

000 001 002 003 004 005 MEDREK: RETRIEVAL-BASED EDITING FOR MEDICAL 006 LLMS WITH KEY-AWARE PROMPTS 007 008 009

010 **Anonymous authors**
011 Paper under double-blind review
012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027

ABSTRACT

028 LLMs hold great promise for healthcare applications, but the rapid evolution of
029 medical knowledge and errors in training data often cause them to generate outdated
030 or inaccurate information, limiting their applicability in high-stakes clinical practice.
031 Model editing has emerged as a potential remedy without full retraining. While
032 parameter-based editing often compromises locality and is thus ill-suited for the
033 medical domain, retrieval-based editing offers a more viable alternative. However,
034 it still faces two critical challenges: (1) representation overlap within the medical
035 knowledge space often causes inaccurate retrieval and reduces editing accuracy; (2)
036 existing methods are restricted to single-sample edits, while batch-editing remains
037 largely unexplored despite its importance for real-world medical applications. To
038 address these challenges, we first construct MedVersa, an enhanced benchmark with
039 broader coverage of medical subjects, designed to evaluate both single and batch
040 edits under strict locality constraints. We then propose MedREK, a retrieval-based
041 editing framework that integrates a shared query-key module for precise matching
042 with an attention-based prompt encoder for informative guidance. Experimental
043 results on various medical benchmarks demonstrate that our MedREK achieves
044 superior performance across different core metrics and provides the first validated
045 solution for batch-editing in medical LLMs.
046

047 1 INTRODUCTION 048

049 The remarkable success of large language models (LLMs) (Touvron et al., 2023a;b; Bai et al., 2023) in
050 recent years has attracted significant attention from the medical community, leading to the emergence
051 of specialized medical LLMs such as BioGPT (Luo et al., 2022), Med-PaLM (Singhal et al., 2023),
052 ChatDoctor (Li et al., 2023) and PMC-LLaMA (Wu et al., 2024). However, due to the rapid evolution
053 of medical knowledge and limitations in training data, these models may generate inaccurate or
054 even fabricated responses (hallucinations), which can be particularly harmful in real-world medical
055 advising and decision-making scenarios (Tian et al., 2024; He et al., 2025).

056 To address this issue, model editing (Meng et al., 2022; 2023; Wang et al., 2024) has emerged
057 as a promising approach for updating the knowledge of pre-trained LLMs without requiring full
058 retraining. Despite its growing popularity, model editing remains underexplored in the medical
059 domain. A pioneering effort, MedLaSA (Xu et al., 2024), follows the locate-then-edit paradigm and
060 constructs medical benchmarks for evaluating single-edit scenarios. However, its benchmark remains
061 confined to simple single-edit settings and overlooks the more realistic batch-edit scenario. Moreover,
062 locate-then-edit methods, which modify a small subset of parameters, often induce side effects on
063 unrelated knowledge and compromise locality (Fang et al., 2025). This limitation is particularly
064 concerning in medical applications, where reliability and consistency are critical.

065 To enable more realistic evaluation, we introduce MedVersa, the first benchmark designed to explore
066 batch-editing in medical scenarios. It better reflects real-world use cases in which multiple related
067 facts require simultaneous updates. We also substantially broaden subject coverage, yielding a more
068 comprehensive benchmark that facilitates future research on knowledge editing for medical LLMs.
069 Under batch-editing scenarios, we investigate retrieval-based approaches (Huang et al., 2023a; Yu
070 et al., 2024) as an alternative to parameter-based editing, since they store new knowledge in an
071 external memory module without altering the model’s original parameters and thereby better preserve
072 locality. However, our preliminary experiments show that the state-of-the-art retrieval-based method
073

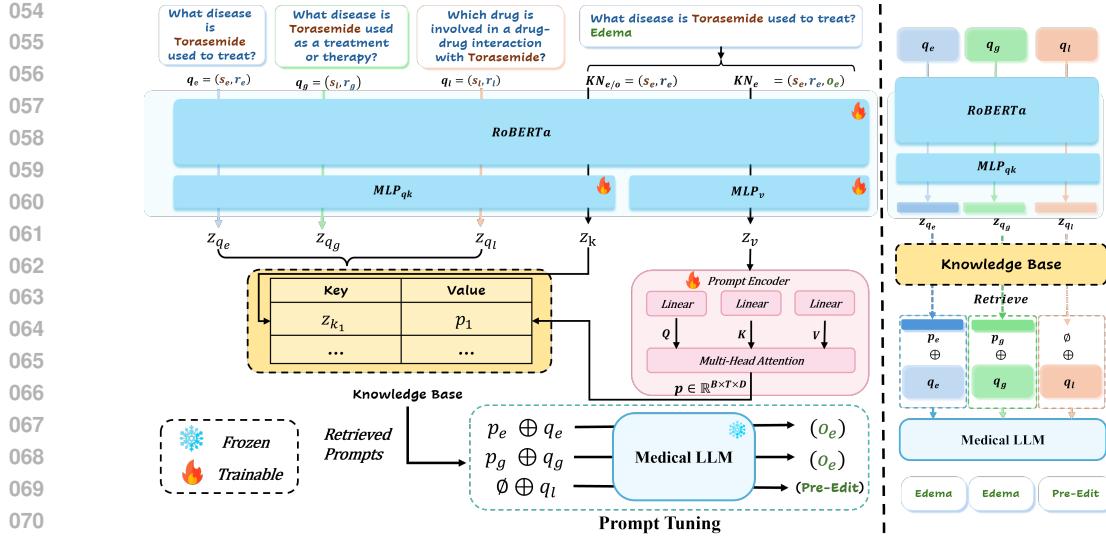


Figure 1: Pipeline of our MedREK. Left: Construction of the knowledge base by encoding medical knowledge into key-value pairs. Right: Inference process where different types of queries are encoded to retrieve relevant knowledge and generate attention-based prompts for precise model editing.

RECIPE (Chen et al., 2024b) often fails to retrieve the correct factual entries on the challenging medical evaluations. Further analysis indicates that the medical domain contains many textually similar but factually distinct knowledge items, which causes *representation overlap* in the retrieval space and thus hinders accurate knowledge matching. Based on these observations, we propose **Medical Retrieval-based Editing with Key-aware prompts (MedREK)**.

As shown in Fig. 1, our method introduces two key components: (1) *a shared query-key MLP*, which unifies the representation space of queries and keys for more precise knowledge retrieval; (2) *an attention-based prompt encoder*, which generates more informative prompts to guide editing. Building on these components, our MedREK achieves strong performance in both single-edit and batch-edit evaluations. Comprehensive experiments on various medical benchmarks further demonstrate that our MedREK achieves state-of-the-art performance across *Efficacy*, *Generality*, and *Locality* metrics, validating the effectiveness of the proposed *shared query-key MLP* and *attention-based prompt encoder*. In summary, our main contributions are as follows:

- We construct MedVersa, a medical factual knowledge editing benchmark that enables a more realistic batch-editing setting and offers broader subject coverage than existing benchmarks.
- We are the first to explore retrieval-based model editing for medical LLMs, proposing two novel components that separately enhance key-query alignment and prompt quality.
- We propose MedREK, a novel method that markedly improves knowledge editing performance by enhancing key-query alignment. Extensive experiments confirm that MedREK achieves state-of-the-art results across core metrics, demonstrating particularly strong gains in locality.

2 RELATED WORKS

Model Editing aims to efficiently update a pre-trained model's behavior in response to new or corrected knowledge, without full retraining or negatively affecting unrelated predictions. Existing methods fall into three categories: locate-then-edit methods (Meng et al., 2022; Li et al., 2024), meta-learning-based strategies (Tan et al., 2024; Mitchell et al., 2022a), and retrieval-based approaches (Wang et al., 2024; Chen et al., 2024b).

- *Locate-then-edit methods*, such as ROME (Meng et al., 2022) and MEMIT (Meng et al., 2023), identify the parameters associated with specific knowledge in LLMs and directly modify them to incorporate new information.
- *Meta-learning-based methods* take a different route by manipulating gradients to perform global parameter updates, aiming for more generalizable knowledge integration. For example, both

108 KE (Cao et al., 2021) and MEND (Mitchell et al., 2022a) employ a lightweight hyper-network
 109 to adjust the updating gradient. InstructEdit (Wang et al., 2023) builds on MEND by introducing
 110 instruction tuning for handling various tasks. Moreover, MALMEN (Tan et al., 2024) further
 111 generalizes MEND to batch-editing under memory constraints.

- 112 • *Retrieval-based approaches* edit model output by storing new knowledge in external memory
 113 modules, without altering pre-trained weights. These modules can take the form of codebooks,
 114 neurons, or auxiliary models, as demonstrated in SERAC (Mitchell et al., 2022b), T-Patcher (Huang
 115 et al., 2023a), GRACE (Hartvigsen et al., 2023), and MELO (Yu et al., 2024). More recently,
 116 WISE (Wang et al., 2024) designs dual memory and routes the model to the side memory in FFN
 117 for editing. RECIPE (Chen et al., 2024b) utilizes continuous prompt learning to prefix knowledge
 118 to the input query and dynamically retrieves knowledge from the knowledge base.

