Aligning Language Models with Real-time Knowledge Editing

Anonymous ACL submission

Abstract

Knowledge editing aims to modify outdated knowledge in large language models (LLMs) efficiently while retaining their original capabilities. Mainstream benchmarks for knowledge editing are predominantly static and fail to keep in pace with the evolving real-world knowledge. In this work, we introduce CRAFT, an ever-evolving real-world benchmark for knowledge editing. It features well-designed paired edits for composite reasoning, and evaluates models on alias portability as well as temporal and common-sense locality, making it a challenging knowledge editing benchmark on which previous knowledge editing methods hardly achieve balanced performance. Towards flexible real-time editing, we propose KEDAS, a novel paradigm of knowledge editing alignment featuring diverse edit augmentation and self-adaptive post-alignment inference, which exhibits significant performance gain on CRAFT compared to previous methods. All of our code and data are available at `https://anonymous.4open.science/r/CRAFT-KEDAS`.

1 Introduction

Large language models (LLMs) (Llama Team, 2024; Gemini Team, 2025) have been the core of modern natural language processing (NLP). Once pre-trained, knowledge is injected into LLMs and becomes their static internal capability (Petroni et al., 2019). As time goes by, some knowledge in LLMs inevitably becomes incorrect or out of date, and it is vital to update the outdated knowledge. However, retraining a model from scratch is highly costly, especially for downstream users. To this end, the techniques of knowledge editing have been developed (Zhang et al., 2024), aimed at editing specific knowledge in LLMs efficiently without heavy re-training.

There have been a large number of benchmarks to evaluate the effectiveness of knowledge editing methods. Nevertheless, most traditional knowledge editing benchmarks like ZsRE (Levy et al., 2017), MQuAKE (Zhong et al., 2023) and RippleEdits (Cohen et al., 2024a) are static and not real-time. Once the datasets are constructed and released, they remain fixed and can not be updated anymore, which is far from real-world knowledge editing applications. Recently, some work is devoted to building real-time datasets. For example, WikiBigEdit (Thede et al., 2025) collects real-world changes in Wikipedia, but it requires processing massive Wiki data (usually hundreds of gigabytes) from the Internet, which is costly and inconvenient in practice. Meanwhile, the automatically collected dataset lacks necessary filtering and thus suffers from severe sparsity (for example, only 3.6% of the collected data have an entry of portability evaluation). EvoWiki (Tang et al., 2025) is another real-time dataset also based on Wikipedia, but it focuses on retrieval-augmented generation (RAG) and continual learning. Overall, there still lacks a high-quality, easily-updated benchmark for real-world and real-time knowledge editing.

Benchmark	Real-time	Real-world	Reasoning Portability	Composite Reasoning Portability
ZsRE	✗	✓	✗	✗
MQuAKE-T	✗	✓	✓	✗
MQuAKE-CF	✗	✗	✓	✓
RippleEdits	✗	✓	✓	✗
WikiBigEdit	✓	✓	✗	✗
EvoWiki	△	✓	✓	✗
CRAFT (Ours)	✓	✓	✓	✓

Table 1: Comparison of knowledge editing benchmarks. **Composite reasoning portability** evaluates a model's ability to integrate multiple distinct edits into a single reasoning query, whereas **reasoning portability** tests reasoning chains derived from a single edit.

In this work, we introduce CRAFT (**C**hinese **R**eal-time statistics **A**nd **F**inance knowledge edi**T**ing benchmark), an ever-evolving real-world

benchmark for knowledge editing in Chinese. It leverages publicly available official data that are continuously updated, ensuring both temporal freshness and real-world applications. Furthermore, it is organized in well-designed paired edits, each serving as a composite reasoning test to evaluate models' ability to integrate multiple related factual updates. Moreover, CRAFT supports evaluations on alias portability and temporal and common-sense locality, providing a comprehensive assessment of model adaptability and robustness under dynamic knowledge changes. The differences between CRAFT and previous datasets are summarized in Table 1.

We assess the exposure rate of CRAFT and existing benchmarks with five different LLMs, either open-source or closed-source. The results demonstrate that a large portion of knowledge in existing benchmarks has been known to LLMs while the exposure rate of CRAFT is nearly zero on all the five models. This reveals the disadvantages of static datasets and validates the real-time property of CRAFT, which makes it challenging to LLMs.

We evaluate a suite of representative knowledge editing methods on CRAFT and demonstrate that current approaches exhibit inherent limitations. For example, parameter-based approaches like ROME (Meng et al., 2022) and WISE (Wang et al., 2024a) suffer from gradual model degradation and struggle to achieve successful editing in the setting of sequential editing. Simple retrieval-based approaches like IKE (Zheng et al., 2023) and EREN (Chen et al., 2024b) fail to achieve consistent balanced performance due to lack of alignment. LTE (Jiang et al., 2024), which aligns LLMs with knowledge editing via post-training, shows weak locality due to overfitting on irrelevant queries.

Towards flexible real-time knowledge editing, we propose KEDAS [1], namely **K**nowledge **E**diting alignment with **D**iverse **A**ugmentation and **S**elf-adaptive inference, an advanced knowledge editing framework featuring diverse representations of edits and flexible inference paths. The alignment is a one-time offline stage, where the LLM is finetuned with low-rank adaptation (LoRA) (Hu et al., 2022). Once the alignment is finished, users can perform knowledge editing without modifying any parameters. During editing, a novel technique of diverse edit augmentation is introduced to store different forms of each edit and thus improve the generalization of applied edits. In the inference phase, a filter-enhanced smart retriever is employed to adaptively select the base model or the aligned model. If any edits are recalled from the memory, the prompt will go through the LLM with LoRA activated. Otherwise, the prompt just passes to the original model before alignment to guarantee locality and avoid over-fitting. On CRAFT, KEDAS exhibits outstanding performance in all metrics and significantly outperforms previous methods, illustrating an ideal paradigm of knowledge editing alignment.

Our contributions are as follows:

- We introduce CRAFT, a high-quality, easily-updated and fully open-source benchmark for real-time knowledge editing.
- We evaluate existing knowledge editing methods on CRAFT and reveal the limitations of these methods in real-time knowledge editing.
- We propose KEDAS, a novel paradigm of aligning LLMs with real-time knowledge editing featuring diverse edit augmentation and self-adaptive post-alignment inference, enabling dynamic routing between pre- and post-alignment LLMs, which significantly outperforms existing methods on CRAFT.

