Embedding Domain Knowledge for Large Language Models via Reinforcement Learning from Augmented Generation

Chaojun Nie[1,2], **Jun Zhou**[1,2*], **Guanxiang Wang**[1,2]
Shisong Wu[3], **Zichen Wang**[1,2]
[1]Laboratory of Speech and Intelligent Information Processing,
Institute of Acoustics, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
[3]China Southern Power Grid Artificial Intelligence Technology Co., Ltd.
{niechaojun, zhoujun, wangguanxiang, wangzichen}@hccl.ioa.ac.cn, wuss@csg.cn

Abstract

Large language models (LLMs) often exhibit limited performance on domain-specific tasks due to the natural disproportionate representation of specialized information in their training data and the static nature of these datasets. Knowledge scarcity and temporal lag create knowledge gaps for domain applications. While post-training on domain datasets can embed knowledge into models, existing approaches have some limitations. Continual Pre-Training (CPT) treats all tokens in domain documents with equal importance, failing to prioritize critical knowledge points, while supervised fine-tuning (SFT) with question-answer pairs struggles to develop the coherent knowledge structures necessary for complex reasoning tasks. To address these challenges, we propose Reinforcement Learning from Augmented Generation (RLAG). Our approach iteratively cycles between sampling generations and optimizing the model through calculated rewards, effectively embedding critical and contextually coherent domain knowledge. We select generated outputs with the highest log probabilities as the sampling result, then compute three tailored reward metrics to guide the optimization process. To comprehensively evaluate domain expertise, we assess answer accuracy and the rationality of explanations generated for correctly answered questions. Experimental results across medical, legal, astronomy, and current events datasets demonstrate that our proposed method significantly outperforms baseline approaches. Our code and data are open sourced at https://github.com/ChaojunNie/RLAG.

1 Introduction

Large language models (LLMs) have demonstrated exceptional capabilities in capturing and storing factual knowledge across diverse disciplines, attributed to their comprehensive training corpora (Roberts et al., 2020; Cohen et al., 2023; Hu et al., 2023; Wang et al., 2024). However, foundation models trained on broad datasets inherently underrepresent specialized domains relative to their significance in specific applications, creating knowledge gaps in downstream applications. Due to the static nature of training data and the difficulty of accounting for all potential downstream applications during development, LLMs often struggle to answer highly specialized questions (Bang et al., 2023; Ji et al., 2023; Zhang et al., 2023a).

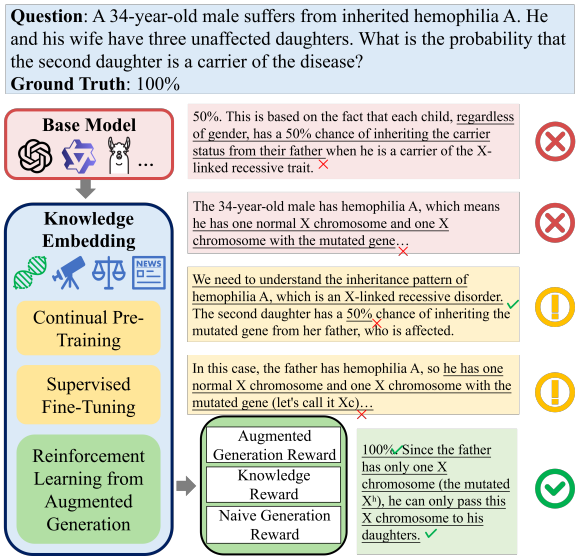


Figure 1: Illustrative example. Base models often struggle with certain task due to limited knowledge. While embedding knowledge into model helps, previous methods may still lead to errors. Our proposed Reinforcement Learning from Augmented Generation (RLAG) incorporates three rewards to optimize models iteratively, improving answer accuracy and explanation rationality.

In-context learning (ICL) enhances performance on downstream tasks by providing models with exemplars during inference, enabling adaptation without parameter updates (Wang et al., 2023a; Li et al., 2023; Highmore, 2024). Retrieval Augmented Generation (RAG) augments model outputs by integrating relevant information from external knowledge

*Corresponding author

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 19401–19418
November 4-9, 2025