119 Accurate and up-to-date medical knowledge is critical for the safe and reliable application of large
 120 language models in healthcare. As medical facts evolve rapidly with new research and clinical
 121 guidelines, the ability to edit existing knowledge without retraining is essential. Given the importance
 122 of medical knowledge editing, MedLaSA (Xu et al., 2024) pioneers this area and introduces the
 123 MedCF benchmark, but direct model edits can degrade locality and are limited to single-edit protocols.
 124 Building on this medical model editing line, we extend the setting to batch-editing and adopt the
 125 retrieval-based strategy. We further identify medical domain retrieval failures and remedy them with
 126 targeted improvements, culminating in MedREK, which delivers more reliable knowledge updates.
 127

128 **Prompt Tuning.** As a representative parameter-efficient fine-tuning technique, prompt tuning is
 129 widely used in adapting foundation models to downstream tasks. It includes discrete prompts,
 130 expressed as actual text strings, and continuous prompts, encoded directly within the embedding
 131 space of the language model. Specifically, discrete prompts (Shin et al., 2020; Schick & Schütze,
 132 2021a;b; Gao et al., 2021) are manually designed or automatically searched to elicit desired behavior
 133 from the model, while continuous prompts (Li & Liang, 2021; Liu et al., 2022; 2023; Xu et al.,
 134 2023) are trainable embeddings learned through optimization, capable of capturing more nuanced
 135 task-specific information. In this work, we train a prompt encoder to generate a continuous prompt
 136 that serves as external memory and enables targeted medical knowledge editing. These lightweight
 137 prompts can be stored efficiently, enabling effective editing with minimal resource overhead.

138 3 MOTIVATION

139 3.1 PRELIMINARIES

140 **Model Editing.** Let $f_\theta \in \mathcal{F} : \mathcal{Q} \rightarrow \mathcal{A}$ denote the large language model which can map an input query
 141 $q \in \mathcal{Q}$ to the output answer $a = f_\theta(q)$. Given an edit sample pair (q_e, o_e) that $f_\theta(q_e) \neq o_e$, model
 142 editing hopes to modify f_θ to f'_θ so that,

$$144 \quad f'_\theta = \text{ME}(f_\theta, q_e, o_e), \quad f'_\theta(q_e) = o_e, \quad (1)$$

145 where $\text{ME}(\cdot, \cdot, \cdot)$ means the model editor.

146 For a factual knowledge triple $\text{KN}_e = (s_e, r_e, o_e)$, the components s_e , r_e , and o_e denote the subject,
 147 relation, and object, respectively (Mallen et al., 2023). We define the pair $\text{KN}_{e/o} := (s_e, r_e)$ as the
 148 pre-defined knowledge key, and thus $\text{KN}_e = (\text{KN}_{e/o}, o_e)$.

149 **Evaluation Metrics.** Beyond accurate knowledge editing (*i.e.*, $f'_\theta(q_e) = o_e$), an ideal model editor
 150 should meet a range of additional requirements, each evaluated through specific metrics (Huang et al.,
 151 2023b; Yao et al., 2023; Zhang et al., 2024).

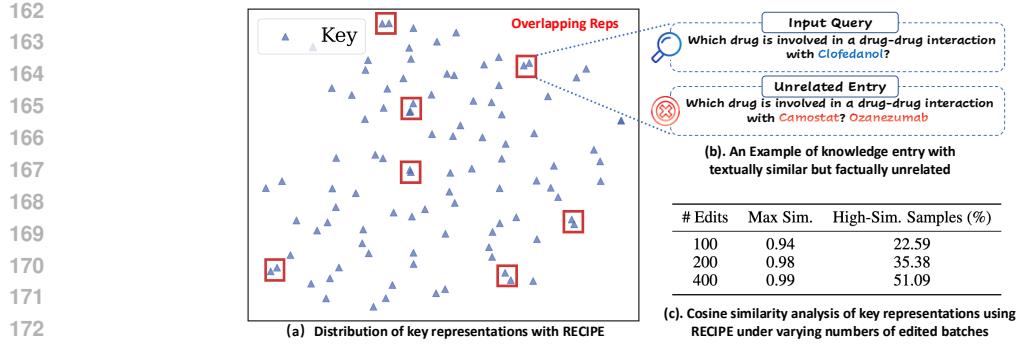
152 **Generality.** The edited model f'_θ is expected to generalize beyond the editing query q_e to correctly
 153 answer similar questions q_g .

$$154 \quad \mathbb{E}_{(q_g, o_e)} \mathbb{I}\{f'_\theta(q_g) = o_e\}, \quad (2)$$

155 **Locality.** The edited model f'_θ should preserve the overall capabilities of the original large language
 156 model (*i.e.*, f_θ) by ensuring that examples unrelated to the editing target remain unaffected.

$$157 \quad \mathbb{E}_{(q_l, o_l)} \mathbb{I}\{f'_\theta(q_l) = f_\theta(q_l) = o_l\}, \quad (3)$$

158 where (q_l, o_l) denotes question-answer pairs that are semantically irrelevant to the edited samples
 159 (q_e, o_e) and should ideally remain unaffected. Note that, given the high precision requirements in
 160 medical domains, this metric becomes particularly critical when performing knowledge editing.



173
174 Figure 2: (a) Visualization of key representations in retrieval-based editing methods. Key overlap
175 raises retrieval errors, decreasing editing accuracy. (b) Example of incorrect retrieval caused by key
176 overlap. (c) Cosine similarity measurements quantifying degree of overlap among key representations.

177 *Efficacy.* This metric calculates the accuracy of the modified model f'_θ on the edited examples (q_e, o_e) ,
178 thereby reflecting the effectiveness of the edit.

$$\mathbb{E}_{(q_e, o_e)} \mathbb{I}\{f'_\theta(q_e) = o_e\}. \quad (4)$$

180 *Fluency.* This metric is designed to reflect the linguistic coherence of the response. It is quantified by
181 computing the weighted average of bi-gram and tri-gram entropies, given by:

$$\sum_k f_n(k) \log_2 f_n(k), \quad (5)$$

185 where $f_n(\cdot)$ is the n -gram frequency distribution, and k denotes a model-generated output.

186 **Editing Setting.** Single-editing refers to modifying one knowledge item at a time, while batch-editing
187 involves applying multiple edits simultaneously. Real-world updates such as revised treatment guide-
188 lines, newly discovered drug interactions, or retracted clinical findings often affect multiple related
189 facts simultaneously. This makes batch-editing a realistic and practically valuable setting. However,
190 within medical domains, current approaches are primarily designed for single-edit scenarios, leaving
191 the more realistic challenge of batch-editing largely unexplored. This further highlights the necessity
192 of more precise retrieval mechanisms.

194 3.2 LOCATE-THEN-EDIT METHODS HARM LOCALITY

196 Locate-then-edit approaches (Meng et al., 2023; 2022) typically use causal tracing to identify
197 influential model components and modify the corresponding parameters. However, such parameter
198 updates often introduce undesirable side effects on unrelated knowledge, leading to a notable drop
199 in locality metrics (Fang et al., 2025). This makes them ill-suited for medical applications, where
200 high locality (i.e., preserving irrelevant knowledge) is essential to ensure reliability and safety. To
201 mitigate these issues, we turn our attention to the alternative retrieval-based methods, which avoid
202 direct parameter modification and generally offer better locality preservation.

203 3.3 RETRIEVAL-BASED METHODS STRUGGLE WITH OVERLAPPING KNOWLEDGE

205 In medical scenarios, many knowledge entries may be textually similar but factually unrelated,
206 making accurate retrieval especially challenging. This calls for a more precise matching mecha-
207 nism between queries and stored knowledge. In particular, competing retrieval-based methods, such
208 as RECIPE (Chen et al., 2024b), fail to retrieve the correct factual entry from the knowledge base on
209 the medical knowledge editing benchmark. As shown in Figure 2 (a), the representation space of
210 RECIPE exhibits significant overlap among distinct knowledge items, leading to frequent confusion.
211 This issue is exemplified in Figure 2 (b): for the query "Which drug is involved in a drug-drug
212 interaction with Clofedanol?", the model incorrectly retrieves the unrelated entry "Which drug is
213 involved in a drug-drug interaction with Camostat? Ozanezumab" from the knowledge base, due to
214 overlapping key representations.

215 To quantify this phenomenon, Figure 2 (c) further measures the overlap among key representations.
Specifically, we report the percentage of unique samples involved in at least one pair with cosine

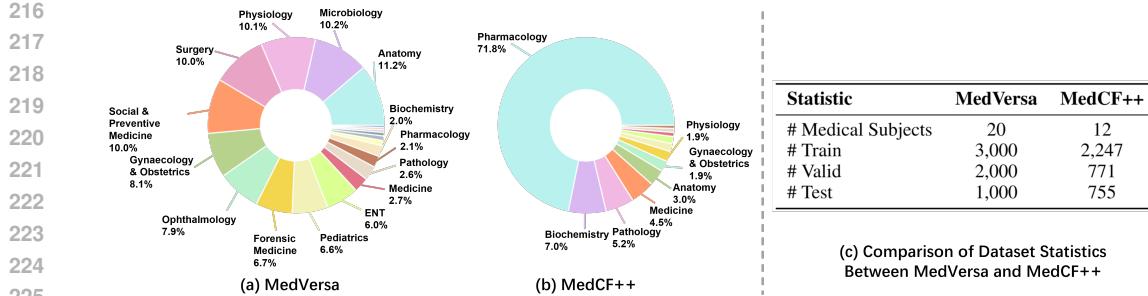


Figure 3: Comparison of MedVersa and MedCF++ in terms of medical subjects and dataset statistics. We provide the full distribution of medical subjects in the two datasets in Appendix.

similarity > 0.6 , denoted as “High-Sim. Samples (%)”, which reflects how many representations are highly aligned with others in the same batch. It can be observed that, as batch size increases, the share of samples participating in at least one high-similarity pair rises, reaching 51.09%, indicating greater representation overlap in larger batches. This provides quantitative support for our motivation and underscores the need for more precise retrieval tailored to medical knowledge editing.