2 Preliminary

2.1 Task Formulation

An LLM system f can be regarded as a function $f : \mathcal{Q} \mapsto \mathcal{A}$ mapping an query $q \in \mathcal{Q}$ to its output answer $a = f(q) \in \mathcal{A}$.

Knowledge editing aims to change the behavior of the LLM after modifying some knowledge. Edits of knowledge are usually in the form of query-answer (QA) pairs:

$$\mathcal{E} = \{e^t\}_{t=1}^N = \{(q_e^t, a_e^t)\}_{t=1}^N, \quad (1)$$

where q_e^t is an input query triggering knowledge (*e.g.*, `The current US president is`), a_e^t is the corresponding target of editing (*e.g.*, `Donald Trump`), t is the index of each edit, and N denotes the total number of edits.

To assess the efficacy of editing, the post-edit LLM f^* is evaluated via the following criteria (Zhang et al., 2024):

Edit success measures the accuracy of editing, requiring f^* to correctly recall the edits:

$$\mathbb{E}_{(q_e,a_e)\in\mathcal{E}}\, \mathbb{1}\{f^*(q_e) = a_e\}. \quad (2)$$

[1] /ˈkiːdəs/, pronounced as "kee-das".

Locality measures the precision of editing, requiring f^* not to change its behavior out of the scope of edits:

$$\mathbb{E}_{(q_l,a_l)\in\mathcal{E}_l}\ \mathbb{1}\{f^*(q_l) = f(q_l)\}, \tag{3}$$

where $\mathcal{E}_l$ are QA pairs of unrelated queries. Note that the target of locality is unchanged answer rather than exactly the ground truth answer.

Portability measures how well edits transfer to related queries, requiring f^* to correctly answer such queries:

$$\mathbb{E}_{(q_p,a_p)\in\mathcal{E}_p}\ \mathbb{1}\{f^*(q_p) = a_p\}, \tag{4}$$

where $\mathcal{E}_p$ are QA pairs related to existing edits.

2.2 Settings of Knowledge Editing

Single Editing In single editing, each edit is evaluated directly after it is applied to the original model f:

$$f^t = \mathbf{Edit}(f, q_e^t, a_e^t), \quad 1 \le t \le N. \tag{5}$$

This traditional setting is widely employed by earlier work. It is far from real-world applications, since only one edit can be applied each time.

Sequential Editing In sequential editing, edits are applied step by step (let f^0 denote f for consistency):

$$f^t = \mathbf{Edit}(f^{t-1}, q_e^t, a_e^t), \quad 1 \le t \le N. \tag{6}$$

The final model f^N after applying all N edits is evaluated. This setting is frequently used to evaluate the lifelong performance and scalability of knowledge editing. We focus on this setting in this work because it is closer to practical applications, especially the real-time knowledge editing setting.

Knowledge Editing Alignment LTE (Jiang et al., 2024) relates knowledge editing to LLM alignment, aligning LLMs with ever-changing real-time knowledge edits. During alignment, LLMs' capabilities of knowledge updating are enlightened by an editing prompt:

```
[Updated Information] {edit}
[Query] {query}
```

The training data consist of in-scope and out-of-scope queries with or without the specific edit to improve edit efficacy while keeping locality.

In the inference phase, LLMs are required to conduct on-the-fly and streaming knowledge editing by retrieving relevant edits to the query from the stored memory.

3 The CRAFT Dataset

We introduce CRAFT, a real-time, real-world benchmark designed to evaluate knowledge editing under dynamically evolving factual contexts. Unlike static benchmarks that rely on outdated or widely exposed Wikipedia facts, CRAFT continuously updates with publicly available official data sources, ensuring fairness and real-world relevance.

3.1 Dataset Overview

CRAFT consists of two complementary subsets: **CRAFT-Statistics** and **CRAFT-Finance**, representing stable and highly correlated factual domains that evolve periodically. Example instances from the two subsets are shown in Appendix E and F respectively.

CRAFT-Statistics We collect monthly statistical reports from the official National Bureau of Statistics of China [2], covering July 2024 to June 2025 in the current version used in this work. The dataset spans four major domains—finance, fiscal affairs, telecommunications, and transportation—containing 221 indicators and 4,468 data points. Our data collection pipeline, leveraging the `cn-stats` API [3], supports automatic temporal updates, enabling effortless retrieval of the most recent monthly data.

CRAFT-Finance CRAFT-Finance includes annual financial statements of 390 publicly listed Chinese companies (2023–2024), sourced from Eastmoney Financial Database [4] via the `AKShare` API [5]. It features financial indicators such as debt ratio, shareholder equity, and operating cash flow, supporting yearly updates to reflect the latest financial disclosures.

3.2 Evaluation Metrics

To comprehensively assess model adaptability, besides the conventional edit success (ES) as described in Section 2.1, CRAFT provides four evaluation dimensions:

Composite Reasoning Portability ($\mathbf{P_{reasoning}}$) We propose *composite reasoning portability*, a new evaluation designed specifically for CRAFT. Unlike traditional *multi-hop reasoning portability*, which constructs reasoning chains from a single

[2] https://data.stats.gov.cn/
[3] https://github.com/songjian/cnstats
[4] https://data.eastmoney.com/
[5] https://github.com/akfamily/akshare

Dataset	GPT-3.5 (OpenAI, 2023)	Mistral-7B (Jiang et al., 2023)	Llama-3.1-8B (Llama Team, 2024)	DeepSeek-chat (DeepSeek-AI, 2025)	Doubao-1.5-pro-32k (ByteDance, 2025)	Size
ZsRE (Levy et al., 2017)	43.98%	23.48%	22.45%	52.47%	26.77%	6,000
MQuAKE (Zhong et al., 2023)	35.46%	76.21%	46.11%	42.58%	76.09%	5,604
EvoWiki_evolved (Tang et al., 2025)	8.07%	16.22%	6.43%	10.24%	5.08%	1,338
EvoWiki_stable (Tang et al., 2025)	28.45%	42.37%	25.90%	39.76%	12.32%	1,494
EvoWiki_uncharted (Tang et al., 2025)	8.19%	22.62%	7.39%	16.02%	3.43%	1,136
CRAFT-Statistics (Ours)	0.01%	0.00%	0.30%	1.80%	0.99%	4,468
CRAFT-Finance (Ours)	0.00%	0.00%	0.14%	0.42%	0.00%	7,800

Table 2: Exposure rate (%) of different LLMs to benchmark datasets. Model release years are indicated in citations. CRAFT exhibits minimal exposure due to real-time data collection and domain freshness.

edit, composite portability evaluates a model's ability to integrate multiple distinct edits into a single reasoning query.

