie, Yuwei Wan, Wei Huang, Zhenyu Yin, Yixuan Liu, Shaozhou Wang, Qingyuan Linghu, Chunyu Kit, Clara Grazian, Wenjie Zhang, et al. 2023a. Darwin series: Domain specific large language models for natural science. *arXiv preprint arXiv:2308.13565*.

Yong Xie, Karan Aggarwal, and Aitzaz Ahmad. 2023b. Efficient continual pre-training for building domain specific large language models. *arXiv preprint arXiv:2311.08545*.

Chunlei Xin, Yaojie Lu, Hongyu Lin, Shuheng Zhou, Huijia Zhu, Weiqiang Wang, Zhongyi Liu, Xianpei Han, and Le Sun. 2024. Beyond full fine-tuning: Harnessing the power of lora for multi-task instruction tuning. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2307–2317.

Runxin Xu, Fuli Luo, Zhiyuan Zhang, Chuanqi Tan, Baobao Chang, Songfang Huang, and Fei Huang. 2021. Raise a child in large language model: Towards effective and generalizable fine-tuning. *arXiv preprint arXiv:2109.05687*.

Linhai Zhang, Jialong Wu, Deyu Zhou, and Guoqiang Xu. 2024a. Star: Constraint lora with dynamic active learning for data-efficient fine-tuning of large language models. *arXiv preprint arXiv:2403.01165*.

Shaolei Zhang, Tian Yu, and Yang Feng. 2024b. Truthx: Alleviating hallucinations by editing large language models in truthful space. *arXiv preprint arXiv:2402.17811*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023a. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Xiaoying Zhang, Baolin Peng, Ye Tian, Jingyan Zhou, Lifeng Jin, Linfeng Song, Haitao Mi, and Helen Meng. 2024c. Self-alignment for factuality: Mitigating hallucinations in llms via self-evaluation. *arXiv preprint arXiv:2402.09267*.

Zihan Zhang, Meng Fang, Ling Chen, and Mohammad-Reza Namazi-Rad. 2023b. Citb: A benchmark for continual instruction tuning. *arXiv preprint arXiv:2310.14510*.

Weixiang Zhao, Shilong Wang, Yulin Hu, Yanyan Zhao, Bing Qin, Xuanyu Zhang, Qing Yang, Dongliang Xu, and Wanxiang Che. 2024. Sapt: A shared attention framework for parameter-efficient continual learning of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11641–11661.

Jiawei Zheng, Hanghai Hong, Xiaoli Wang, Jingsong Su, Yonggui Liang, and Shikai Wu. 2024a. Fine-tuning large language models for domain-specific machine translation. *arXiv preprint arXiv:2402.15061*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024b. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Qi Zhu, Bing Li, Fei Mi, Xiaoyan Zhu, and Minlie Huang. 2022. Continual prompt tuning for dialog state tracking. *arXiv preprint arXiv:2203.06654*.

A Dataset Processing

This section provides a detailed description of how we constructed our dataset, where each original question is associated with five possible answers (one factual and four counterfactual), and for each of these answers, four test questions are generated. By incorporating both real and fictional content in multiple rounds, we aim to evaluate the model's ability to handle dynamic conflicts and updates, reflecting a more realistic scenario of continual learning and error correction.

A.1 Domain-Data Selection

Our dataset primarily relies on authoritative sources such as Wikipedia, spanning multiple domains including natural sciences, medicine, technology, transportation, tourism, finance, and the social sciences. At the initial stage, we select entries that balance timeliness with broad coverage, ensuring the inclusion of up-to-date knowledge such as recent events or venue information. This design simulates practical scenarios where a model must continually adapt to new information over time. After filtering out noisy and redundant samples, we arrive at a set of well-structured, temporally relevant source questions.

A.2 Training Data Construction

To simulate scenarios in which factual and counterfactual information coexist, we make use of GPT-4o to generate question–answer pairs based on previously extracted questions. In the first round (Step 1), we ensure that the model is presented with answers reflecting real-world facts, establishing an initial baseline of correct knowledge. From Steps 2 through 5, we iteratively introduce answers that conflict with those presented in the previous round. To minimize data leakage or contamination during training, we apply several filtering strategies. First, we employ automated scripts to identify and remove duplicate counterfactual samples, as well as to detect any form of prior exposure the model might have to certain fabricated facts. Second, we conduct trial inference runs: if the model can correctly respond to a counterfactual query before it has been explicitly trained on that query, then the corresponding sample is replaced or discarded. This ensures that the model does not retain "prior memory" of newly introduced material, enabling a clearer assessment of how well it learns, forgets, or corrects knowledge over multiple rounds.

Type	Construction and Format
Q1	**Method:** Rewrite original Q in MC format with five options (A–E) separated by "|." **Format:** *"Please select the correct option..."* **Answer:** Use the content of the current correct answer (e.g., Answer C).
Q2	**Method:** Form a true statement from the original Q and current answer. Avoid real-world claims. **Format:** *"True/False: (Current Answer) is (Description)..."* **Answer:** "True."
Q3	**Method:** Use a conflicting answer to create a false statement. **Format:** *"True/False: (Another Answer) is (Description)..."* **Answer:** "False."
Q4	**Method:** Rewrite the original Q, replacing the answer part with "____." **Format:** *"Question: XXX (Replace answer with ____)."* **Answer:** A single word or short phrase (current correct answer).