4 METHOD

To address the limitations discussed above, we first construct MedVersa (*i.e.*, **Medical Versatile Knowledge Editing Dataset**), an enhanced benchmark with broader coverage of medical subjects, designed to systematically examine batch-edit scenarios in the medical domain. We then present the **MedREK** algorithm (*i.e.*, **Medical Retrieval-based Editing with Key-aware prompts**), designed to mitigate knowledge forgetting from direct parameter updates and effectively tackle the issue of irrelevant retrievals highlighted in Figure 2.

4.1 CONSTRUCTION OF MEDVERSA DATASET

The current MedCF (Xu et al., 2024) benchmark—built on the Drug Repurposing Knowledge Graph (DRKG) that links compounds, diseases, biological processes, side effects, and symptoms—presents two limitations: (i) Some prompts are reused for both Efficacy and Locality across the train/validation/test

splits, which confounds batch-edit evaluation because edits targeting the Efficacy answer should not influence unrelated Locality responses. (ii) MedCF (Xu et al., 2024) remains limited in its topical breadth: most selected entities come from drugs and compounds, concentrating the benchmark in pharmacology. To address these issues, we construct MedVersa built on MedMCQA (Pal et al., 2022), which eliminates Efficacy–Locality prompt overlap to enable reliable batch-edit assessment and broadens coverage across medical subjects for more comprehensive evaluation. For clarity, the intermediate version that fixes only limitation (i) is denoted as MedCF++. An illustrative example is shown in Table 1, and additional construction details are provided in Appendix B and C.

Comparison of MedVersa and MedCF++. In Figure 3, we present a detailed comparison of MedVersa and MedCF++. MedVersa spans a broader range of 20 medical subjects, including areas absent in MedCF++ such as Pediatrics and Social & Preventive Medicine. In contrast, MedCF++ covers only 12 medical subjects, with Pharmacology accounting for the majority (71.8%). The broader and more balanced subject coverage of MedVersa fills the gap in evaluating medical knowledge editing across a wider range of medical domains, enabling more comprehensive assessment.

4.2 THE PROPOSED MEDREK ALGORITHM

As shown in Figure 1, our proposed MedREK comprises two main components: ① a representation model featuring a unique *shared query-key MLP* (*i.e.*, MLP_{qk}) designed for effective knowledge retrieval, and ② an *attention-based prompt encoder* for generating more informative prompts.

270 4.2.1 THE SHARED QUERY-KEY MLP
271

272 To ensure precise retrieval in model editing, we design a retrieval mechanism that satisfies both
273 relevance and selectivity. Specifically, it aims to (1) accurately retrieve the specific knowledge
274 associated with the given query, which reflects the Efficacy and Generality criteria, and (2) avoid
275 unnecessary retrieval when the query is unrelated to stored knowledge, which corresponds to the
276 locality requirement. To this end, we employ a *shared query-key MLP* encoder that encodes both
277 memory keys and incoming queries into a unified representation space. Importantly, by constructing
278 keys solely from the subject-relation pair (s_i, r_i) instead of the full triplet (s_i, r_i, o_i) , our design
279 avoids incorporating irrelevant object information, thereby reducing retrieval noise.
280

281 Given an input sentence (either in the form of queries q_e, q_g or q_l , knowledge key $\text{KN}_{e/o}$ or a full
282 knowledge triplet KN_e), we first tokenize it using a pre-trained RoBERTa (Liu et al., 2019) tokenizer
283 and obtain contextualized embeddings from the last hidden layer:
284

$$\mathbf{H} \in \mathbb{R}^{L \times d}, \quad \mathbf{p} \in \mathbb{R}^d \quad (6)$$

285 where L is the sequence length and d is the hidden dimension. We aggregate these embeddings by
286 concatenating the CLS token embedding \mathbf{p} with pooled statistics (mean, max, and min):
287

$$\mathbf{x} = [\mathbf{p}; \text{mean}(\mathbf{H}); \text{max}(\mathbf{H}); \text{min}(\mathbf{H})] \in \mathbb{R}^{4d}. \quad (7)$$

288 Next, this vector \mathbf{x} is fed into the representation model’s MLP layers. For queries q_e, q_g or q_l and
289 knowledge key $\text{KN}_{e/o}$, we apply a shared MLP encoder MLP_{qk} :
290

$$\mathbf{z}_q = \text{MLP}_{qk}(\mathbf{x}) = \text{ReLU}(\mathbf{W}_{q2}(\mathbf{W}_{q1}\mathbf{x})) + \mathbf{W}_{q1}\mathbf{x}, \quad (8)$$

$$\mathbf{z}_k = \text{MLP}_{qk}(\mathbf{x}) = \text{ReLU}(\mathbf{W}_{q2}(\mathbf{W}_{q1}\mathbf{x})) + \mathbf{W}_{q1}\mathbf{x}, \quad (9)$$

293 which projects them into a shared representation space for accurate matching in the retrieval stage.
294 For full knowledge triplets KN_e used in prompt generation, a separate MLP encoder MLP_v is applied:
295

$$\mathbf{z}_v = \text{MLP}_v(\mathbf{x}) = \text{ReLU}(\mathbf{W}_{k2}(\mathbf{W}_{k1}\mathbf{x})) + \mathbf{W}_{k1}\mathbf{x}, \quad (10)$$

297 where \mathbf{z}_k is stored as the key of the (k, v) pair in the knowledge base, while \mathbf{z}_v is further passed to
298 the prompt encoder to generate continuous prompt as discussed next.
299

300 4.2.2 ATTENTION-BASED PROMPT ENCODER
301

302 To enable precise edits, we further design a prompt encoder that maps each knowledge representation
303 into a sequence of continuous prompt tokens via multi-head attention. Unlike traditional prompt
304 tuning with fixed prompts for broad tasks, medical knowledge editing demands fine-grained, fact-
305 specific modifications. Given the subtle differences between medical facts, dynamically generating
306 prompts conditioned on the input knowledge is essential. Our prompt encoder learns to produce
307 knowledge-specific prompts, leading to more accurate and effective editing. Specifically, given the
308 knowledge representation $\mathbf{z}_v \in \mathbb{R}^{d_{in}}$ obtained from the representation model, we first project it into a
309 set of query vectors for prompt tokens, and a single key and value vector:
310

$$\mathbf{Q} = \text{reshape}(\mathbf{W}_q \mathbf{z}_v) \in \mathbb{R}^{T \times d}, \quad \mathbf{K} = \mathbf{W}_k \mathbf{z}_v \in \mathbb{R}^{d \times 1}, \quad \mathbf{V} = \mathbf{W}_v \mathbf{z}_v \in \mathbb{R}^{d \times 1},$$

311 where T is the number of prompt tokens, d_{in} and d denote input and output dimensions, and
312 $\mathbf{W}_q \in \mathbb{R}^{Td \times d_{in}}$, $\mathbf{W}_k, \mathbf{W}_v \in \mathbb{R}^{d \times d_{in}}$ are learnable parameters in the attention mechanism, following
313 the standard formulation in (Vaswani et al., 2017). Next, a multi-head attention module is used to
314 allow each prompt token to attend to the same knowledge vector and obtain contextualized prompt
315 representations:
316

$$\mathbf{p} = \text{MultiHeadAttn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \in \mathbb{R}^{T \times d}. \quad (11)$$

317 In this way, $\mathbf{p}_{\text{final}} \in \mathbb{R}^{B \times T \times d}$, which is stored as the value of the (k, v) pair in the knowledge base.
318 The number of continuous prompt tokens T is a hyper-parameter, and we describe its selection for
319 different datasets in Section 5.1.
320

321 4.3 RETRIEVAL PIPELINE AND TRAINING
322

323 **Retrieval Pipeline.** We employ a trainable knowledge prototype representation \mathbf{z}_{pt} as a dynamic
324 threshold for retrieval in the representation model. During inference, retrieval is performed before the
325 query tokens are fed into the embedding layer:
326

324 Table 2: The overall results for single-editing (# Editing=1) and batch-editing (# Editing>1) using
 325 Meditron-7B on MedCF++ and MedVersa datasets.