In CRAFT, all data are organized in *paired format*: each test instance consists of *two edits* and one corresponding composite reasoning query. Formally, given a pair of edits $\{e_1, e_2\}$, the composite portability query q_p requires reasoning over both edits simultaneously. For example, consider a CRAFT-Finance instance with a pair of edits:

e_1 : 2024 Company A's total assets = 2M

e_2 : 2024 Company A's total liabilities = 1M

The corresponding composite reasoning query and answer are:

$$q_p = \text{2024 Company A's debt ratio}$$
$$a_p = 0.5$$

CRAFT-Finance includes ten types of such combinations, demonstrating the generality and practical relevance of composite reasoning portability. For CRAFT-Statistics, composite portability is constructed by pairing two edits of the same indicator from adjacent years.

Alias Portability ($\mathbf{P_{alias}}$) We propose *alias portability* to evaluate a model's robustness to synonymous indicator names by replacing key terms with domain-specific aliases constructed from a manually curated alias list.

Temporal Locality ($\mathbf{L_{temporal}}$) We propose *temporal locality* to measure a model's sensitivity to temporal shifts by querying facts from neighboring time periods.

Common-sense Locality ($\mathbf{L_{common}}$) We propose *common-sense locality* to measure whether a model's irrelevant general knowledge and reasoning capability remain consistent after knowledge editing. For CRAFT, the questions for this evaluation are directly sampled from the C^3 dataset (Sun et al., 2020), which contains 19,577 Chinese multiple-choice comprehension questions.

3.3 Dataset Statistics

Table 2 compares the exposure ratio of CRAFT and existing benchmarks across five representative LLMs. Due to its temporal freshness and domain specificity, CRAFT exhibits extremely low exposure rates across all the five LLMs, mitigating data contamination and ensuring fair evaluation. Meanwhile, for other datasets, a great portion of the knowledge has been known to LLMs, making them less challenging and less fair in practice.

Dataset	Edit	$\mathbf{P}_{\text{reasoning}}$	$\mathbf{P}_{\text{alias}}$	$\mathbf{L}_{\text{temporal}}$	$\mathbf{L}_{\text{common}}$
CRAFT-Statistics	4,468	2,234	4,468	4,468	4,468
CRAFT-Finance	7,800	3,900	7,800	7,800	7,800

Table 3: Number of instances per evaluation type in CRAFT. Composite reasoning uses edit pairs; other metrics are per edit.

Table 3 shows instance counts in CRAFT subdatasets. Composite reasoning portability is evaluated per edit pair, and other metrics per edit. For experiments, we split CRAFT into training and test sets. The test set contains 500 instances from the two subsets each (1,000 instances in total) and the training set contains the remaining instances.

3.4 Data Summary

CRAFT offers a scalable and continuously evolving benchmark reflecting real-world knowledge updates. Its dynamic nature, factual authenticity, and comprehensive reasoning evaluation make it a robust foundation for future research on reliable and interpretable knowledge editing. **Anyone who wants to construct CRAFT within an arbitrary time period can easily obtain a customized benchmark with our provided open-source scripts.**

4 KEDAS: A Proposal for Real-time Knowledge Editing Alignment

As will be shown in the experiments, previous methods show certain limitations either in edit success and portability or locality on CRAFT. To improve edit success and generalization while keeping a good locality in real-time knowledge editing, we propose KEDAS, an all-round approach consisting of several complementary techniques.

The overall framework of KEDAS is illustrated in Figure 1, including alignment, editing and inference. In alignment, the LLM is aligned with knowledge editing using LoRA. In editing, each edit is converted into diverse forms and stored to the memory module. In the inference phase, edits are retrieved from the memory based on the user query and then filtered by a trained classifier. If there exists relevant edits, the model with trained LoRA adapters will be called for inference. Otherwise, the query will go through the frozen base model. The order of editing and inference is flexible based on practical needs ("edit as you go"), as shown by the bidirectional arrow. For example, we can apply several edits and employ the LLM for inference immediately. If there come further new edits, we can return to the editing phase to conduct these edits and then resume inference.

4.1 Alignment Based on LoRA

LTE (Jiang et al., 2024) suffers from a limited locality because it employs the post-alignment model, whose behavior can be significantly different from the original one, to all incoming queries. To address this, we propose the idea of self-adaptive post-alignment inference. To achieve this elegantly, we align the base model via parameter-efficient fine-tuning with LoRA (Hu et al., 2022), which tunes additional adapters during alignment. This costs low computational resources for both training and inference, making KEDAS efficient. We follow LTE to construct alignment data using the training set of CRAFT and details are in Appendix A.

4.2 Diverse Edit Augmentation

Form	Content
QA	What are Microsoft's total assets? $619B.
Declarative	Microsoft's total assets are $619B.
Aliased	MSFT's total assets are $619B.

Table 4: A demonstration of diverse edit expressions.

In previous work, each piece of edited knowledge is directly applied to the system as it is, limiting generalization. Since each knowledge can be expressed in various ways, leveraging just one fixed form is inflexible, especially for challenging portability evaluations like in CRAFT.

In this work, we propose diverse edit augmentation, a novel technique that augments each edit by converting it into multiple expressions. The *de facto* standard form of edits in knowledge editing is a QA pair, typically describing the relation between entities or attributes of them. Based on this observation, we manually design several ways of form augmentation, including *declarative* and *aliased* forms, as demonstrated in Table 4.

Concretely, for an edit $e^t = (q_e^t, a_e^t)$ at step t [6], its conventional QA form is:

$$e_{\text{qa}}^t = q_e^t \oplus a_e^t, \tag{7}$$

where $\oplus$ denotes concatenation. Then, it is augmented as:

$$e_{\text{dec}}^t = \textbf{Declarative}(q_e^t, a_e^t), \tag{8}$$
$$e_{\text{als}}^t = \textbf{Aliased}(q_e^t, a_e^t). \tag{9}$$

Specifically, *declarative* is to convert the QA pair into a declaration while *aliased* is to replace specialized terms in the declaration with their aliases for more flexible retrieval.