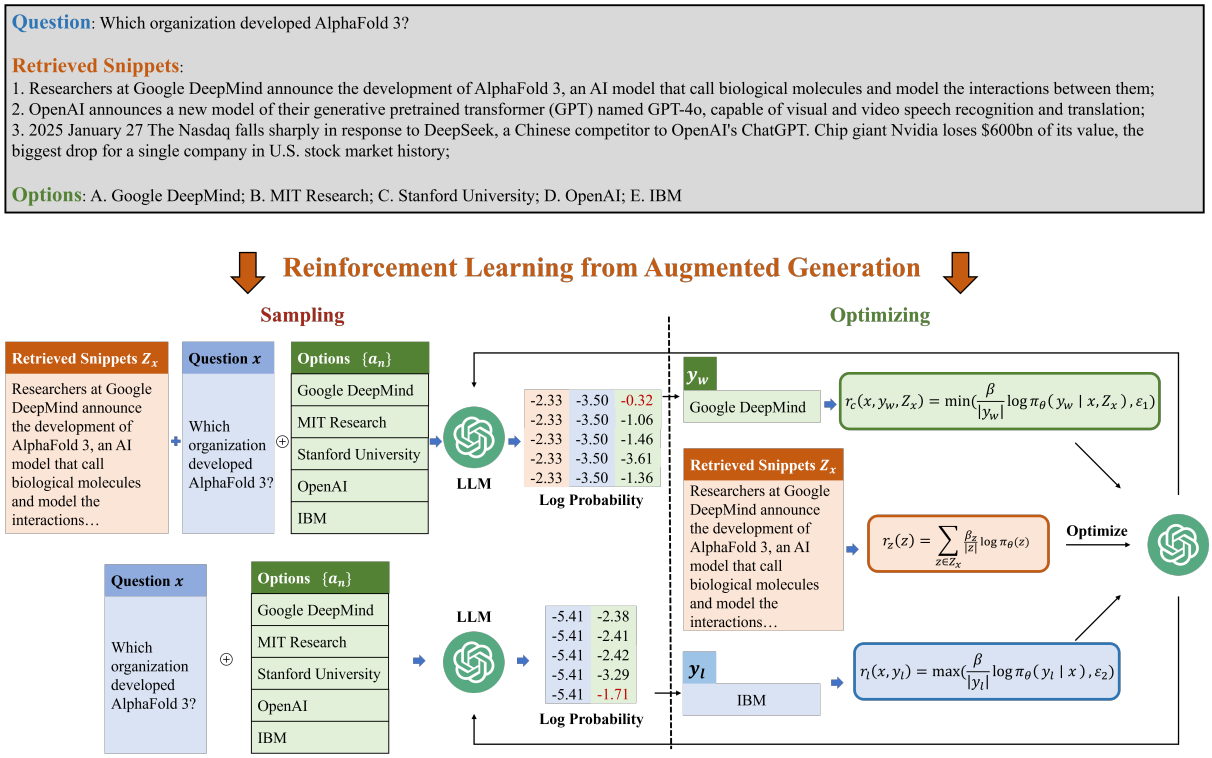


Figure 2: Overview of proposed method: Reinforcement Learning from Augmented Generation (RLAG). Augmented generation y_w (with retrieved snippets Z_x) and naive generation y_l (without retrieved snippets) are sampled using Eq 6. The model is then optimized to increase augmented generation reward r_c and knowledge reward r_z while reducing naive generation reward r_l. This process iterates using the updated model for subsequent samples.

bases, improving factual accuracy and reasoning capabilities (Guu et al., 2020; Lewis et al., 2020; Jiang et al., 2023). Since both ICL and RAG enhance performance through external information at inference time, neither permanently improves the model's intrinsic capabilities for downstream tasks.

This study focuses on ***embedding knowledge*** into model weights. Training on downstream datasets embeds domain-specific knowledge directly into model parameters, enabling autonomous reasoning without external support (Gururangan et al., 2020; Ke et al., 2023; Song et al., 2025).

While Continual Pre-Training (CPT) (Ke et al., 2023) processes entire domain corpora, its effectiveness is limited by the uniform importance assigned to tokens during training (Liu et al., 2024; Zhang et al., 2024). Supervised fine-tuning (SFT) (Wei et al., 2021) effectively embeds key information through targeted training; however, models trained exclusively on labeled knowledge pairs often exhibit reduced performance on complex reasoning tasks.

Inspired by reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022; Rafailov et al., 2023), we introduce **R**einforcement **L**earning from **A**ugmented **G**eneration (**RLAG**). In our scenario, generation augmented with relevant literature is preferred over unaugmented generation when addressing downstream questions. The core principle involves *optimizing the model to generate preferred generations independently while continuously improving these generations through iterative refinement*. Notably, our objective extends beyond enabling models to merely reproduce literature-augmented answers (achievable through SFT); we aim for models to thoroughly assimilate knowledge contained within domain literature, thereby maintaining robust knowledge capabilities throughout conversations as shown in Figure 1.

As illustrated in Figure 2, RLAG comprises two principal components: sampling and optimizing. During sampling, we employ a broadcasting operation to concatenate each option with the question, generating two responses differentiated by the presence or absence of retrieved snippets as a prefix. We compute log probabilities for each component through the model's output logits and select the maximum from the option-specific segment as prediction. The optimization phase leverages three predefined reward functions calculated from the

sampling results and retrieved snippets to update the model. In the next iteration, we use the updated model for sampling and optimization.

To further isolate LLMs' abilities to learn new knowledge, we built a dataset covering events post-model training cutoff. Current events dataset is sourced from Wikipedia (Wikipedia contributors, 2025).

We conduct experiments across biomedicine, law, science, and current events. Our experimental results show that the proposed RLAG significantly outperforms prior methods. E.g., in the terms of log-likelihood accuracy, RLAG surpasses prior methods by 14.03% on average on current events dataset.

2 Preliminaries

In a training iteration, we define output distribution $\pi_{\theta_0}(\cdot \mid x_i, Z_{x_i})$ as the preferred distribution from the LLM (parameters θ_0) conditioned on question x_i and relevant literature Z_{x_i}, and $\pi_{\theta_0}(\cdot \mid x_i)$ as the naive distribution conditioned only on question x_i.

For a model with parameters θ_1, given problem x_i without relevant literature, its output distribution can match the preferred distribution:

$$\pi_{\theta_1}(\cdot \mid x_i) \approx \pi_{\theta_0}(\cdot \mid x_i, Z_{x_i}) \succ \pi_{\theta_0}(\cdot \mid x_i) \quad (1)$$

The model with parameters θ_1 has internalized downstream knowledge, demonstrating better proficiency than θ_0 (Song et al., 2025). When given another downstream problem x_j, the distribution $\pi_{\theta_1}(\cdot \mid x_j, Z_{x_j})$ typically outperforms $\pi_{\theta_0}(\cdot \mid x_j, Z_{x_j})$ (Ovadia et al., 2023):

$$\pi_{\theta_1}(\cdot \mid x_j, Z_{x_j}) \succ \pi_{\theta_0}(\cdot \mid x_j, Z_{x_j}) \quad (2)$$

Our iterative training objective is to optimize parameters from θ_0 toward θ_1.

3 Methodology

3.1 Sampling

During each sampling, the naive generation y_l is sampled from model by concatenating the question x with each option as input, while the augmented generation y_w is sampled from the model with retrieved snippets Z_x, question x, and each option combined as input, as illustrated in the sampling section of Figure 2.