# Editing	Method	MedCF++								MedVersa													
		Eff.		Gen.		Loc.			Flu.		Avg.			Eff.		Gen.		Loc.		Flu.		Avg.	
		TD	EM	SS	TS	Flu.	Avg.	TD	EM	SS	TS	Flu.	Avg.	TD	EM	SS	TS	Flu.	Avg.				
1	MEND	26.60	28.14	89.03	90.47	88.73	87.13	575.98	58.10	27.09	28.65	94.24	583.09	61.05									
	MEMIT	79.06	69.23	98.02	77.40	97.65	93.57	562.91	82.90	70.02	46.81	99.42	575.86	78.92									
	MedLaSA	72.16	68.84	80.11	85.13	80.03	79.54	578.48	75.85	74.30	71.57	87.01	555.69	79.97									
	RECIPE	72.66	77.12	92.64	99.80	90.29	90.59	586.68	84.11	57.89	57.44	99.03	599.66	78.35									
	Ours	78.50	80.61	99.42	98.96	99.34	98.67	587.00	89.33	74.49	70.46	100.00	579.63	86.24									
10	MEND	27.48	28.18	76.24	79.57	77.55	76.73	579.21	52.68	28.52	29.85	86.30	585.75	57.74									
	MEMIT	79.24	69.81	94.93	75.99	93.88	89.34	562.82	81.53	66.60	46.25	97.41	574.47	76.92									
	RECIPE	72.09	76.46	89.06	96.07	88.17	87.81	586.29	82.28	57.89	57.41	96.72	600.27	77.18									
	Ours	78.52	80.63	99.33	98.89	99.35	98.51	589.56	89.30	74.49	70.46	99.90	600.17	86.19									
	MEND	25.05	25.80	61.99	61.33	64.78	60.25	578.74	43.88	27.29	28.07	68.97	583.66	48.33									
50	MEMIT	75.59	67.08	90.14	77.04	89.53	85.99	562.07	78.51	68.98	48.51	93.04	573.18	75.89									
	RECIPE	70.33	73.82	78.75	87.20	79.15	76.48	589.75	76.23	57.89	57.44	89.65	599.59	73.66									
	Ours	78.54	80.61	98.55	98.70	98.80	97.87	589.79	89.03	74.49	70.39	99.68	599.25	86.06									
	MEND	24.88	25.14	65.03	62.49	66.24	66.24	578.46	44.52	26.07	26.26	61.43	577.92	43.80									
	MEMIT	76.36	66.42	87.91	76.74	87.26	83.52	562.53	77.62	70.26	49.20	89.73	573.27	74.73									
100	RECIPE	68.30	70.74	72.74	79.73	72.48	71.58	588.21	71.83	57.89	57.37	84.19	600.31	70.91									
	Ours	77.96	79.88	97.76	98.20	97.57	97.19	588.01	88.30	74.49	70.46	99.45	598.67	85.96									

$$f_r(q) = \begin{cases} \mathbf{p}_i, & \text{if } \mathbf{z}_q \cdot \mathbf{z}_{k_i} > \mathbf{z}_q \cdot \mathbf{z}_{pt} \\ \emptyset, & \text{otherwise} \end{cases} \quad (12)$$

where $f_r(\cdot)$ denotes the retrieval process for the query q , and \mathbf{z}_{k_i} is the most similar key representation in the knowledge base. A prompt is retrieved only if it is more similar to the query than the learned prototype. If no prompt is returned, the model proceeds as usual, with inference unaffected.

Training. Following RECIPE (Chen et al., 2024b), we utilize the training loss as $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contra}} + \mathcal{L}_{\text{edit}}$, where $\mathcal{L}_{\text{edit}} = \frac{1}{B} \sum_{i=1}^B \left(\mathcal{L}_{\text{eff}}^{(i)} + \mathcal{L}_{\text{gen}}^{(i)} + \mathcal{L}_{\text{loc}}^{(i)} \right)$. Details of each sub-loss are provided in Appendix E.

5 EXPERIMENTS

In this section, we conduct experiments to answer the following research questions: **RQ1:** Does MedREK outperform strong baseline editors on medical LLMs across core metrics and under batch-editing? **RQ2:** Do the proposed modules contribute significant gains individually and jointly? **RQ3:** Is the retrieval mechanism effective at locating the correct knowledge?

5.1 EXPERIMENTAL SETUP

Settings & Benchmarks. We evaluate model editors under both *Single-editing* and *Batch-editing* settings to comprehensively assess their robustness and generalization capabilities. Experiments are conducted on the improved MedCF++ benchmark and our newly constructed MedVersa dataset. Unlike prior work (Xu et al., 2024), which only considers single-editing, we explore more realistic large-scale update scenarios by evaluating performance under 10/50/100-edit configurations.

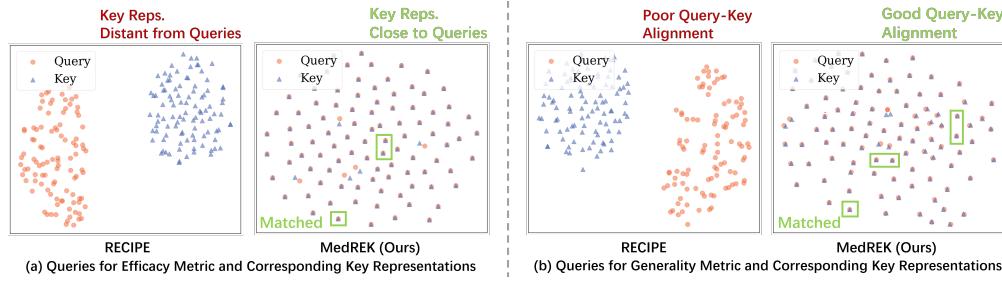
Implementation Details. Following MedLaSA (Xu et al., 2024), we use LLaMA2-based (Touvron et al., 2023c) Meditron-7B (Li et al., 2023) as the primary model and include LLaMA3-based (Grattafiori et al., 2024) HuatuoGPT-o1-8B (Chen et al., 2024a) for additional evaluation. We train MedREK for 200 epochs and report the results using the checkpoint with the smallest loss. We use 3 and 8 prompt tokens for MedCF++ and MedVersa, respectively. See Appendix A for details of hyper-parameters.

Baselines. We compare MedREK with several knowledge editing baselines, including MEND (Mitchell et al., 2022a), MEMIT (Meng et al., 2023), MedLaSA (Xu et al., 2024), and RECIPE (Chen et al., 2024b). Note that MedLaSA (Xu et al., 2024) is excluded from batch-editing experiments, as its parameter modification strategy is inherently designed for single-edit settings and cannot be directly extended to handle multiple simultaneous edits. This limitation underscores the importance of developing batch-capable editing methods. We re-implement RECIPE in the medical domain for fair comparison.

Evaluation metrics. In line with MedLaSA (Xu et al., 2024), we adopt four evaluation metrics: *Efficacy* (Eff.), *Generality* (Gen.), *Locality* (Loc.), *Fluency* (Flu.). For *Locality*, we report the

378 Table 3: The overall results for single-editing (# Editing=1) and batch-editing (# Editing>1) using
379 HuatuoGPT-o1-8B on MedCF++ and MedVersa datasets.

# Editing	Method	MedCF++						MedVersa					
		Eff.	Gen.	Loc.			Flu.	Avg.	Eff.	Gen.	Loc.	Flu.	Avg.
				TD	EM	SS							
1	MEND	17.74	18.23	73.58	73.09	71.28	71.55	629.71	45.18	22.77	24.40	86.90	634.22
	MEMIT	52.92	46.47	95.94	77.45	94.42	92.54	628.60	69.89	62.33	44.43	98.77	632.54
	MedLaSA	61.32	60.98	70.39	75.30	69.65	69.95	626.28	66.24	66.93	65.27	86.43	625.09
	RECIPE	72.98	77.31	91.84	99.64	92.91	91.70	652.46	84.58	51.52	51.54	98.99	633.00
10	Ours	77.05	78.66	99.60	97.90	98.95	98.24	633.11	88.26	69.47	63.43	100.00	634.05
	MEND	15.13	16.18	54.80	57.32	53.05	51.63	630.62	34.93	20.96	22.05	68.76	636.48
	MEMIT	52.09	46.39	89.10	76.47	85.42	83.67	627.76	66.45	63.97	45.25	93.16	631.95
	RECIPE	72.82	76.86	88.80	96.24	90.89	88.02	659.48	82.91	51.52	51.57	96.34	662.74
50	Ours	77.07	78.67	99.31	97.74	98.78	98.07	659.16	88.17	69.47	63.48	99.88	662.68
	MEND	10.72	11.47	31.29	34.16	29.89	31.00	603.95	21.34	16.19	16.72	636.52	31.62
	MEMIT	49.47	44.06	82.71	74.03	77.14	77.88	628.15	62.35	65.66	47.47	81.81	631.25
	RECIPE	72.20	75.90	79.31	87.38	81.36	79.57	659.15	77.98	51.52	51.46	86.56	662.54
100	Ours	77.10	78.63	98.11	97.45	97.60	97.18	659.39	87.72	69.47	63.40	99.62	662.87
	MEND	7.45	7.54	19.69	21.91	19.49	17.87	555.39	13.62	13.62	13.90	32.66	554.89
	MEMIT	51.40	45.77	75.82	72.04	73.90	73.71	627.77	61.23	55.71	45.17	79.54	631.35
	RECIPE	70.73	74.23	72.25	78.62	73.60	72.65	659.67	73.38	51.52	51.35	79.55	663.31
394	Ours	76.42	78.15	96.59	96.34	96.57	96.00	659.51	86.83	69.47	63.28	99.06	663.02



404 Figure 4: Distribution of query and corresponding key representations (i.e., the keys of the $k-v$ pairs
405 in the knowledge base) under a batch of 100 edits using Meditron-7B on the MedCF++ dataset.