In experiments, we conduct these augmentations by prompting `gpt-4o-mini` [7], an API-based off-the-shelf LLM. The prompts are in Appendix C.

We store the augmented diverse edits into memory for future retrieval:

$$\mathcal{M}^0 = \emptyset, \quad \mathcal{M}^t = \mathcal{M}^{t-1} \cup \{e_{\text{qa}}^t, e_{\text{dec}}^t, e_{\text{als}}^t\}, \tag{10}$$

where $\mathcal{M}^t$ denotes the memory at time step t.

Note that our augmentation methods are specifically designed for the CRAFT dataset. There can be more diverse augmentations like paraphrasing in practice for broader use.

4.3 Filter-based Smart Retriever

During inference, the common practice regarding retrieval in previous knowledge editing work is to directly employ an embedder and perform single-stage retrieval to obtain the top-k hits. This suffers from a poor locality due to the lack of a filter

[6] For simplicity, we deem the two edits in each CRAFT instance as two separate time steps.

[7] https://platform.openai.com

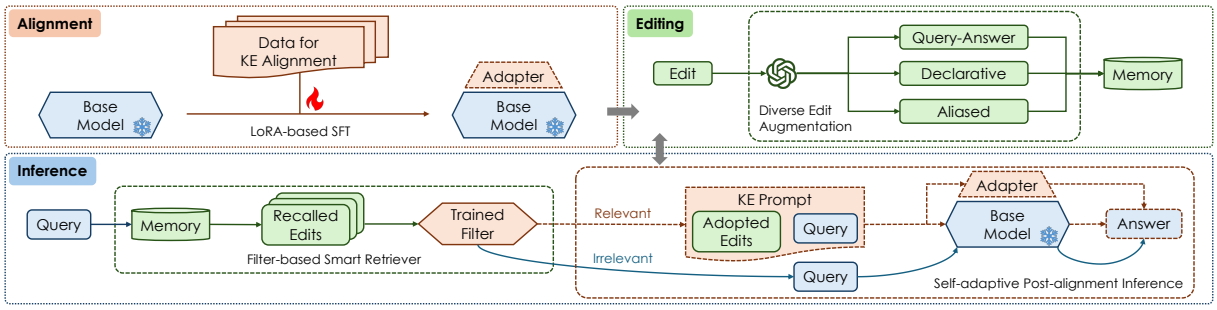

Figure 1: Illustration of our proposed KEDAS framework.

measuring the relevance of queries. In this work, we propose a filter-based smart retriever that pairs the high-recall retrieval with a high-precision filter by precisely identifying relevant queries. In this way, on challenging benchmarks like CRAFT, the post-edit system can keep consistent behaviors for neighboring locality queries while preserving its answers to general queries.

Firstly, we retrieve top-n candidates from the memory using a normal embedding-based retriever. Thanks to our diverse edit augmentation, the memory contains various forms of edits, improving the recall rate. At the time step t, for an input query q, a set of top-n candidates $\mathcal{C}^{\text{top-}n}$ is retrieved as:

$$\mathcal{C}^{\text{top-}n} = \mathbf{Retriever}^{\text{top-}n}(q, \mathcal{M}^t). \quad (11)$$

Then, a binary classifier θ is exploited to predict the relevance between candidates and a query:

$$\mathcal{C}^{\text{filtered}} = \{e \mid \mathbf{Filter}_\theta(e) = 1, e \in \mathcal{C}^{\text{top-}n}\}. \quad (12)$$

Then we only adopt the top-k edits with the highest similarity to the query if the filtered candidate set contains too many edits:

$$\mathcal{E}^* = \arg\text{top}_k(\mathbf{Similarity}(e, q)) \text{ for } e \in \mathcal{C}^{\text{filtered}}. \quad (13)$$

The filter is trained based on the training set of CRAFT. Details are specified in Appendix D.

4.4 Self-adaptive Post-alignment Inference

In the inference phase, we employ both the original pre-alignment model and the post-alignment adapter, which does not consume extra memory while enabling two inference paths. Each time a query q is requested, the filter-based smart retriever is utilized to recall possible relevant edits in the memory. If there exists a set of relevant edits $\mathcal{E}^*$, indicating that the query involves edited knowledge, we fill both $\mathcal{E}^*$ and q into the knowledge editing prompt (see Section 2.2) and then feed the prompt into the model with the post-alignment adapters activated. Otherwise, the query directly goes through the pre-alignment model without any processing. The inference strategy is formulized below:

$$a = \begin{cases} f_\Phi(q) & \text{irrelevant} \\ f_{\Phi+\Delta\Phi(\Theta)}(\mathbf{KEPrompt}(\mathcal{E}^*, q)) & \text{relevant} \end{cases}, \quad (14)$$

where a denotes the final answer, f_Φ denotes the original model parametrized by Φ, $f_{\Phi+\Delta\Phi(\Theta)}$ denotes the model with post-alignment adapters, and $\mathbf{KEPrompt}(\cdot,\cdot)$ denotes the knowledge editing prompt template.

5 Experiments

We evaluate representative knowledge editing methods and our proposed KEDAS on CRAFT. Our experiments are based on the framework of EasyEdit (Wang et al., 2024b).

5.1 Experimental Setup

Language Model In our experiments, we focus on Llama-3.1-8B-Instruct (Llama Team, 2024), a widely used open-source language model.

Evaluated Methods We evaluate the following knowledge editing methods:

- **LoRA** (Zhang et al., 2024) directly updates knowledge by LoRA-based finetuning.
- **ROME** (Meng et al., 2022) locates multilayer perceptron (MLP) weights of related knowledge and writes in new key-value pairs.
- **IKE** (Zheng et al., 2023) leverages in-context examples of copying, updating and retaining knowledge to edit knowledge via prompting. Due to limited resources, we only evaluate the 8-shot setting.