Table 9: An overview of four question types (MC, True/False-True,True/False-False and Fill-in-the-Blank) with their construction methods and formats.

A.3 Test Data Construction

During the test phase, each original question and its five associated answers (one factual plus four counterfactuals) are transformed into four distinct test items to examine the model's comprehension and retention of the respective answer. We adopt four types of questions including multiple choice, True/False (True), True/False (False), and fill-in-the-blank tasks to systematically evaluate performance from various angles. In the multiple-choice questions, we rewrite the original query into a clearly stated prompt and provide the five potential answers (Answers A through E) as options. The model is instructed to select only the correct choice for the current round. In the True/False (True) questions, a statement is constructed to be consistent with the current answer, and the model should indicate "True" if it accurately comprehends the answer. Conversely, the True/False (False) questions involve pairing the statement with a conflicting answer to verify whether the model can identify logical discrepancies. Finally, the fill-in-the-blank questions mask the correct answer slot in the rephrased prompt using a placeholder such as "__" which the model must populate accurately with the

Step	Metric	CIT	MoCL	F-Learning	Ours
Step1	KGR	61.00	–	–	–
	ACC	74.20	–	–	–
Step2	KGR	41.20	44.80	62.40	**77.90**
	ACC	44.65	51.95	64.76	**76.32**
Step3	KGR	38.50	55.60	44.70	**81.10**
	ACC	38.90	58.85	51.80	**81.46**
Step4	KGR	32.80	42.60	62.20	**74.10**
	ACC	36.40	49.40	62.45	**74.68**
Step5	KGR	21.60	26.30	53.20	**82.60**
	ACC	23.50	23.50	53.26	**82.30**

Table 10: Performance of Llama3-70B under Setting I using LoRA for efficient fine-tuning.

current correct answer.

By employing these four question types for each of the five possible answers, we offer a multifaceted assessment of the model's ability to distinguish among factual knowledge, conflicting claims, and self-contradictory content. This approach allows us to analyze how well the model updates its representations in the presence of misinformation and whether it can preserve previously learned correct knowledge without succumbing to catastrophic forgetting. The resulting dataset effectively balances clarity for human interpretation and high utility for automated evaluation in a continuous learning framework.

B Evaluation

Post-Injection Accuracy (ACC) is used to evaluate the overall improvement in the model's answering accuracy after knowledge injection. This metric focuses on the model's precision in absorbing and correctly applying new knowledge, reflecting its reliability in practical application scenarios. The calculation formula for ACC is:

$$ACC = \left\{ \frac{|\mathcal{C}_{\text{correct\_post}}|}{|\mathcal{C}_{\text{total\_post}}|} \;\middle|\; \mathcal{C}_{\text{total\_post}} \neq \varnothing \right\} \quad (5)$$

where $\mathcal{C}_{\text{correct\_post}}$ represents the set of questions the model answered correctly after additional training, and $\mathcal{C}_{\text{total\_post}}$ represents the total set of questions evaluated after further training. The resulting value is interpreted as a percentage, indicating that a higher ACC value signifies the model's enhanced ability to answer questions accurately, demonstrating the positive impact of incorporating new information on the model's overall performance.

Knowledge Gain Ratio (KGR) is a key metric used to measure the degree of improvement in the model's answering capabilities before and after the learning process. Specifically, KGR quantifies the proportion of questions that were answered incorrectly before training but were corrected afterward. This metric reflects the effectiveness of the learning process in addressing the model's knowledge gaps and improving its adaptability to new information.The calculation formula for KGR is as follows:

$$KGR = \left\{ \frac{|\mathcal{C}_{\text{inc\_pre}} \cap \mathcal{C}_{\text{cor\_post}}|}{|\mathcal{C}_{\text{inc\_pre}}|} \;\middle|\; \mathcal{C}_{\text{inc\_pre}} \neq \varnothing \right\} \quad (6)$$

where $\mathcal{C}_{\text{incorrect\_pre}}$ represents the set of questions the model answered incorrectly prior to additional training, and $\mathcal{C}_{\text{correct\_post}}$ represents the set of questions the model answered correctly after further training. By calculating the intersection of these two sets, KGR intuitively measures the model's ability to effectively incorporate new information to improve performance. The resulting value is then interpreted as a percentage.