406 original four sub-metrics on MedCF++, and a simplified overall score on MedVersa. Definitions of
407 each Loc. sub-metric are provided in Appendix A. The weighted average (Avg.) is computed as in
408 MedLaSA (Xu et al., 2024) to capture the trade-off between editing success (Eff. and Gen.) and Loc.

$$\text{Average} = \left(\frac{\text{Eff.} + \text{Gen.}}{2} + \frac{\sum_{m \in \text{Loc.}} m}{|\text{Loc.}|} \right) / 2. \quad (13)$$

5.2 RESULTS ON MEDCF++ AND MEDVERSA (RQ1)

416 **Single-editing.** From Table 2 and Table 3, we observe that MedREK achieves competitive single-
417 editing performance across both MedCF++ and MedVersa, using different LLM backbones. It
418 outperforms baselines on most sub-metrics, with a notably improved overall average. Interestingly,
419 MedLaSA (Xu et al., 2024) exhibits a significant drop in Locality on both datasets, supporting our
420 earlier claim (Section 3.2) that parameter-modifying methods introduce side effects on unrelated
421 knowledge. In contrast, MedREK performs consistently well across all evaluation setups, highlighting
422 its robustness.

423 **Batch-editing.** Detailed evaluations with 10, 50, and 100 edits are shown in Table 2 and Table 3. We
424 observe that RECIPE (Chen et al., 2024b) suffers a clear performance drop compared to single-editing,
425 with degradation worsening as the number of edits increases. This suggests that overlapping key
426 representations limit its effectiveness under batch-editing scenarios. In contrast, MedREK consistently
427 performs well across all sub-metrics, achieving the best or second-best results, which highlights the
428 strength of our improved representation model and prompt encoder in handling large-scale edits.

429 **Remark.** The experimental results on the original MedCF dataset are provided in Appendix D.

430 5.3 ABATION STUDY (RQ2)

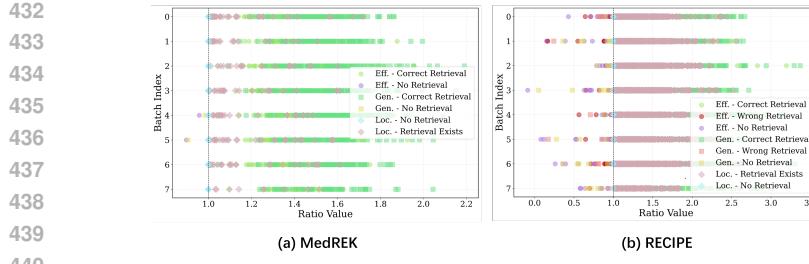


Figure 5: (a-b) Distribution of relative similarity between the query, the ground-truth knowledge, the most similar knowledge and the prototype for test samples in 100-edit batch-editing using Meditron-7B on MedCF++. For Eff. and Gen., x values < 1 indicate incorrect or no retrieval; values > 1 indicate correct retrieval. For Loc., x values $= 1$ indicate no retrieval (locality preserved); values > 1 imply unintended retrieval. (c) Quantitative Results of Retrieval Behavior

To understand the contribution of each proposed component, i.e., the shared query-key MLP and the attention-based prompt encoder, we conduct an ablation study on batch-editing with 100 edits using MedCF++. From Table 4, we observe both components contribute significantly to the final performance of our MedREK. The shared query-key MLP is critical for precise alignment between queries and stored knowledge, enabling more effective retrieval. Without it, performance drops notably due to mismatched representations, highlighting the importance of this component for accurate query-key interaction. The attention mechanism in the prompt encoder also plays a key role by generating high-quality prompts, which further enhances overall editing performance.

5.4 KNOWLEDGE RETRIEVAL EFFECTIVENESS (RQ3)

Obs1: MedREK achieves better retrieval via precise query-key alignment. We analyze the distribution of query representations and key representations from the $k-v$ pairs in one 100-edit batch. As shown in Figure 4, MedREK aligns query and key representations well for both Efficacy and Generality inputs, indicating precise retrieval. In contrast, RECIPE (Chen et al., 2024b) shows more scattered and distant representations, leading to lower scores in Eff. (68.30 vs. 77.96) and Gen. (70.74 vs. 79.88). These observations suggest that RECIPE struggles to retrieve the correct knowledge, while MedREK benefits from more accurate query-key alignment, resulting in better editing success.

Obs2: MedREK achieves more accurate and controlled retrieval. To evaluate retrieval effectiveness, we visualize the distribution of test samples across Eff., Gen., and Loc. metrics in the 100-edit batch setting on MedCF++, along with quantitative retrieval statistics (Figure 5). For Eff. and Gen., the x-axis denotes the ratio between the query’s similarity to ground-truth knowledge vs. to the prototype. Values < 1 indicate incorrect or no retrieval. Samples are categorized as correct (green), incorrect, or no retrieval. MedREK achieves a significantly higher rate of correct retrievals than RECIPE (Chen et al., 2024b). For Loc., the x-axis represents the ratio between the query’s similarity to the most similar knowledge entry (including prototype) and the prototype itself. Values > 1 indicate unintended retrieval of real knowledge, which harms locality. MedREK shows fewer such cases, indicating better locality preservation. Quantitative results further confirm these observations: MedREK retrieves correct knowledge in nearly all Eff. and Gen. cases while avoiding unwanted retrievals in Loc., in stark contrast to RECIPE. More details are provided in Appendix A.

6 CONCLUSION

To address the practical challenge of updating clinical knowledge in medical LLMs, we introduce MedVersa, a benchmark for batch-wise model editing with broad coverage of medical domains. We further propose MedREK, a retrieval-based editing framework tailored for medical LLMs. By incorporating a shared query-key MLP and an attention-based prompt encoder, MedREK enables precise retrieval and effective knowledge updates. Experiments showcase the superiority of MedREK.

486 ETHICS STATEMENT
487488 This research follows the ICLR Code of Ethics. Our work focuses on developing method for
489 knowledge editing in medical LLMs. The primary goal is to improve the adaptability and reliability
490 of LLMs when incorporating updated or corrected medical information, thereby supporting more
491 accurate and trustworthy downstream applications.
492493 REPRODUCIBILITY STATEMENT
494495 To support reproducibility, we provide a complete description of our proposed MedREK in Section 4.2,
496 with training and evaluation setups detailed in Section 5.1. Hyper-parameter settings are included in
497 Appendix A. Dataset details are described in Section 4.1 and Appendix C.
498499 REFERENCES
500501 Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge,
502 Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.503 Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. 2021.
504 URL <https://arxiv.org/abs/2104.08164>.
505506 Junying Chen, Zhenyang Cai, Ke Ji, Xidong Wang, Wanlong Liu, Rongsheng Wang, Jianye Hou,
507 and Benyou Wang. Huatuogpt-0.1, towards medical complex reasoning with llms, 2024a. URL
508 <https://arxiv.org/abs/2412.18925>.509 Qizhou Chen, Taolin Zhang, Xiaofeng He, Dongyang Li, Chengyu Wang, Longtao Huang, and
510 Hui Xue. Lifelong knowledge editing for llms with retrieval-augmented continuous prompt
511 learning. In *EMNLP*, 2024b. URL https://2024.emnlp.org/program/accepted_main_conference/.
512513 Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He,
514 and Tat-Seng Chua. Alphaedit: Null-space constrained model editing for language models. In
515 *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=HvSytv3Jh>.
516517 Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot
518 learners. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the*
519 *59th Annual Meeting of the Association for Computational Linguistics and the 11th International*
520 *Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3816–3830,
521 Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.
522 295. URL [https://aclanthology.org/2021.acl-long.295/](https://aclanthology.org/2021.acl-long.295).
523524 Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad
525 Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan,
526 Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev,
527 Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru,
528 Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak,
529 Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu,
530 Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle
531 Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego
532 Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova,
533 Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel
534 Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon,
535 Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan
536 Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet,
537 Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde,
538 Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie
539 Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua
Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak,
Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley

540 Chiu, Kunal Bhalla, Kushal Lakhota, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence
 541 Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas
 542 Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri,
 543 Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie
 544 Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes
 545 Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne,
 546 Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal
 547 Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong,
 548 Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic,
 549 Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie
 550 Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana
 551 Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie,
 552 Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon
 553 Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan,
 554 Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas
 555 Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami,
 556 Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti,
 557 Vitor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier
 558 Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao
 559 Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song,
 560 Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe
 561 Papakipos, Aditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya
 562 Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei
 563 Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu,
 564 Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit
 565 Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury,
 566 Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer,
 567 Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu,
 568 Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido,
 569 Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu
 570 Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer,
 571 Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu,
 572 Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc
 573 Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily
 574 Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers,
 575 Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank
 576 Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee,
 577 Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan,
 578 Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph,
 579 Helen Suk, Henry Aspegen, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog,
 580 Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James
 581 Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny
 582 Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings,
 583 Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai
 584 Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik
 585 Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle
 586 Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng
 587 Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madien Khabsa, Manav Avalani, Manish
 588 Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim
 589 Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle
 590 Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang,
 591 Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam,
 592 Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier,
 593 Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia
 Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin
 Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu,