Method	CRAFT-Statistics					CRAFT-Finance				
	ES	$L_{temporal}$	L_{common}	$P_{reasoning}$	P_{alias}	ES	$L_{temporal}$	L_{common}	$P_{reasoning}$	P_{alias}
LoRA	47.93	0.71	0.00	41.56	47.62	44.19	13.97	0.00	9.81	7.43
ROME	21.59	0.00	12.20	13.67	21.59	4.25	4.00	14.00	7.62	4.40
IKE	**99.82**	5.49	11.20	43.03	55.73	**99.98**	27.50	11.60	40.71	57.86
EREN	22.12	91.68	63.20	24.99	22.29	24.37	**70.93**	65.80	25.48	22.27
WISE	47.54	29.74	81.00	41.26	47.30	32.66	49.62	79.80	34.41	32.48
LTE	97.90	5.44	12.60	68.58	54.14	80.68	27.94	8.40	50.21	60.98
KEDAS (ours)	99.80	**100.00**	**100.00**	**81.20**	**62.17**	99.85	62.50	**100.00**	**55.95**	**70.61**

Table 5: Main results on CRAFT. The highest and second-highest scores are **bolded** and underlined, respectively.

- **EREN** (Chen et al., 2024b) prompts LLMs to decide the relevance of retrieved edits and then queries LLMs with or without edits.
- **WISE** (Wang et al., 2024a) stores edits in a parametric side memory module and utilized a router to decide which memory to go through.
- **LTE** (Jiang et al., 2024) aligns LLMs with the knowledge editing task by post-training. We adopt paraphrase-multilingual-MiniLM-L12-v2 (Reimers and Gurevych, 2019) as the retriever and set the number of adopted edits k as 3, the default setting of LTE.
- **KEDAS** is our proposed method. We use the same retriever as LTE and adopt bert-base-chinese (Devlin et al., 2019) as the binary filter. We set the number of retrieved candidates n as 8 and that of adopted edits k as 3, in line with LTE for fair comparison.

Evaluation Metrics We use the metrics described in Section 3.2, including ES, $L_{temporal}$, L_{common}, $P_{reasoning}$ and P_{alias}.

5.2 Main Results

The main results on CRAFT are presented in Table 5. Previous methods of knowledge editing show limitations in certain aspects. LoRA and ROME, as traditional methods primarily designed for single editing or limited number of edits, show no advantage when applied to the sequential editing setting on CRAFT. IKE, the retrieval-based approach, suffers from poor locality and mediocre portability, albeit achieves the highest edit success scores. The prompting-based EREN requires specific prompt optimization for different LLMs and can scarcely perform editing with its official default prompt, as can be seen from the low edit success. WISE is parameter-based and suffers from downgrading performance under the sequential setting, with weak performance in most metrics. LTE, with an alignment stage, achieves relatively robust performance in edit success and portability but struggles to achieve locality due to its fixed inference path. These results indicate that CRAFT is a challenging benchmark for existing knowledge editing methods, making it hard for them to achieve a consistently balanced performance.

Meanwhile, KEDAS exhibits outstanding performance on CRAFT, securing the highest scores in 7 out of 10 metrics. Due to some false negatives of the filter, the edit success of KEDAS is slightly lower than IKE inevitably. On the Finance subset of CRAFT, KEDAS is outperformed by EREN in $L_{temporal}$ possibly because the BERT-based filter suffers from limited generalization in certain cases. Overall, KEDAS outperforms WISE, the representative parameter-based approach, by 39.26 and 31.99 points of averaged metrics on the Statistics and Finance subsets respectively, and surpasses LTE, a strong alignment-based baseline, by 40.90 and 32.14 averaged points on the two subsets.

In Appendix G, we also compare KEDAS with previous methods on traditional knowledge editing benchmarks. The results show that KEDAS significantly outperform previous methods with balanced performance and confirm the effectiveness of KEDAS even on traditional benchmarks.

5.3 Ablation Study

In this section, we assess the indispensability of components of KEDAS, including diverse edit augmentation (DEA), filter (FLT) and self-adaptive post-alignment inference (SPI). The results are presented in Table 6. The exclusion of diverse edit augmentation leads to a decline in edit success and portability, indicating that it contributes to the recall of the retriever. Meanwhile, both the filter and the paradigm of self-adaptive post-alignment inference show great contribution to locality, demonstrating that they are playing important roles in KEDAS. These results confirm that all the core

Method	CRAFT-Statistics					CRAFT-Finance				
	ES	$L_{temporal}$	L_{common}	$P_{reasoning}$	P_{alias}	ES	$L_{temporal}$	L_{common}	$P_{reasoning}$	P_{alias}
KEDAS	**99.80**	**100.00**	**100.00**	**81.20**	**62.17**	99.85	**62.50**	**100.00**	**55.95**	**70.61**
w/o DEA	96.51	**100.00**	**100.00**	71.02	59.89	98.10	**62.50**	**100.00**	53.90	63.35
w/o FLT	**99.80**	5.44	11.40	**81.20**	**62.17**	**100.00**	27.48	8.40	**55.95**	**70.61**
w/o SPI	**99.80**	5.68	10.80	**81.20**	**62.17**	99.85	27.42	8.40	**55.95**	**70.61**

Table 6: Ablation results. The highest scores are **bolded**.

components of KEDAS make contributions to the final performance and are indispensable.

6 Related Work

6.1 Knowledge Editing Benchmarks

Conventionally, knowledge editing benchmarks are created based on temporal changes or counter-fact modifications of world knowledge either from existing QA datasets (Levy et al., 2017; Yao et al., 2023) or Wikipedia (Cohen et al., 2024a; Zhong et al., 2023), in which Zhong et al. (2023) first propose multi-hop composite portability that involves multiple edits. However, these are all static and suffer from data contamination as time goes by.

Thede et al. (2025) propose WikiBigEdit, a dynamic benchmark based on periodical changes in Wikipedia. However, only a small fraction (3.6%) of its samples support reasoning portability evaluation, limiting its comprehensiveness. Similarly, Tang et al. (2025) propose EvoWiki, a Wiki-based real-time dataset (primarily designed for the RAG), where each instance is associated with reasoning portability tests. Although EvoWiki claims to automatically update with new edits, its implementation and data pipeline remain unpublished.