For $z' \notin Z_x$, the probability $\pi_\theta(y_w \mid x, z') \approx 0$. Thus, $\pi_\theta(y_w \mid x)$ can be approximated as:

$$\begin{aligned} \pi_\theta(y_w \mid x) &= \sum_z \pi_\theta(z \mid x)\pi_\theta(y_w \mid x, z) \\ &\approx \sum_{z \in Z_x} \pi_\theta(z \mid x)\pi_\theta(y_w \mid x, z) \end{aligned} \quad (3)$$

This decomposition shows that improving $\pi_\theta(y_w \mid x)$ requires increasing either $\pi_\theta(z \mid x)$ or $\pi_\theta(y_w \mid x, z)$. Since $\pi_\theta(y_w \mid x, z)$ is already high and further optimization risks overfitting, we focus on enhancing the posterior probability $\pi_\theta(z \mid x)$.

Directly optimizing $\pi_\theta(z \mid x)$ is computationally challenging. Instead, we enhance the prior probability $\pi_\theta(z)$ to improve $\pi_\theta(z \mid x)$. The relationship between these probabilities is captured by the partial derivative:

$$\frac{\partial \pi_\theta(z \mid x)}{\partial \pi_\theta(z)} = \frac{\pi_\theta(x \mid z) \sum_{z' \notin Z_x} \pi_\theta(x \mid z')\pi_\theta(z')}{\pi_\theta(x)^2} \quad (4)$$

See Appendix C.1 for a complete derivation. Since $\pi_\theta(x) > 0$ and $\pi_\theta(x \mid z) > 0$ for $z \in Z_x$ (as z represents one of the top-k retrieved documents), and making the reasonable and accessible assumption that a sufficiently large document corpus contains at least one relevant snippets $z' \notin Z_x$ with $\pi_\theta(x \mid z') > 0$, we can conclude that the derivative is positive:

$$\frac{\partial \pi_\theta(z \mid x)}{\partial \pi_\theta(z)} > 0 \quad (5)$$

This demonstrates that increasing the prior $\pi_\theta(z)$ effectively enhances the posterior $\pi_\theta(z \mid x)$.

To eliminate prompt template bias, we concatenate each question x with its corresponding options a_n^l and input them into the model, then calculate log probabilities only for the option segment. The prediction is defined as $\mathcal{P}_\theta(x) = c_n$, where:

$$c_n = \arg\max_l \{\mathcal{P}_\theta(x||a_n^1), \cdots, \mathcal{P}_\theta(x||a_n^L)\} \quad (6)$$

and $\mathcal{P}_\theta(x||a_n^l) = \log \pi_\theta(x||a_n^l)$.

3.2 Reward

To approximate the target described in Section 2, we define two reward functions, r_w and r_l, which guide the model optimizing. r_w is designed to embed knowledge into model weights and is expressed

as:

$$
\begin{aligned}
&r_w(x, y_w, Z_x) \\
&= \sum_{z \in Z_x} \frac{\beta_z}{|z|} \log \pi_\theta(z) + \frac{\beta}{|y_w|} \log \pi_\theta(y_w \mid x, Z_x) \\
&= r_z(Z_x) + r_c(x, y_w, Z_x)
\end{aligned} \tag{7}
$$

where Z_x denotes retrieved snippets relevant to question x, and y_w represents the augmented generation. Parameter β_z controls the weight of the knowledge reward r_z and β adjusts the augmented generation reward r_c. Length normalization prevent the model from favoring excessively long outputs. The naive generation reward r_l is defined for naive generation y_l generated without Z_x:

$$
r_l(x, y_l) = \frac{\beta}{|y_l|} \log \pi_\theta(y_l \mid x) \tag{8}
$$

3.3 Reinforcement Learning from Augmented Generation

We employ a Bradley-Terry (Bradley and Terry, 1952) model with target reward margin γ (Meng et al., 2025). The preference probability is defined as:

$$
P(y_w \succ y_l \mid x) = \sigma(r_w - r_l - \gamma) \tag{9}
$$

where σ denotes the sigmoid function.

Sampling-driven β adaption. Similar to RLHF, when sampling yields identical outputs ($y_w = y_l$), the generation signal becomes invalid, prompting us to set $\beta = 0$ to disable generation rewards while retaining the knowledge reward controlled by β_z. When $y_w \neq y_l$, optimization proceeds with all three rewards activated to optimize model. Full configurations appear in Appendix A.2.

Clipping strategy. To mitigate overfitting, we introduce a clipping strategy. Probabilities $\pi_\theta(y_w \mid x, Z_x)$ exceeding a threshold ϵ_1 and $\pi_\theta(y_l \mid x)$ falling below a threshold ϵ_2 are clipped. Substituting r_w(Eq 7),r_l(Eq 8) into Eq 9. The resulting RLAG loss function is:

$$
\begin{aligned}
&\mathcal{L}_{\text{RLAG}} = \\
&- \mathbb{E}_{(x, y_w, y_l, Z_x) \sim \mathcal{D}} \Bigg[\log \sigma \Bigg(\sum_{z \in Z_x} \frac{\beta_z}{|z|} \log \pi_\theta(z) \\
&+ \min \left(\frac{\beta}{|y_w|} \log \pi_\theta(y_w \mid x, Z_x), \epsilon_1 \right) \\
&- \max \left(\frac{\beta}{|y_l|} \log \pi_\theta(y_l \mid x), \epsilon_2 \right) - \gamma \Bigg) \Bigg].
\end{aligned} \tag{10}
$$

where ϵ_1 and ϵ_2 are adjustable hyperparameters. The complete derivation appears in the Appendix C.2. Specifically, ϵ_1 caps the maximum probability for augmented generation to avoid overfitting to specific knowledge contexts, while ϵ_2 sets a minimum probability for naive generation to ensure the model does not overly suppress naive generation in the absence of knowledge documents.