- 594 Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh
 595 Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay,
 596 Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang,
 597 Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie
 598 Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta,
 599 Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman,
 600 Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun
 601 Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria
 602 Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru,
 603 Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz,
 604 Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv
 605 Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi,
 606 Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait,
 607 Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The
 608 llama 3 herd of models, 2024. URL <https://arxiv.org/abs/2407.21783>.
- 609 Thomas Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi.
 610 Aging with GRACE: Lifelong model editing with discrete key-value adaptors. In *Thirty-seventh*
 611 *Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=Oc1SIKxwdV>.
- 612
- 613 Kai He, Rui Mao, Qika Lin, Yucheng Ruan, Xiang Lan, Mengling Feng, and Erik Cambria. A survey
 614 of large language models for healthcare: from data, technology, and applications to accountability
 615 and ethics. *Information Fusion*, 118:102963, 2025. ISSN 1566-2535. doi: <https://doi.org/10.1016/j.inffus.2025.102963>. URL <https://www.sciencedirect.com/science/article/pii/S1566253525000363>.
- 614
- 615 Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang,
 616 and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International*
 617 *Conference on Learning Representations*, 2022. URL <https://openreview.net/forum?id=nZeVKeeFYf9>.
- 618
- 619 Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. Transformer-
 620 patcher: One mistake worth one neuron. *arXiv preprint arXiv:2301.09785*, 2023a.
- 621
- 622 Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. Transformer-
 623 patcher: One mistake worth one neuron. In *The Eleventh International Conference on Learning*
 624 *Representations*, 2023b. URL <https://openreview.net/forum?id=4oYUGeGBPm>.
- 625
- 626 Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation.
 627 In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th*
 628 *Annual Meeting of the Association for Computational Linguistics and the 11th International Joint*
 629 *Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, Online,
 630 August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353.
 631 URL <https://aclanthology.org/2021.acl-long.353/>.
- 632
- 633 Xiaopeng Li, Shasha Li, Shezheng Song, Jing Yang, Jun Ma, and Jie Yu. Pmet: Precise model editing
 634 in a transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp.
 635 18564–18572, 2024.
- 636
- 637 Yunxiang Li, Zihan Li, Kai Zhang, Rui long Dan, Steve Jiang, and You Zhang. Chatdoctor: A medical
 638 chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge.
 639 *Cureus*, 15(6), 2023.
- 640
- 641 Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning:
 642 Prompt tuning can be comparable to fine-tuning across scales and tasks. In Smaranda Muresan,
 643 Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the*
 644 *Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 61–68, Dublin, Ireland,
 645 May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.8. URL
 646 <https://aclanthology.org/2022.acl-short.8/>.

- 648 Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt
 649 understands, too, 2023. URL <https://arxiv.org/abs/2103.10385>.
- 650
- 651 Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike
 652 Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining
 653 approach, 2019. URL <https://arxiv.org/abs/1907.11692>.
- 654 Renqian Luo, Lai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu.
 655 Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings*
 656 in *Bioinformatics*, 23(6):bbac409, 09 2022. ISSN 1477-4054. doi: 10.1093/bib/bbac409. URL
 657 <https://doi.org/10.1093/bib/bbac409>.
- 658 Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi.
 659 When not to trust language models: Investigating effectiveness of parametric and non-parametric
 660 memories. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the*
 661 *61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,
 662 pp. 9802–9822, Toronto, Canada, July 2023. Association for Computational Linguistics. doi:
 663 10.18653/v1/2023.acl-long.546. URL <https://aclanthology.org/2023.acl-long.546/>.
- 664
- 665 Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual
 666 associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.
- 667
- 668 Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. Mass-editing
 669 memory in a transformer. In *The Eleventh International Conference on Learning Representations*,
 670 2023. URL <https://openreview.net/forum?id=MkbcAH1YgyS>.
- 671 Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. Fast
 672 model editing at scale. In *International Conference on Learning Representations*, 2022a. URL
 673 <https://openreview.net/pdf?id=0DcZxeWfOPT>.
- 674
- 675 Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Memory-
 676 based model editing at scale. In *International Conference on Machine Learning*, 2022b. URL
 677 <https://arxiv.org/pdf/2206.06520.pdf>.
- 678 Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-
 679 subject multi-choice dataset for medical domain question answering. In Gerardo Flores, George H
 680 Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann (eds.), *Proceedings of the Conference*
 681 *on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*,
 682 pp. 248–260. PMLR, 07–08 Apr 2022. URL <https://proceedings.mlr.press/v174/pal22a.html>.
- 683
- 684 Timo Schick and Hinrich Schütze. Exploiting cloze-questions for few-shot text classification and
 685 natural language inference. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty (eds.), *Proceedings*
 686 of the 16th Conference of the European Chapter of the Association for Computational Linguistics:
 687 *Main Volume*, pp. 255–269, Online, April 2021a. Association for Computational Linguistics. doi:
 688 10.18653/v1/2021.eacl-main.20. URL [https://aclanthology.org/2021.eacl-main.20/](https://aclanthology.org/2021.eacl-main.20).
- 689
- 690 Timo Schick and Hinrich Schütze. It's not just size that matters: Small language models are
 691 also few-shot learners. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek
 692 Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao
 693 Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association*
 694 *for Computational Linguistics: Human Language Technologies*, pp. 2339–2352, Online, June
 695 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.185. URL
 696 [https://aclanthology.org/2021.naacl-main.185/](https://aclanthology.org/2021.naacl-main.185).
- 697
- 698 Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. AutoPrompt:
 699 Eliciting Knowledge from Language Models with Automatically Generated Prompts. In Bonnie
 700 Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on*
 701 *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4222–4235, Online, November
 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.346. URL
[https://aclanthology.org/2020.emnlp-main.346/](https://aclanthology.org/2020.emnlp-main.346).

- 702 Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan
 703 Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne,
 704 Paul Gamble, Chris Kelly, Abubakr Babiker, Nathanael Schärli, Aakanksha Chowdhery, Philip
 705 Mansfield, Dina Demner-Fushman, Blaise Agüera y Arcas, Dale Webster, Greg S. Corrado, Yossi
 706 Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral,
 707 Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. Large language models encode
 708 clinical knowledge. *Nature*, 620(7972):172–180, Aug 2023. ISSN 1476-4687. doi: 10.1038/
 709 s41586-023-06291-2. URL <https://doi.org/10.1038/s41586-023-06291-2>.
- 710 Chenmien Tan, Ge Zhang, and Jie Fu. Massive editing for large language models via meta learning.
 711 In *International Conference on Learning Representations*, 2024. URL <https://openreview.net/pdf?id=L6L1CJQ2PE>.
- 712 Shubo Tian, Qiao Jin, Lana Yeganova, Po-Ting Lai, Qingqing Zhu, Xiuying Chen, Yifan Yang, Qingyu
 713 Chen, Won Kim, Donald C Comeau, Rezarta Islamaj, Aadit Kapoor, Xin Gao, and Zhiyong Lu.
 714 Opportunities and challenges for chatgpt and large language models in biomedicine and health.
 715 *Briefings in Bioinformatics*, 25(1):bbad493, 01 2024. ISSN 1477-4054. doi: 10.1093/bib/bbad493.
 716 URL <https://doi.org/10.1093/bib/bbad493>.
- 717 Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée
 718 Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and
 719 efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.
- 720 Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay
 721 Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation
 722 and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.
- 723 Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei,
 724 Niko lay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas
 725 Blecher, Cris tian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes,
 726 Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S.
 727 Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa,
 728 Isabel M. Kloemann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril,
 729 Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar
 730 Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan
 731 Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross
 732 Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang,
 733 Angela Fan, Melissa Hall Melanie Kambadur, Sharan Narang, Aur’elien Rodriguez, Robert Stojnic,
 734 Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models.
 735 *ArXiv*, abs/2307.09288, 2023c. URL <https://api.semanticscholar.org/CorpusID:259950998>.
- 736 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez,
 737 Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von
 738 Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.,
 739 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fb053c1c4a845aa-Paper.pdf.
- 740 Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang,
 741 and Huajun Chen. WISE: Rethinking the knowledge memory for lifelong model editing of large
 742 language models. In *The Thirty-eighth Annual Conference on Neural Information Processing
 743 Systems*, 2024. URL <https://openreview.net/forum?id=VJMYOFJVC2>.
- 744 Qian Wang, Biao Zhang, Michael Birsak, and Peter Wonka. Instructedit: Improving automatic masks
 745 for diffusion-based image editing with user instructions. *arXiv preprint arXiv:2305.18047*, 2023.
- 746 Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Weidi Xie, and Yanfeng Wang. Pmc-llama:
 747 toward building open-source language models for medicine. *Journal of the American Medical
 748 Informatics Association*, 31(9):1833–1843, 04 2024. ISSN 1527-974X. doi: 10.1093/jamia/
 749 ocae045. URL <https://doi.org/10.1093/jamia/ocae045>.

- 756 Derong Xu, Ziheng Zhang, Zhihong Zhu, Zhenxi Lin, Qidong Liu, Xian Wu, Tong Xu, Wanyu
 757 Wang, Yuyang Ye, Xiangyu Zhao, Enhong Chen, and Yefeng Zheng. Editing factual knowledge
 758 and explanatory ability of medical large language models. In *Proceedings of the 33rd ACM*
 759 *International Conference on Information and Knowledge Management*, CIKM '24, pp. 2660–2670,
 760 New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704369. doi:
 761 10.1145/3627673.3679673. URL <https://doi.org/10.1145/3627673.3679673>.
- 762 Ziyun Xu, Chengyu Wang, Minghui Qiu, Fuli Luo, Runxin Xu, Songfang Huang, and Jun Huang.
 763 Making pre-trained language models end-to-end few-shot learners with contrastive prompt tuning.
 764 In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*,
 765 WSDM '23, pp. 438–446, New York, NY, USA, 2023. Association for Computing Machinery.
 766 ISBN 9781450394079. doi: 10.1145/3539597.3570398. URL <https://doi.org/10.1145/3539597.3570398>.
- 767 Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and
 768 Ningyu Zhang. Editing large language models: Problems, methods, and opportunities, 2023. URL
 769 <https://arxiv.org/abs/2305.13172>.
- 770 Lang Yu, Qin Chen, Jie Zhou, and Liang He. Melo: Enhancing model editing with neuron-indexed
 771 dynamic lora. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp.
 772 19449–19457, 2024.
- 773 Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi,
 774 Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu,
 775 Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and
 776 Huajun Chen. A comprehensive study of knowledge editing for large language models, 2024. URL
 777 <https://arxiv.org/abs/2401.01286>.
- 778
 779
 780
 781
 782
 783
 784
 785
 786
 787
 788
 789
 790
 791
 792
 793
 794
 795
 796
 797
 798
 799
 800
 801
 802
 803
 804
 805
 806
 807
 808
 809

APPENDIX

A MORE IMPLEMENTATION DETAILS.