Besides conventional knowledge editing benchmarks, there have also been some benchmarks designed for special use. Mitchell et al. (2022b) propose ConvSent to edit LLMs' sentiments on given topics. Ishibashi and Shimodaira (2024) propose Sanitation to forget specific information stored in LLMs and thus address privacy concerns. Huang et al. (2025) propose HalluEditBench to evaluate whether knowledge editing correct hallucinations.

Our proposed CRAFT benchmark leverages statistical and financial data, which are both real-time and inherently dynamic. The strong interdependence among indicators allows each pair of edits to correspond to a meaningful reasoning portability test. This enables comprehensive evaluation across five dimensions: benchmark type, real-time property, real-world grounding, reasoning portability, and composite reasoning portability.

6.2 Knowledge Editing Methods

Parameter-based Editing This line of work aims to edit model parameters or add extra parameters to perform knowledge editing. Meng et al. (2022), Meng et al. (2023) and Fang et al. (2025) adopt a locate-then-edit manner and aim to precisely edit MLP weights. Mitchell et al. (2022a) adopts meta-learning (Hospedales et al., 2022) to predict the change of model parameters. Dong et al. (2022), Huang et al. (2023) and Wang et al. (2024a) add extra parameters to store edits in a parametric form. Mitchell et al. (2022b) trains a separate counter-fact model and does not involve retrieval (thus categorized as parameter-based), failing to fully exploit the capability of LLMs themselves.

Retrieval-based Editing This line of work stores edits in a memory module and retrieve from it when needed. Zheng et al. (2023) leverages in-context examples of copying, updating and retaining knowledge to edit knowledge via prompting. Chen et al. (2024b) first prompts LLMs to decide the relevance of retrieved edits and then queries LLMs with or without in-context edits. Chen et al. (2024a) converts edits into soft prompt tokens with trained models and then concatenates user prompts to modify the behavior of frozen LLMs. Jiang et al. (2024) aligns LLMs with knowledge editing by post-training with in-context edits. Our proposed KEDAS can also be categorized as retrieval-based.

7 Conclusion

In this work, we introduce CRAFT, a novel automatically updatable real-time benchmark for knowledge editing that focuses on national statistics and finance of China. We evaluate representative knowledge editing methods on CRAFT and reveal that existing methods have certain limitations. We then propose a simple yet effective approach, KEDAS, to better align LLMs with real-time knowledge editing. Experiments validate the effectiveness of KEDAS on CRAFT in all aspects including edit success, locality and portability.

Limitations

First, the CRAFT benchmark focuses on the Chinese language only (but our methodology is universal and can be applied to any other languages) and is limited to the domains of national statistics and finance (but it is easily expandable to domains that data are periodically updated and automatically accessible). Second, the composite reasoning portability of CRAFT is merely 2-hop (but can be expanded to multiple hops via combination of time and indicators). Third, the Finance subset of CRAFT can be updated annually only (but can be improved to be quarterly updated by substituting the annual financial statements to quarter ones). Fourth, only one Llama-3 LLM is employed due to limited time and a limited set of existing knowledge editing methods is evaluated in our experiment due to limited resources and compatibility issues.

Ethical Considerations

Computational Budget All our experiments are conducted on a machine with Ubuntu 20.04.6 LTS, Intel® Xeon® Silver 4310 CPU and 256G memory. We use one NVIDIA A40 48G GPU for all the experiments. The training of KEDAS takes about 2 hours. The editing of KEDAS takes about 50 minutes on both subsets of CRAFT.

Reproducibility All the experiments are fully reproducible since all the methods are deterministic and sampling is disabled during LLM generation.

Potential Risks To the best of our knowledge, there are no potential risks concerning our work.

Scientific Artifacts We cite all the creators of scientific artifacts we use in this paper. Licenses of these scientific artifacts are shown in Table 7. Our use of these artifacts is consistent with their intended use.

Privacy and Offense Concerns Our data source ensures that our CRAFT dataset contains no information that names or uniquely identifies individual people or offensive content.

References

ByteDance. 2025. Doubao-1.5-pro-32k model.

Qizhou Chen, Taolin Zhang, Xiaofeng He, Dongyang Li, Chengyu Wang, Longtao Huang, and Hui Xue'. 2024a. Lifelong knowledge editing for LLMs with retrieval-augmented continuous prompt learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*.

Yingfa Chen, Zhengyan Zhang, Xu Han, Chaojun Xiao, Zhiyuan Liu, Chen Chen, Kuai Li, Tao Yang, and Maosong Sun. 2024b. Robust and scalable model editing for large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024a. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024b. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*.

DeepSeek-AI. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

Qingxiu Dong, Damai Dai, Yifan Song, Jingjing Xu, Zhifang Sui, and Lei Li. 2022. Calibrating factual knowledge in pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. 2025. Alphaedit: Null-space constrained model editing for language models. In *The Thirteenth International Conference on Learning Representations*.

Gemini Team. 2025. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. 2022. Meta-learning in neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Baixiang Huang, Canyu Chen, Xiongxiao Xu, Ali Payani, and Kai Shu. 2025. Can knowledge editing really correct hallucinations? In *The Thirteenth International Conference on Learning Representations*.

Artifact	License
C^3	Non-commercial Research Purpose
cn-stats	MIT
AKShare	MIT
BERT	Apache
paraphrase-multilingual-MiniLM-L12-v2	Apache
EasyEdit	MIT
Llama-3	Llama 3 Community License Agreement

Table 7: Licenses of scientific artifacts we use.

Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. 2023. Transformer-patcher: One mistake worth one neuron. In *The Eleventh International Conference on Learning Representations*.

Yoichi Ishibashi and Hidetoshi Shimodaira. 2024. Knowledge sanitization of large language models. *Preprint*, arXiv:2309.11852.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Yuxin Jiang, Yufei Wang, Chuhan Wu, Wanjun Zhong, Xingshan Zeng, Jiahui Gao, Liangyou Li, Xin Jiang, Lifeng Shang, Ruiming Tang, Qun Liu, and Wei Wang. 2024. Learning to edit: Aligning LLMs with knowledge editing. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*.

Llama Team. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. 2022a. Fast model editing at scale. In *International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022b. Memory-based model editing at scale. In *Proceedings of the 39th International Conference on Machine Learning*.

OpenAI. 2023. Gpt-3.5 technical report. https://platform.openai.com/docs/models/gpt-3-5.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*.