Role of reward components. The knowledge reward r_z facilitates the embedding of downstream knowledge into the model by increasing the prior probability of relevant knowledge documents. The augmented generation reward r_w ensures that knowledge embedding aligns with the target parameters, guiding the model toward preferred model. Meanwhile, naive generation reward r_l reduces the likelihood of y_l, further reinforcing knowledge integration. Notably, while the generation rewards themselves don't directly embed knowledge into the model, they serve as guides in this optimizing process—architects of direction rather than builders of content.

4 Knowledge Base Creation

4.1 Task Selection and Statistics of Data

Experiments were conducted across four distinct downstream tasks.

Biomedicine: The USMLE task from MedQA (Jin et al., 2021), drawn from U.S. National Medical Licensing Examinations, represents a high-difficulty challenge in medical reasoning. USMLE comprises $10,178$ training instances, $1,272$ validation instances, $1,273$ testing instances, and 18 biomedicine books.

Law: The BarExamQA (Zheng et al., 2025) task comprises legal questions from practical bar exams. BarExamQA incorporates 954 training instances, 124 validation instances, 117 testing instances, and legal documents.

Astronomy: Astronomy task from the MMLU (Hendrycks et al., 2020) benchmark, with training data generated using GPT-4 Turbo (Hurst et al., 2024) and DeepSeek-R1 (Guo et al., 2025). This tested the model's scientific knowledge. The astronomy task contains $2,000$ training instances, 134 validation instances, and 152 testing instances.

Current Events: We developed a dataset encompassing post-training temporal phenomena, consisting of $1,300$ training instances, 169 validation instances, and 162 testing instances.

Table 1: Results for USMLE (Jin et al., 2021), BarExamQA (Zheng et al., 2025), and Astronomy (Hendrycks et al., 2020). Accuracy quantified by Eq 6; explanation win rates at temperature 0.3 assessed by GPT-4 Turbo and Grok-3.

Method	Llama-3.1-8B-Instruct								
	USMLE			BarExamQA			Astronomy		
	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)
Base	27.6	26.2	27.2	39.3	**38.5**	**39.3**	46.7	**45.4**	46.7
SFT	32.2	26.9	29.9	37.6	26.5	29.1	49.3	37.5	42.8
CPT	29.2	25.3	28.5	35.0	27.4	32.5	48.7	46.1	47.4
CPT+SFT	33.3	25.0	30.7	36.8	26.5	23.9	48.0	38.2	42.1
RLAG	**34.8**	**32.4**	**33.9**	**41.9**	35.9	38.5	**51.3**	**45.4**	**50.0**
Method	**Qwen2-7B-Instruct**								
	USMLE			BarExamQA			Astronomy		
	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)
Base	25.8	21.4	24.9	34.2	32.5	32.5	50.7	50.0	50.7
SFT	27.7	15.0	23.2	31.6	19.7	30.8	**53.9**	50.7	50.7
CPT	26.4	21.3	25.5	35.0	32.5	35.0	48.7	46.7	46.7
CPT+SFT	27.0	15.9	23.3	34.2	21.4	17.9	52.0	47.4	49.3
RLAG	**29.4**	**23.6**	**27.8**	**40.2**	**35.0**	**38.5**	53.3	**52.0**	**52.0**
Method	**Llama-3.2-3B-Instruct**								
	USMLE			BarExamQA			Astronomy		
	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)
Base	26.2	22.8	24.8	34.2	**25.6**	29.1	49.3	44.1	47.4
SFT	**30.2**	25.6	27.9	33.3	21.4	24.8	50.0	46.1	48.0
CPT	27.4	22.8	25.5	28.2	16.2	22.2	47.4	40.1	44.7
CPT+SFT	29.3	22.3	27.2	29.9	19.7	17.1	46.7	40.8	42.1
RLAG	29.7	**25.9**	**28.1**	**36.8**	**25.6**	**33.3**	**52.0**	**46.7**	**51.3**

Table 2: Results for the Current Events in terms of log-likelihood accuracy (Eq 6)

Task	Model	Base	SFT	CPT	CPT+SFT	RLAG
Current Events	Qwen2-7B-Instruct	25.3	32.1	27.2	34.6	**48.8**
	Llama-3.1-8B-Instruct	30.2	34.0	29.6	35.8	**54.9**
	Llama-3.2-3B-Instruct	23.5	25.9	22.8	27.2	**37.0**

The original developers released these research-focused datasets, which have been extensively cited in academic literature. We strictly comply with each dataset's usage terms, ensuring their application remains limited to scholarly research.

4.2 Knowledge Base Creation

USMLE: For the USMLE task, we curated a knowledge base from 18 biomedical textbooks provided by the MedQA (Jin et al., 2021) through systematic text cleaning and structural normalization. The USMLE knowledge base (KB) has 17.3M tokens. All token counts use LlamaTokenizer.

BarExamQA: For the BarExamQA (Zheng et al., 2025) task, we utilized gold passages provided with each sample as reference documents. The BarExamQA KB has 93.1M tokens.

Astronomy: For the MMLU astronomy task (Hendrycks et al., 2020), we followed a structured process: DeepSeek-R1 (Guo et al., 2025) extracted keywords from astronomy questions. Then we collected text by searching keywords with the Wikipedia API [1] and generated samples using Deepseek-R1. The Astronomy KB has 3.1M tokens. Following previous work (Guo et al., 2024), we removed questions from the training set that had 3-gram overlaps with the test set to prevent test set contamination. Curation was performed via the Claude-3.7-Sonnet API[2] and manual review eliminated ambiguous/incorrect questions.

Current events: For the current events task, we collected events after the model training data cutoff date from Wikipedia (Wikipedia contributors, 2025), including: 2024-2025 U.S. events, 2025 German federal election, and 2024 Summer Olympics. The text was segmented and cleaned using spaCy (Honnibal et al., 2020). The Current events KB has 51.5K tokens. GPT-4 Turbo (Hurst et al., 2024) generated questions for each five-line segment. Equal samples were generated per topic, with test sets uniformly sampled across all events. Recognizing RLAG's potential privacy risks from personal information in training data, we manually screened the dataset to eliminate ethical concerns. This dataset is for academic research use only.