Knowledge Retention Rate (KRR) is a crucial metric for evaluating the model's capability to preserve its original knowledge while incorporating newly injected information. Specifically, KRR quantifies the proportion of questions that were answered correctly both before and after knowledge injection, reflecting the model's resistance to negative interference from new data. The formal definition of KRR is as follows:

$$KRR = \left\{ \frac{|\mathcal{C}_{\text{cor\_pre}} \cap \mathcal{C}_{\text{cor\_post}}|}{|\mathcal{C}_{\text{cor\_pre}}|} \;\middle|\; \mathcal{C}_{\text{cor\_pre}} \neq \varnothing \right\} \quad (7)$$

where $\mathcal{C}_{\text{cor\_pre}}$ represents the set of questions answered correctly prior to injection, and $\mathcal{C}_{\text{cor\_post}}$ denotes the set of questions still answered correctly post-injection. By computing the intersection of these two sets, KRR intuitively captures the model's ability to retain its prior knowledge base. A higher KRR suggests a more stable retention of the original knowledge, demonstrating the model's robustness against forgetting.

Topic	Questions	Correct Answer	Candidates
Medicine	Which technology is the most commonly used to remotely monitor patients with chronic conditions?	Telemedicine	A. Artificial Intelligence B. Virtual Reality C. Augmented Reality D. Bioprinting
	Which healthcare data analysis tool is mostly widely used for patient cohort identification and analysis?	SAS	A. SPSS B. R C. Tableau D. Microsoft Excel
Finance	Which of the following government programs is primarily designed to help seniors with their healthcare costs?	Medicare	A. Medicaid B. Social Security C. TANF D. SNAP
	Which of these following companies is NOT a major player in the travel finance industry?	Amazon	A. Skyscanner B. Booking Holdings C. Expedia D. TripAdvisor
Travel	Which of these factors is considered the most significant driver of tourism economic forecasting?	Economic growth	A. Weather patterns B. Technology advancements C. Currency exchange rates D. Political stability
	Which country's national airline was the first in the world to offer a luxury space tourism package?	United Arab Emirates	A. Japan B. United Kingdom C. United States D. China
Science	What is the earliest black hole that was discovered by humans?	Cygnus X-1	A. Leo X-1 B. Aquila X-1 C. Eridanus X-1 D. Orion X-1
	What constant is used to express the speed of the expansion of the universe?	Hubble	A. Newton B. Maxwell C. Planck D. Einstein
...	...	...	...

Table 11: Several examples from our meticulously crafted dataset. As demonstrated above, all the questions come from diverse active fields, covering up-to-date information with exceptional quality.

C Training Setup

For our experiments, we employed the Llama-Factory (Zheng et al., 2024b) framework to facilitate the training process. Notably, for the LLaMA3-70B model, we utilized DeepSpeed ZeRO (Rasley et al., 2020) to enhance memory efficiency and accelerate training. During the model evaluation phase, we leveraged the vLLM (Kwon et al., 2023) framework to streamline inference and ensure efficient evaluation. All experiments were conducted on NVIDIA A100 GPUs with 80GB of memory to meet the computational demands of large-scale models.

D Supplementary Experiments for Large-Scale Models

In this subsection, we conduct supplementary experiments on Llama3-70B under Setting I. Due to resource constraints, LoRA is employed for parameter-efficient fine-tuning, with the rank set to 16. Appendix F shows that during multi-stage continual learning, both full fine-tuning and parameter-efficient fine-tuning exhibit similar trends. As shown in Table 10, our experimental results demonstrate that our approach remains effective even on larger-scale models.

E Impact of Different Quantization Methods

Table 12 presents the performance variations of the two metrics under different quantization levels, with our LoRA rank set to 16. The experimental results indicate that model performance improves as the quantization level increases, suggesting that higher precision quantization yields better performance, highlighting the advantages of using higher bit-width quantization for knowledge retention and

QL	CIT (%)		Ours (%)	
	KGR	ACC	KGR	ACC
Int2	14.90	19.20	$66.90_{\uparrow 52.00}$	$67.85_{\uparrow 48.65}$
Int4	55.80	56.75	$86.10_{\uparrow 30.30}$	$85.95_{\uparrow 29.20}$
Int8	49.00	50.35	$84.10_{\uparrow 35.10}$	$84.10_{\uparrow 33.75}$
FP16	53.80	55.25	$83.60_{\uparrow 29.80}$	$83.90_{\uparrow 28.65}$

Table 12: Impact of Quantization Methods on KGR and ACC. Subscript values indicate the absolute increase (in red) compared to CIT.

Rank	CIT (%)		Ours (%)	
	KGR	ACC	KGR	ACC
4	39.60	41.50	$78.10_{\uparrow 38.50}$	$78.35_{\uparrow 36.85}$
8	48.80	49.90	$71.30_{\uparrow 22.50}$	$71.45_{\uparrow 21.55}$
16	53.80	55.25	$83.60_{\uparrow 29.80}$	$83.90_{\uparrow 28.65}$
Full	62.80	63.40	$90.40_{\uparrow 27.60}$	$90.30_{\uparrow 26.90}$

Table 13: Impact of LoRA Rank Settings on KGR and ACC.

task accuracy.

F Setting Impact on Performance

Table 13 presents the performance variations of the two metrics under different rank settings, with our quantization level set to FP16. Increasing the LoRA rank from 4 to 16 resulted in significant improvements in both KGR and ACC. The results indicate that higher LoRA ranks can enhance the model's ability to retain and generate knowledge, and using full fine-tuning may yield even better learning outcomes.