Table 5: Hyper-parameters for MedREK on MedCF++ and MedVersa.

Hyper-parameter	MedCF++	MedVersa
Meditron-7B		
Learning Rate	1×10^{-5}	1×10^{-5}
Batch Size	8	8
Representation Dimension	4096	4096
Model Hidden Size	4096	4096
# Prompt Tokens	3	8
# Knowledge Prototype Tokens	10	10
HuatuoGPT-o1-8B		
Learning Rate	1×10^{-5}	1×10^{-5}
Batch Size	8	8
Representation Dimension	4096	4096
Model Hidden Size	4096	4096
# Prompt Tokens	3	8
# Knowledge Prototype Tokens	10	10

Training Setup. We train our MedREK and RECIPE (Chen et al., 2024b) on the training set of MedCF++ and MedVersa. As we observe the checkpoints after around 150 epochs all exhibit a trend of increasing loss for both methods, we stop our training at epoch 200. The hyper-parameters for training and evaluation are kept the same as the baseline methods, except for the number of continuous prompt tokens, which we tune for RECIPE (Chen et al., 2024b). For RECIPE (Chen et al., 2024b) and MedREK, we both use 3 prompt tokens on MedCF++ and 8 prompt tokens on MedVersa. The complete hyper-parameter settings of training for MedREK are shown in Table 5. All experiments are conducted on NVIDIA RTX 5090 GPUs.

Locality Metrics of MedCF and MedCF++. Following MedLaSA (Xu et al., 2024), the sub-metrics of Locality in MedCF (Xu et al., 2024) and MedCF++ are defined as follows:

- Target Distribution (TD): Does the editing operation alter the probability distribution of the ground truth tokens?
- Entity Mapping (EM): Does the editing operation solely learn the mapping relationship between head and tail entities?
- Structural Similarity (SS): Does the editing operation influence unrelated knowledge with similar graph structures?
- Textual Similarity (TS): Does the editing operation have an impact on unrelated knowledge that contains similar semantic text?

Details for Figure 5. We present the visualization of the distribution of test samples for each metric focusing on the effectiveness of the knowledge retrieval, i.e., whether the correct piece of knowledge is retrieved for Eff. and Gen, and whether the knowledge is “ignored” for Loc. The result is obtained with batch-editing of 100 edits on MedCF++. The y-axis corresponds to the batch index the test samples belonging to.

Recall that the prompt selection criteria for each metric is as follows: For Eff. and Gen., the correct prompt is supposed to be selected. We calculate the similarity between the query and the top-1 similar key of the knowledge entry in the knowledge base, $sim(q, top_1)$, and the similarity between the query and the knowledge prototype, $sim(q, proto)$. If $sim(q, top_1) > sim(q, proto)$, we select the prompt that generates $sim(q, top_1)$. Note that there are three cases here: (1) Correct selection, where the prompt corresponding to $sim(q, top_1)$ is the target knowledge. (2) Wrong selection, where the prompt corresponding to $sim(q, top_1)$ is a piece of unrelated knowledge. (3) No selection, where $sim(q, top_1) = sim(q, proto)$, meaning the most similar entry is the prototype, and no retrieval is

864 triggered. For Loc., no prompt is supposed to be selected. There are two cases: (1) If $sim(q, top_1) = sim(q, proto)$, which is the desired behavior, no prompt will be retrieved. (2) If $sim(q, top_1) > sim(q, proto)$, the query will falsely retrieve a prompt.
 865
 866
 867

868 We calculate the relative magnitude between $sim(q, top_1)$, $sim(q, gt)$ (for Eff. and Gen.) and
 869 $sim(q, proto)$ to obtain the x value. For editing success (Eff. and Gen.), we present cases of correct
 870 retrieval, wrong retrieval, and no retrieval and label the three cases out. For Loc., we do not label
 871 correct or wrong, instead, if $x = 1$, it means the query selects no prompt as desired, and if $x > 1$, it
 872 means the query falsely selects a prompt, which is not desired. For editing success, more samples
 873 labeled as Correct Retrieval (green) means better retrieval performance. For Loc., less samples with x
 874 > 1 means better retrieval.
 875

876 **Analysis of MedLaSA.** We find that MedLaSA (Xu et al., 2024) has the following critical design
 877 flaw and major weaknesses. First, it violates the spirit of knowledge editing since it considers the
 878 rephrase and locality queries for evaluation as known knowledge and applies causal tracing on them
 879 to calculate the corresponding layer-wise scaling factors. When testing, it retrieves the scaling factors
 880 calculated beforehand with the query text as key. In real-world scenarios, input queries are often
 881 unknown or unavailable in advance, making it impractical to apply causal tracing to determine the
 882 scaling factors for the adapters. As a result, effective knowledge editing cannot be achieved in
 883 practice. Second, despite the utilization of scaling factors to control the impact of adapters on each
 884 layer, it does not guarantee precise control. The LoRA (Hu et al., 2022) process may still introduce
 885 unintended intervention in model where no modification is desired, thereby affecting unrelated
 886 knowledge. In contrast, our method transforms the queries into representations and matches them
 887 with the knowledge key representations for prompt retrieval without operations on the queries in
 888 advance, making it practical in real-world scenarios. Additionally, we achieve better locality by
 889 leaving the original model parameters frozen and only retrieving the prompt when necessary, which
 890 is crucial for editing medical LLMs.
 891

890 B CONSTRUCTION OF MEDCF++ DATASET.

891 As mentioned in 4.1, the only existing medical factual knowledge editing benchmark, MedCF (Xu
 892 et al., 2024), suffers from design issues that prevent it from supporting batch-editing, which we
 893 address by proposing an improved version, MedCF++. Specifically, we remove any records in which
 894 the same prompt is used for both Eff. and Loc., as well as any records where Eff. and Loc. share the
 895 same prompt across different data entries. As a result, 181 records are removed from the training set,
 896 47 from the validation set, and 47 from the test set. The cleaned dataset MedCF++ avoids prompt
 897 overlap to ensure reliable evaluation for batch-editing.
 898

900 C CONSTRUCTION OF MEDVERSA DATASET.

901 Although MedCF++ supports batch-editing, it is still limited in its size and range of medical domains,
 902 as most of the entities it selects from the source dataset DRKG are drugs and compounds, causing the
 903 majority of the knowledge to fall under the pharmacology domain. To address these limitations, we
 904 construct the MedVersa dataset derived from MedMCQA (Pal et al., 2022), which allows for broader
 905 coverage of medical knowledge and supports batch-editing.
 906

907 **Efficacy and Generality Data Construction.** The Efficacy QA pair (q_e, o_e) is used to measure the
 908 effectiveness of model editing. We utilize the MedMCQA (Pal et al., 2022) dataset, which contains
 909 medical knowledge in the form of multiple-choice questions, each comprising a correct answer and
 910 three incorrect options. Since the original “question” field in MedMCQA (Pal et al., 2022) is often
 911 expressed as a phrase instead of an interrogative, we first employ Gemini to rewrite it into a clear
 912 question form to ensure linguistic clarity and consistency. To construct the Efficacy QA pair, the
 913 rewritten question is paired with the correct answer as the ground truth, while one incorrect option is
 914 sampled as the counterfactual edit target. For example, as shown in Table 1, the original question in
 915 MedMCQA (Pal et al., 2022) “Treatment of multiple carboxylase deficiency” is reformulated into the
 916 well-formed interrogative “What is the treatment for multiple carboxylase deficiency?”. The original
 917 correct answer “Biotin” is retained as the ground truth, while one incorrect option “Thiamine” is
 918 selected as the edit target. For the Generality QA pair (q_g, o_e) , which evaluates editing effectiveness

918
919
920 Table 6: The prompts for querying Gemini 2.0 Flash.
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971

Prompts for Querying LLMs
Question Transformation:
You are given a phrase and its corresponding answer. Please rewrite the phrase into a clear question.
Input:
Phrase: {original question}
Answer: {correct answer}
Output only the rewritten question.
Question Rephrase:
Please rephrase the following question using precise medical terminology, ensuring that the original meaning is fully preserved.
Question: {question}
Output only the rephrased question.

Table 7: The distribution of medical subjects in MedVersa (%).