[1] https://www.mediawiki.org/wiki/API:Main_page

[2] https://www.anthropic.com/claude/sonnet

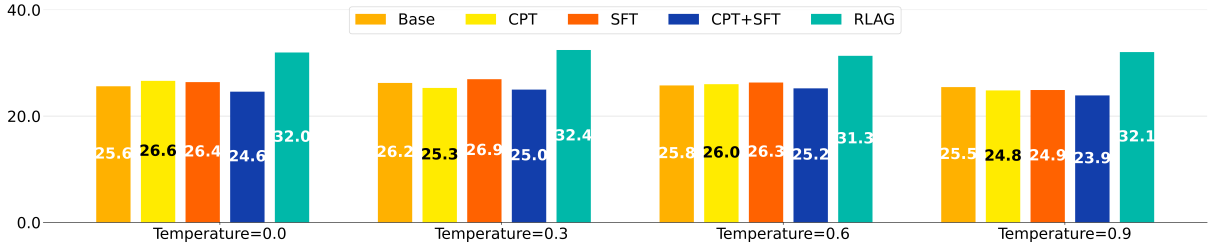


Figure 3: Evaluate explanation quality of questions correctly answered by Llama-3.1-8B-Instruct across temperatures on USMLE dataset, which is conducted by GPT-4 Turbo.

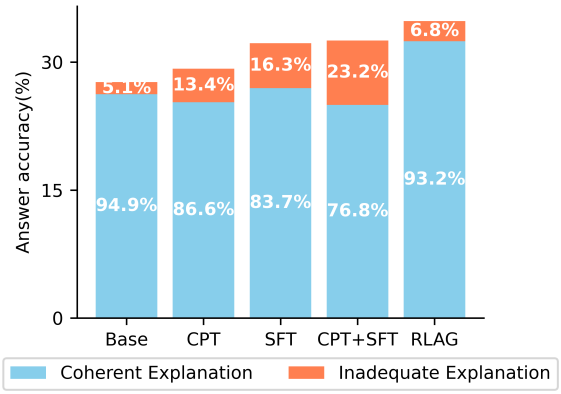


Figure 4: Performance comparison between RLAG and baseline approaches on the USMLE dataset with temperature set to 0.3. Results show answer accuracy and explanation rationality, with explanations evaluated by GPT-4 Turbo.

5 Experiments and Results

5.1 Experimental Setup

Models and training settings. Knowledge embedding experiments used two model families: Qwen2 (Yang et al., 2024) and Llama3 (Grattafiori et al., 2024). We selected both large and small variants: Qwen2-7B-Instruct, Llama-3.1-8B-Instruct, and Llama-3.2-3B-Instruct to analyze knowledge embedding effects across different parameter scales. We used instruction-tuned models off-the-shelf, as these are commonly deployed in practice, making the embedding of downstream knowledge into these models practically significant. *NV-Embed-v2* (Lee et al., 2024) was selected as the embedding model, and used FAISS (Johnson et al., 2019) as its vector-store. We report the best performance obtained via a grid search of hyperparameters, while ablation studies and evaluation of explanations were conducted with single experimental runs.

Tokenizers configured with padding token to the end-of-sequence token and assigned Qwen2Tokenizer's beginning-of-sequence token to $<|\text{im\_start}|>$[3]. Details of training configurations and retrieval methods are provided in Appendix A.1 and Appendix A.8, respectively. Retrieval ablation experiments in Appendix A.9 show RLAG's robust performance across different retrieval methods.

Baselines. The SFT loss function is defined as:

$$\mathcal{L}_{\text{SFT}} = -\sum_{i=1}^{B} \frac{1}{|y_i|} \sum_{j=1}^{|y_i|} \pi_\theta(y_{i,j} \mid x, y_{i,<j}) \quad (11)$$

where B is the batch size, y_i is the answer sequence, and $y_{i,j}$ is its j-th token. We apply length normalization to prevent bias toward longer outputs.

The CPT loss function is:

$$\mathcal{L}_{\text{CPT}} = -\sum_{i=1}^{B} \frac{1}{|z_i|} \sum_{j=1}^{|z_i|} \pi_\theta(z_{i,j} \mid z_{i,<j}) \quad (12)$$

where z_i represents a knowledge document chunk, and $z_{i,j}$ is its j-th token.

To enhance knowledge embedding effectiveness, we also explored a pipeline combining CPT on knowledge documents followed by SFT.

5.2 Evaluation Method

We employed a two-stage sequential evaluation: answer accuracy followed by explanation assessment for correctly answered questions.

Log-likelihood accuracy. We ensured prompt-independent results by connecting each option to the question, calculating generation probabilities, and selecting the highest-probability option as the prediction (Eq 6).

[3] https://huggingface.co/docs/transformers/main/en/chat_templating

Explanation win rates. For correctly answered questions, we evaluated knowledge embedding by prompting models to explain their answers. Explanations were assessed for logical clarity and factual accuracy using GPT-4 Turbo (Hurst et al., 2024) and Grok-3 (xAI, 2024), with win rates calculated as percentages. Complete evaluation templates appear in Appendix A.3.

5.3 Main Results

Downstream tasks results. Table 1 demonstrates RLAG's superior performance across tasks. On USMLE (Jin et al., 2021), RLAG achieves the highest overall answer accuracy and surpasses all baselines in explanation win rate by $2.2-5.5$ points. For BarExamQA (Zheng et al., 2025), RLAG outperforms the best baseline by $3.5-5.2$ points in accuracy while maintaining superior explanation rationality. This legal reasoning task reveals the limitations of baseline methods: SFT merely learns question-answer mappings without robust reasoning, while CPT suffers from catastrophic forgetting as vast legal documents. Even on Astronomy (Hendrycks et al., 2020), where injected knowledge is primarily factual and benefits SFT, RLAG still outperforms all baselines, whereas CPT on the Astronomy knowledge base degrades model performance.