Kai Sun, Dian Yu, Dong Yu, and Claire Cardie. 2020. Investigating prior knowledge for challenging chinese machine reading comprehension. *Transactions of the Association for Computational Linguistics*.

Wei Tang, Yixin Cao, Yang Deng, Jiahao Ying, Bo Wang, Yizhe Yang, Yuyue Zhao, Qi Zhang, Xuanjing Huang, Yu-Gang Jiang, and Yong Liao. 2025. EvoWiki: Evaluating LLMs on evolving knowledge. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Lukas Thede, Karsten Roth, Matthias Bethge, Zeynep Akata, and Tom Hartvigsen. 2025. Wikibigedit: Understanding the limits of lifelong knowledge editing in llms. *Preprint*, arXiv:2503.05683.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2024a. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. In *Advances in Neural Information Processing Systems*.

Peng Wang, Ningyu Zhang, Bozhong Tian, Zekun Xi, Yunzhi Yao, Ziwen Xu, Mengru Wang, Shengyu Mao, Xiaohan Wang, Siyuan Cheng, Kangwei Liu, Yuansheng Ni, Guozhou Zheng, and Huajun Chen. 2024b. EasyEdit: An easy-to-use knowledge editing framework for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, and 3 others. 2024. A comprehensive study of knowledge editing for large language models. *Preprint*, arXiv:2401.01286.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. 2023. Can we edit factual knowledge by in-context learning? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. 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)*.

Zexuan Zhong, Zhengxuan Wu, Christopher Manning, Christopher Potts, and Danqi Chen. 2023. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

A Details of Alignment Data for KEDAS

For each instance of CRAFT's training set $\mathcal{E}^t$, we construct the edit candidate set $\mathcal{E}^*$ by keeping the two edits e_1 and e_2 of the instance and adding 0-3 (randomly decided) extra retrieved edits with the same retriever as LTE that are different from the two to promote model's robustness to incorrect retrieved results.

To promote edit success, for each edit query q_e and answer a_e of either e_1 or e_2, the input is $\mathbf{KEPrompt}(\mathcal{E}^*, q_e)$ and the target is a_e.

To promote portability, for each portability query q_p and answer a_p in $\mathcal{E}_p^t$, the input is $\mathbf{KEPrompt}(\mathcal{E}^*, q_p)$ and the target is a_p.

```
You are a helpful assistant. You are given a query and a target new. Please generate
the declaration form of the query and target new.

Here is an example:
Query: 2024年华大智造的总资产（亿元）是多少?
Target new: 103.15
Declaration: 2024年华大智造的总资产（亿元）是103.15。

Please generate the declaration form of the query and target new below:
Query: {query}
Target new: {target_new}
Declaration:
```

Figure 2: Prompt template for the *declarative* form.

To promote locality, for each locality query q_l and answer a_l in $\mathcal{E}_l^t$, the in-scope input is $\mathbf{KEPrompt}(\mathcal{E}^*, q_l)$ with the target a_l. We also include an out-of-scope input q_l with the target a_l that is without any retrieved edits.

In this way, the LLM learns to leverage retrieved edits for relevant queries while keeping the answer unchanged for locality queries.

B Details of LLM Alignment

Hyperparameter	Value
Batch size	1
Gradient accumulation steps	8
Learning rate	1e-4
Epoches	1
Max length	2048
Optimizer	AdamW
Scheduler	cosine
Warmup ratio	0.1
LoRA rank	8
LoRA alpha	16
LoRA dropout	0

Table 8: Hyper-parameters for LLM finetuning.

We adopt LLaMA-Factory (Zheng et al., 2024) to finetune the LLM. The alignment is done on an NVIDIA A40 48GB GPU and takes 2 hours.

The main training hyperparameters are presented in Table 8, most remain unchanged as the default values. Note that we train the LLM for only one epoch because finetuning for multiple epoches will cause over-fitting.

C Details of Diverse Edit Augmentation

The prompt templates of diverse edit augmentation for `gpt-4o-mini` are presented in Figure 2 and 3. We include manually written in-context examples in the templates to better instruct the model.

D Details of Training Data for the Filter

For each instance of CRAFT's training set $\mathcal{E}^t$ with $\mathcal{E}_e^t$, $\mathcal{E}_p^t$ and $\mathcal{E}_l^t$ as the set of edits, portability QAs

You are a helpful assistant. You are given a query, a target new, and a declaration. Please generate a paraphrased sentence of the declaration with the central term translated to English.

Here is an example:
Query: 2024年华大智造的总资产（亿元）是多少？
Target new: 103.15
Declaration: 2024年华大智造的总资产（亿元）是103.15。
Paraphrased sentence: 2024年华大智造的Total Assets(100 million yuan)是103.15。

Here is another example:
Query: 2024年9月中国的订销报纸份数当期值(万份)是多少？
Target new: 137885.2
Declaration: 2024年9月中国的订销报纸份数当期值(万份)是137885.2。
Paraphrased sentence: 2024年9月中国的Issue of Newspapers, Current Period Value(10000 pieces)是137885.2。

Please generate a paraphrased sentence of the declaration below:
Query: {query}
Target new: {target_new}
Declaration: {declaration}
Paraphrased sentence :

Figure 3: Prompt template for the *aliased* form.

and locality QAs respectively, we simply construct the training data for the filter based on relevance. For any two edits $e = (q_e, a_e), e' = (q_{e'}, a_{e'})$ (may be identical) in $\mathcal{E}_e^t$, the input is $q_{e'}$ [sep] e and the target is 1 (relevant). For any edit $e \in \mathcal{E}_e^t$ and any portability QA pair $(q_p, a_p) \in \mathcal{E}_p^t$, the input is q_p [sep] e and the target is 1 (relevant). For any edit $e \in \mathcal{E}_e^t$ and any locality QA pair $(q_l, a_l) \in \mathcal{E}_l^t$, the input is q_l [sep] e and the target is 0 (irrelevant). In this way, the filter learns to decide the relevance between the query and the edit.