Medical Subject	Percentage (%)
Anatomy	11.22
Microbiology	10.18
Physiology	10.08
Surgery	10.02
Social & Preventive Medicine	9.97
Gynaecology & Obstetrics	8.14
Ophthalmology	7.94
Forensic Medicine	6.78
Pediatrics	6.61
ENT	6.02
Medicine	2.71
Pathology	2.62
Pharmacology	2.05
Biochemistry	1.98
Orthopaedics	0.94
Radiology	0.82
Psychiatry	0.64
Anaesthesia	0.47
Dental	0.44
Skin	0.37

on similar questions, we employ Gemini to rephrase the Efficacy question. The prompts for querying Gemini are shown in Table 6.

Locality Data Construction. Locality aims to evaluate whether the edited model preserves unrelated knowledge, and the model is expected to generate the same answer as before editing. To construct the Locality data, we sample a different entry from MedMCQA (Pal et al., 2022) than the one used for Efficacy, but within the same medical subject, leveraging the “subject name” field in MedMCQA (Pal et al., 2022). The original question is then converted into a standard interrogative, with the correct answer as the ground truth for the Locality question. For example, as shown in Table 1, the Locality question “At what age does purposeful movement typically start in infants?” represents a different piece of knowledge from the Efficacy question, while sharing the same medical subject “Pediatrics”. This design assesses the reliability of the model in preserving unrelated but same-domain knowledge after editing.

Comparison of Medical Subjects in MedVersa and MedCF++. Table 7 and Table 8 show the full distribution of medical subjects in MedVersa and MedCF++. Compared with MedCF++, which is dominated by Pharmacology, MedVersa exhibits a more balanced coverage across subjects. In addition, MedVersa includes 8 subjects that are absent in MedCF++. These underscore the broader coverage of medical domains of MedVersa, enabling a more comprehensive evaluation of medical knowledge editing.

972
973
974 Table 8: The distribution of medical subjects in MedCF++ (%).
975
976
977
978
979
980
981
982
983
984
985
986
987

Medical Subject	Percentage (%)
Pharmacology	71.78
Biochemistry	6.96
Pathology	5.24
Medicine	4.48
Anatomy	2.98
Gynaecology & Obstetrics	1.94
Physiology	1.86
Psychiatry	1.44
Ophthalmology	1.31
Skin	0.73
Orthopaedics	0.73
Surgery	0.55

988 Table 9: The overall results for single editing using Meditron-7B on original MedCF dataset.
989

Editing Type	Method	MedCF							
		Eff.	Gen.	Loc.			Flu.	Avg.	
				TD	EM	SS			
Single Editing	FT	65.97	65.36	48.91	50.39	48.13	46.25	327.76	57.04
	LoRA	72.19	71.80	92.29	91.11	91.36	92.42	572.33	81.90
	MEND	22.87	22.93	71.16	71.21	71.03	72.29	428.38	47.16
	ROME	72.69	72.91	92.79	61.80	90.06	86.93	559.82	77.84
	MEMIT	83.10	83.23	95.01	62.62	92.99	90.50	563.31	84.22
	MedLaSA	72.37	71.06	<u>95.71</u>	94.84	<u>95.04</u>	<u>94.90</u>	582.80	83.42
	RECIPE	72.34	75.40	<u>93.63</u>	97.09	<u>92.91</u>	93.91	<u>573.97</u>	<u>84.13</u>
	Ours	<u>77.91</u>	<u>79.83</u>	99.45	<u>96.80</u>	99.36	98.75	586.34	88.73

1000 D MORE EXPERIMENTAL RESULTS.
10011002
1003 **Results of Single Editing on MedCF.** In Table 9, we present the results of single-editing using
1004 Meditron-7B on the original MedCF (Xu et al., 2024) dataset, which contains duplicate prompts for
1005 Efficacy and Locality in a single record. MedREK performs consistently well on it, and obtains better
1006 result especially in Loc.-EM on the cleaned dataset MedCF++ as shown in Table 2. This supports our
1007 earlier claim (Section 4.1) that duplicated prompts in MedCF (Xu et al., 2024) can lead to unreliable
1008 evaluations.1009 **Edit Time of Different Methods.** To compare the efficiency of different methods, we conduct
1010 single-edit experiments using Meditron-7B (Li et al., 2023) on MedVersa and report the average edit
1011 time in Table 10. We observe that MedREK and RECIPE (Chen et al., 2024b)—both retrieval-based
1012 methods—significantly outperform parameter-modifying methods. In particular, MedLaSA (Xu
1013 et al., 2024) is a parameter-modifying method that leverages LoRA (Hu et al., 2022) finetuning to
1014 perform edits. However, it requires 70 epochs of finetuning for each single edit, which results in
1015 substantially longer editing time. This makes it less efficient and impractical for scenarios requiring
1016 rapid or frequent knowledge updates.1017 E TRAINING LOSS OF MEDREK
10181019 Following RECIPE (Chen et al., 2024b), the loss functions are defined as follows:
1020

1021
1022
$$\mathcal{L}_{\text{eff}}^{(i)} = -\log \hat{f}_{-\theta} \left(o^{(i)}_{-e} \mid p^{(i)} \oplus f_{\text{emb}}(q^{(i)}_{-e}) \right) \quad (1)$$

1023

1024
1025
$$\mathcal{L}_{\text{gen}}^{(i)} = -\log \hat{f}_{-\theta} \left(o^{(i)}_{-g} \mid p^{(i)} \oplus f_{\text{emb}}(q^{(i)}_{-g}) \right) \quad (2)$$

1026 Table 10: Average edit time taken across different methods for single-editing using Meditron-7B on
 1027 MedVersa.

Method	Edit Time (s)
MEND	3.301
MEMIT	18.239
MedLaSA	16.787
RECIPE	0.006
MedREK (Ours)	<u>0.012</u>

$$\mathcal{L}_{\text{loc}}^{(i)} = \text{KL} \left(f_{-\theta}(q^{(i)}_l) \parallel \hat{f}_{-\theta} \left(p^{(i)} \oplus f_{\text{emb}}(q^{(i)}_l) \right) \right) \quad (3)$$

1040 where f_{θ} is the large language model (LLM) to be edited, and \hat{f}_{θ} is f_{θ} with the embedding layer f_{emb}
 1041 removed. The editing loss is then defined as $\mathcal{L}_{\text{edit}} = \frac{1}{B} \sum_{i=1}^B \left(\mathcal{L}_{\text{eff}}^{(i)} + \mathcal{L}_{\text{gen}}^{(i)} + \mathcal{L}_{\text{loc}}^{(i)} \right)$. The contrastive
 1042 learning loss for prompt learning are defined as follows:

$$\mathcal{L}_{\text{no}}^{(i)} = \delta \left(\mathbf{z}_{q_e}^{(i)}, \mathbf{z}_v^{(i)}, R \right) + \delta \left(\mathbf{z}_{q_g}^{(i)}, \mathbf{z}_v^{(i)}, R \right) \quad (4)$$

$$\mathcal{L}_{\text{so}}^{(i)} = \delta \left(\mathbf{z}_{q_l}^{(i)}, \mathbf{z}_{pt}, R \right) + \delta \left(\mathbf{z}_{q_e}^{(i)}, \mathbf{z}_{pt}, R \setminus v_i \right) + \delta \left(\mathbf{z}_{q_g}^{(i)}, \mathbf{z}_{pt}, R \setminus v_i \right) \quad (5)$$

$$\mathcal{L}_{\text{contra}} = \frac{1}{b} \sum_{i=1}^b \left(\mathcal{L}_{\text{no}}^{(i)} + \mathcal{L}_{\text{so}}^{(i)} \right) \quad (6)$$

1053 where $R = \{\mathbf{z}_v^{(i)}\}_{i=1}^b \cup \{r_{\Theta}\}$ and $R \setminus \mathbf{z}_v^{(i)} = R \setminus \{\mathbf{z}_v^{(i)}\}$. $\mathbf{z}_v^{(i)}$ is the representation of the editing
 1054 knowledge triple $\text{KN}_e^{(i)}$ transformed through Equation (9) in the paper. The query representations
 1055 $\mathbf{z}_{q_e}^{(i)}$, $\mathbf{z}_{q_g}^{(i)}$, and $\mathbf{z}_{q_l}^{(i)}$ for $q_e^{(i)}$, $q_g^{(i)}$, and $q_l^{(i)}$ are attained via Equation (8) in the paper, respectively. δ is
 1056 the InfoNCE loss [2], formulated as:

$$\delta \left(q, \text{KN}_e^+, \{\text{KN}_e^{(i)}\}_{i=1}^n \right) = -\log \left(\frac{\exp(q \cdot \text{KN}_e^+ / \tau)}{\sum_{i=1}^n \exp(q \cdot \text{KN}_e^{(i)} / \tau)} \right) \quad (7)$$

1057 where τ is the temperature, typically set to 1 by default. Finally, the total training loss is defined as
 1058 $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contra}} + \mathcal{L}_{\text{edit}}$.

F USE OF LLMs

1066 We leverage Gemini 2.0 Flash for constructing the Efficacy, Generality, and Locality data in MedVersa,
 1067 as detailed in C. For Efficacy and Locality questions, the original phrases from MedMCQA (Pal
 1068 et al., 2022) are transformed into interrogatives using Gemini. For Generality questions, we employ
 1069 Gemini to rephrase the corresponding Efficacy questions. The prompts used for querying the LLM
 1070 are provided in Table 6. We took measures to ensure that the rewritten and rephrased questions
 1071 maintained their original meaning. We used ChatGPT solely to improve the fluency and clarity of
 1072 writing. All content was written by the authors, and the model served only as a language assistant.