Explanation win rates across temperature. As shown in Figure 3, RLAG outperforms all baselines by $5.0-7.2$ points in explanation win rate across temperatures. While baseline training improves answer accuracy, it compromises explanation rationality, with unexplained portions rising from 5.1% to $13.4-23.2\%$ ($>100\%$ relative increase). RLAG enhances accuracy while preserving explanation quality, with unexplained portions increasing marginally from 5.1% to 6.8% (Figure 4). This demonstrates that RLAG embeds domain knowledge comprehensively into the model, ensuring logical coherence without requiring manual annotation. We conducted additional small-scale human evaluation to further substantiate RLAG's effectiveness. The experimental protocol and results are provided in Appendix B.

Current events results. Table 2 presents results on current events. Although CPT+SFT pipeline can effectively improve the performance of the model, RLAG demonstrates significant gains of $9.8-19.1$ points over optimal baselines. Larger 7B-8B models show more substantial improvements (14.2 and 19.1 points respectively), while the 3B model improves by 9.8 points. As this task focuses on factual questions, explanation rationality was not evaluated.

5.4 Ablation Studies

Four components were evaluated in RLAG via ablation studies with Llama-3.1-8B-Instruct: (1) Reward Clipping (w/o Clip), (2) Dynamic β, β_z (Fixed β, β_z), (3) reward margin γ (w/o γ), and (4) directly using the standard answer as the augmented generation in Eq 10 (Std. Ans. as y_w).

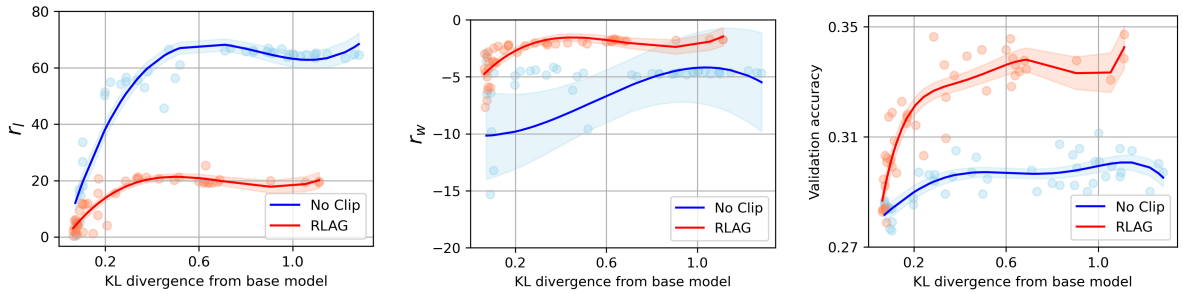
Table 3 shows all components are critical, with reward clipping having the strongest impact. Removing reward clipping significantly affects reasoning tasks, reducing answer accuracy by 4.3% on USMLE and 7.9% on BarExamQA, with explanation rationality decreasing by 9% for both tasks. However, it minimally impacts factual knowledge tasks like Astronomy. Fixed β, β_z and removing reward margin γ also decrease performance. Using standard answers as augmented generation (Eq 10) dramatically reduces performance, causing serious hallucinations—USMLE explanation rationality drops by over 28 points. This indicates models may learn correct answers but fail to develop robust reasoning when answers are directly provided rather than autonomously generated.

The key role of reward clipping. Reward clipping is essential in our method. Figure 5a shows unconstrained naive generation reward r_l rapidly increases as model divergence occurs, while Figure 5b indicates minimal growth in r_w, yielding negligible validation accuracy improvements (Figure 5c). Conversely, RLAG with reward clipping effectively constrains r_l while maintaining superior r_w compared to the w/o clipping. This results in consistently higher validation accuracy, highlighting reward clipping's critical contribution to model performance.

Using standard answer weakens RLAG. Direct substitution of standard answers for augmented generation significantly degrades model performance and induces hallucinations (Table 3), particularly in reasoning-intensive domains like USMLE. Our case study (Appendix D) demonstrates that this approach causes the model to contradict previously answered questions and question the validity of given options. The effectiveness of knowledge embedding strategies ultimately depends on task complexity and reasoning requirements.

Table 3: Ablation study on Llama-3.1-8B-Instruct. We ablate four keys of RLAG:(1) No Clipping in Eq 10 ($i.e.$, w/o Clip), (2) Fix β, β_z in Eq 10 ($i.e.$, Fixed β, β_z), (3) Set $\gamma = 0$ in Eq 10 ($i.e.$, w/o γ), (4) Replace sample y_w with standard answer in Eq 10 ($i.e.$ Std. Ans. as y_w)

Method	Llama-3.1-8B-Instruct								
	USMLE			BarExamQA			Astronomy		
	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)	ACC(%)	GPT-4 Turbo WR(%)	Grok-3 WR(%)
CPT+SFT	33.3	25.0	30.7	36.8	26.5	23.9	48.0	38.2	42.1
RLAG	34.8	32.4	33.9	41.9	35.9	38.5	51.3	45.4	50.0
w/o Clip	30.5	24.1	24.6	35.0	27.4	29.1	52.0	48.0	48.7
Fixed β, β_z	32.9	29.5	30.5	32.5	22.2	29.1	48.7	46.1	46.7
w/o γ	32.1	29.1	29.6	36.8	28.2	29.1	48.7	46.1	48.0
Std. Ans. as y_w	31.0	4.24	5.34	35.0	29.1	31.6	49.3	46.7	49.3



(a) Effect of clipping on r_l (b) Effect of clipping on r_w (c) Effect of clipping on validation set

Figure 5: Ablation study on reward clipping effects: constraining naive reward r_l inflation (a) while steadily increasing r_w (b) and preserving accuracy (c), demonstrating effective reward control.