E Example from CRAFT-Statistics

- **Prompt 1:** 2024年7月中国的货币和准货币(M2)供应量期末值(亿元)是多少？ *(English translation: What is the period-end value (100 million yuan) of China's Money and Quasi-Money (M2) supply in July 2024?)* **Target New:** 3033060.78
- **Prompt 2:** 2023年7月中国的货币和准货币(M2)供应量期末值(亿元)是多少？ *(English translation: What is the period-end value (100 million yuan) of China's Money and Quasi-Money (M2) supply in July 2023?)* **Target New:** 2854031.56

Portability

- **Alias:**
  - Prompt: 2024年7月 中 国 的Money and Quasi-Money (M2) Supply, period-end(100 million yuan)是多少？ *(English translation: What is the Money and Quasi-Money (M2) Supply, period-end (100 million yuan), in July 2024 in China?)* Answer: 3033060.78
  - Prompt: 2023年7月 中 国 的Money and Quasi-Money (M2) Supply, period-end(100 million yuan)是多少？ *(English translation: What is the Money and Quasi-Money (M2) Supply, period-end (100 million yuan), in July 2023 in China?)* Answer: 2854031.56
- **Composite Reasoning:** Prompt: 2024年7月中国的货币和准货币(M2)供应量期末值(亿元)比2023年7月的高多少？ *(English translation: How much higher was China's M2 supply (100 million yuan) in July 2024 than in July 2023?)* Answer: 179029.22

Locality

- **Temporal:**
  - Prompt: 2022年7月中国的货币和准货币(M2)供应量期末值(亿元)是？ *(English translation: What was the M2 supply (100 million yuan) at the end of July 2022 in China?)* Answer: 2578078.57
  - Prompt: 2025年7月中国的货币和准货币(M2)供应量期末值(亿元)是？ *(English translation: What is the M2 supply (100 million yuan) at the end of July 2025 in China?)* Answer: 3299429.06
- **Common-sense:**
  - Prompt: 请根据以下材料回答问题。（略）问题：动物的器官感觉与人的相比有什么不同？ *(English translation: Read the following passage and answer the question. Question: How do animals' sensory organs differ from those of humans?)* Answer: D
  - Prompt: 请根据以下材料回答问题。（略）问题：录音中提到能预报风暴的动物是什么？ *(English translation: Read the following passage and answer the question. Question: Which animal mentioned in the passage can predict storms?)* Answer: C

F Example from CRAFT-Finance

- **Prompt 1:** 2024年三友医疗的总资产（亿元）是多少？ *(English translation: What is the total assets (100 million yuan) of Sanyou Medical in 2024?)* **Target New:** 23.07

- **Prompt 2:** 2024年三友医疗的总负债（亿元）是多少？ *(English translation: What is the total liabilities (100 million yuan) of Sanyou Medical in 2024?)* **Target New:** 2.58

Portability

- **Alias:**
  - Prompt: 2024年三友医疗的Total Assets(100 million yuan)是多少？ *(English translation: What is the total assets (100 million yuan) of Sanyou Medical in 2024?)* Answer: 23.07
  - Prompt: 2024年三友医疗的Total Liabilities(100 million yuan)是多少？ *(English translation: What is the total liabilities (100 million yuan) of Sanyou Medical in 2024?)* Answer: 2.58
- **Composite Reasoning:** Prompt: 2024年三友医疗的资产负债率（%）是多少？ *(English translation: What is the debt-to-asset ratio (%) of Sanyou Medical in 2024?)*

  Answer: 11.2

Locality

- **Temporal:**
  - Prompt: 2023年三友医疗的总资产（亿元）是多少？ *(English translation: What was the total assets (100 million yuan) of Sanyou Medical in 2023?)* Answer: 22.61
  - Prompt: 2023年三友医疗的总负债（亿元）是多少？ *(English translation: What was the total liabilities (100 million yuan) of Sanyou Medical in 2023?)* Answer: 2.19
- **Common-sense:**
  - Prompt: 请根据以下材料回答问题。（略）问题：小羊的结局是什么？ *(English translation: Read the following passage and answer the question. Question: What happened to the little lamb?)* Answer: C
  - Prompt: 请根据以下材料回答问题。（略）问题：这个故事告诉我们什么道理？ *(English translation: Read the following passage and answer the question. Question: What moral lesson does this story convey?)* Answer: A

G Results on Traditional Knowledge Editing Benchmarks

We also evaluate KEDAS and representative methods on the four widely used traditional datasets from the KnowEdit benchmark (Zhang et al., 2024), including ZsRE (Levy et al., 2017), WikiBio (Manakul et al., 2023), $\text{WikiData}_{\text{recent}}$ and $\text{WikiData}_{\text{counterfact}}$ (Cohen et al., 2024b). Note that in this experiment, the training data for LLM alignment (including LTE and KEDAS) and the filter are constructed based on the training sets of these datasets. We also modify the diverse edit augmentation process and add two types of augmentation namely *paraphrased* and *reversed* (reversing the relationship) to better fit in the datasets.

The results are presented in Table 9. KEDAS secures the highest harmonic mean scores of edit success, locality and portability on all the four datasets, outperforming previous methods significantly. This further confirms that besides CRAFT, KEDAS is also effective on traditional knowledge editing benchmarks.

Method	ZsRE				WikiBio			WikiData$_{recent}$				WikiData$_{counteract}$			
	ES	**L**	**P**	**HM**	**ES**	**L**	**HM**	**ES**	**L**	**P**	**HM**	**ES**	**L**	**P**	**HM**
IKE	98.5	43.5	64.4	61.7	96.9	37.9	54.5	97.4	48.8	67.2	65.7	91.3	53.7	61.4	65.4
EREN	32.3	70.7	44.9	44.5	59.4	64.3	61.8	44.4	73.1	37.5	47.7	23.5	**75.4**	22.9	30.2
WISE	60.0	51.3	40.4	49.3	84.5	95.9	89.8	68.8	**94.5**	43.1	62.1	47.1	45.4	35.0	41.8
LTE	**99.6**	55.5	66.7	69.7	**97.7**	59.7	74.1	**99.8**	52.9	74.7	70.9	**98.3**	59.3	73.4	73.8
KEDAS	**99.6**	**90.5**	**71.4**	**85.5**	**97.7**	**100.0**	**98.9**	**99.8**	73.9	**76.1**	**81.7**	**98.3**	69.2	**77.4**	**79.9**

Table 9: Results on traditional knowledge editing benchmarks. Metrics include edit success (ES), locality (L), portability (P) and harmonic mean (HM) of the previous metrics. The highest scores are **bolded**.