Table 4: Computational Budget in terms of GPU hours

Task	Model	SFT	CPT	CPT+SFT	RLAG
USMLE	Qwen2-7B-Instruct	4	4	8	32
	Llama-3.1-8B-Instruct	6	4	10	34
	Llama-3.2-3B-Instruct	3	2	5	18
BarExamQA	Qwen2-7B-Instruct	1	22	23	8
	Llama-3.1-8B-Instruct	1	27	28	9
	Llama-3.2-3B-Instruct	1	12	13	6
Astronomy	Qwen2-7B-Instruct	1	1	2	12
	Llama-3.1-8B-Instruct	1	1	2	10
	Llama-3.2-3B-Instruct	0.3	0.3	0.6	8
Current Events	Qwen2-7B-Instruct	1	0.3	1.3	8
	Llama-3.1-8B-Instruct	0.8	0.3	1	8
	Llama-3.2-3B-Instruct	0.3	0.3	0.6	4

5.5 Computational Budget

All experiments ran on a server with four NVIDIA A800 GPUs (80GB each). As shown in Table 4, RLAG training requires approximately one order of magnitude more GPU hours than the baseline due to online sampling and optimization processes. Future implementation updates could focus on incorporating efficient sampling frameworks such as vLLM (Kwon et al., 2023), which would reduce sampling time and thus decrease the overall training time. Despite the increased computational cost, the significant performance improvements justify this additional investment.

6 Related Work

Knowledge injection. In order to enhance LLMs' capabilities in downstream tasks, knowledge injection is considered a promising research direction (Chen et al., 2022; Ye et al., 2023). Knowledge injection for LLMs can occur during pre-training, fine-tuning, or inference stages. Methods include: (1) RAG, which retrieves text (Guu et al., 2020; Lewis et al., 2020) or graph-structured (Wang et al., 2023b; Zhang et al., 2023b; Li et al., 2024) information during reasoning; (2) Modular adapters, which incorporate domain knowledge through lightweight additional parameters (Zhang et al., 2023c; Lo et al., 2024); (3) Prompt optimization techniques that leverage internal knowledge (Wei et al., 2022); and (4) Direct weight embedding through CPT (Ke et al., 2023) or SFT, which enhances domain expertise and stability (Gururangan et al., 2020; Song et al., 2025). Recent advances focus on optimizing knowledge structures (Zhang et al., 2024), implementing gating mechanisms (Peinelt et al., 2021), and developing structure-aware training strategies (Liu et al., 2024) .

Reinforcement learning from human feedback (RLHF). RLHF technology enhances LLMs' performance using reinforcement learning with pref-

erence data (Ouyang et al., 2022; Stiennon et al., 2020). The approach trains a reward model on preference data, then uses PPO to optimize the policy model, significantly improving generation quality (Shao et al., 2024; Guo et al., 2025). DPO (Radford et al., 2019) reparameterizes reward model and directly using preference data to optimize the policy model. RLHF does not focus on embedding knowledge into the model, but improves the output by aligning with humans.

7 Conclusion and Future Work

In this work, we propose RLAG for knowledge embedding. Compared with traditional knowledge embedding methods, RLAG can solve knowledge-intensive tasks that require reasoning. The core idea of RLAG is to enable the model to independently generate augmented generation and optimize these generation through a reward-based approach. The training process is implemented iteratively by sampling and optimization. Experiments show that RLAG outperforms baseline methods. In future work, we aim to dynamically embed knowledge into LLMs, rather than performing offline training.

Limitations

RLAG, while showing promising results in embedding knowledge into LLMs, has several limitations. (i)Although RLAG eliminates the need for manual annotation during training, it requires knowledge documents relevant to each question. These document fragments can be collected by searching knowledge bases through retrieval systems. However, the quality of these retrieved fragments depends on retriever performance and knowledge base structure, potentially affecting overall system effectiveness. (ii)The training process of RLAG encompasses two phases: sampling and optimization. While we have demonstrated the sampling process to be effective, it may require more computational time than training directly on existing datasets. (iii)Both sampling and training processes within RLAG require access to token probabilities, making our approach unsuitable for closed-source models that do not provide such access. (iv)Due to hardware constraints, our research primarily focuses on language models with 3B, 7B, and 8B parameters and does not extend to larger-scale models that might yield different performance characteristics. (v)The datasets for law, astronomy, and current events datasets are relatively smaller compared to the medical dataset, which may affect the generalizability of individual dataset results. However, reporting results across multiple datasets collectively enhances the overall credibility of our findings. (vi)This study employs two powerful commercial large language models—GPT-4 Turbo and Grok-3—to evaluate explanation win rates. Although the results demonstrate reasonable reproducibility, the closed-source nature of these models may introduce variability in evaluation outcomes.

Ethical Considerations

Data Collection and Privacy Our training datasets pose minimal privacy risks: current events data was exclusively sourced from public Wikipedia with manual screening to exclude personal information, while other datasets are established benchmarks from prior research. RLAG's knowledge embedding capabilities could theoretically raise privacy concerns, but our data sources and screening protocols mitigate these risks.

Human Evaluation Protocol We conducted small-scale human evaluation with anonymous participants recruited transparently via social media. Participants were volunteers residing in Beijing and Shenzhen, China. All participants provided informed consent after being fully briefed on task requirements, the approximate 3-minute time commitment per annotation, and how their anonymized responses would be used for research evaluation purposes. Compensation was set at 5 CNY per annotation, ensuring fair payment above minimum wage standards. To protect participant privacy, we implemented strict anonymization protocols using random IDs and secure data storage with restricted access. Participation remained entirely voluntary throughout the study, with participants explicitly informed of their right to withdraw at any time without penalty. The evaluation tasks focused exclusively on benign text quality assessment, ensuring no exposure to harmful or distressing content. Detailed recruitment and evaluation procedures are documented in Appendix B.

Acknowledgements

This work was funded by the Xinjiang Uygur Autonomous Region Key Research and Development Project (2023B03024). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, and 1 others. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM Web conference 2022*, pages 2778–2788.

Roi Cohen, Mor Geva, Jonathan Berant, and Amir Globerson. 2023. Crawling the internal knowledge-base of language models. *arXiv preprint arXiv:2301.12810*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yu Wu, YK Li, and 1 others. 2024. Deepseek-coder: When the large language model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*.

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

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Clyde Highmore. 2024. In-context learning in large language models: A comprehensive survey. *Preprints*, 202407:v1.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python. Zenodo. Software library, version 2.3.0.

Linmei Hu, Zeyi Liu, Ziwang Zhao, Lei Hou, Liqiang Nie, and Juanzi Li. 2023. A survey of knowledge enhanced pre-trained language models. *IEEE Transactions on Knowledge and Data Engineering*, 36(4):1413–1430.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. 2023. Continual pre-training of language models. *arXiv preprint arXiv:2302.03241*.

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

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Jia Li, Chongyang Tao, Jia Li, Ge Li, Zhi Jin, Huangzhao Zhang, Zheng Fang, and Fang Liu. 2023. Large language model-aware in-context learning for code generation. *ACM Transactions on Software Engineering and Methodology*.

Yading Li, Dandan Song, Changzhi Zhou, Yuhang Tian, Hao Wang, Ziyi Yang, and Shuhao Zhang. 2024. A framework of knowledge graph-enhanced large language model based on question decomposition and atomic retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11472–11485.

Kai Liu, Ze Chen, Zhihang Fu, Rongxin Jiang, Fan Zhou, Yaowu Chen, Yue Wu, and Jieping Ye. 2024. Educating llms like human students: Structure-aware injection of domain knowledge. *arXiv e-prints*, pages arXiv–2407.

Ka Man Lo, Yiming Liang, Wenyu Du, Yuantao Fan, Zili Wang, Wenhao Huang, Lei Ma, and Jie Fu. 2024. m2mkd: Module-to-module knowledge distillation for modular transformers. *arXiv preprint arXiv:2402.16918*.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2025. Simpo: Simple preference optimization with a reference-free reward. *Advances in Neural Information Processing Systems*, 37:124198–124235.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

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

Nicole Peinelt, Marek Rei, and Maria Liakata. 2021. Gibert: Enhancing bert with linguistic information using a lightweight gated injection method. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2322–2336.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Zirui Song, Bin Yan, Yuhan Liu, Miao Fang, Mingzhe Li, Rui Yan, and Xiuying Chen. 2025. Injecting domain-specific knowledge into large language models: A comprehensive survey. *arXiv preprint arXiv:2502.10708*.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021.

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

Mengru Wang, Yunzhi Yao, Ziwen Xu, Shuofei Qiao, Shumin Deng, Peng Wang, Xiang Chen, Jia-Chen Gu, Yong Jiang, Pengjun Xie, and 1 others. 2024. Knowledge mechanisms in large language models: A survey and perspective. *arXiv preprint arXiv:2407.15017*.

Yujie Wang, Hu Zhang, Jiye Liang, and Ru Li. 2023b. Dynamic heterogeneous-graph reasoning with language models and knowledge representation learning for commonsense question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14048–14063.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Wikipedia contributors. 2025. Wikipedia. https://www.wikipedia.org/.

xAI. 2024. Grok-3. https://x.ai/grok-3.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Qichen Ye, Junling Liu, Dading Chong, Peilin Zhou, Yining Hua, Fenglin Liu, Meng Cao, Ziming Wang, Xuxin Cheng, Zhu Lei, and 1 others. 2023. Qilin-med: Multi-stage knowledge injection advanced medical large language model. *arXiv preprint arXiv:2310.09089*.

Jiaxin Zhang, Wendi Cui, Yiran Huang, Kamalika Das, and Sricharan Kumar. 2024. Synthetic knowledge ingestion: Towards knowledge refinement and injection for enhancing large language models. *arXiv preprint arXiv:2410.09629*.

Jiaxin Zhang, Zhuohang Li, Kamalika Das, Bradley A Malin, and Sricharan Kumar. 2023a. Sac3: Reliable hallucination detection in black-box language models via semantic-aware cross-check consistency. *arXiv preprint arXiv:2311.01740*.

Taolin Zhang, Ruyao Xu, Chengyu Wang, Zhongjie Duan, Cen Chen, Minghui Qiu, Dawei Cheng, Xiaofeng He, and Weining Qian. 2023b. Learning knowledge-enhanced contextual language representations for domain natural language understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15663–15676.

Zhengyan Zhang, Zhiyuan Zeng, Yankai Lin, Huadong Wang, Deming Ye, Chaojun Xiao, Xu Han, Zhiyuan Liu, Peng Li, Maosong Sun, and 1 others. 2023c. Plug-and-play knowledge injection for pre-trained language models. *arXiv preprint arXiv:2305.17691*.

Lucia Zheng, Neel Guha, Javokhir Arifov, Sarah Zhang, Michal Skreta, Christopher D Manning, Peter Henderson, and Daniel E Ho. 2025. A reasoning-focused legal retrieval benchmark. In *Proceedings of the Symposium on Computer Science and Law on ZZZ*, pages 169–193.

A Hyperparameters

A.1 Training Setups

All our experiments were performed on 4 A800 GPUs, using the AdamW optimizer, cosine learning rate rise, and warm up ratio of 0.1. RLAG experimental Epochs set to 5, Learning Rate set to 1.0×10^{-5}, Updates 2 per Iteration. We performed parallel experiments using three random seeds: 62512, 34, and 767. We wrapped documents with the tokenizer's beginning- and end-of-sequence tokens, segmented them into 256-token chunks, and normalized them by length for CPT.

The number of splits in our training set is equal to the number of iterations in an epoch, and we divide it according to the number of training sets.

We use FSDP[4] for training, the Qwen2 model wraps Qwen2DecoderLayer for training, and the Llama3 model wraps LlamaDecoder for training.

A.2 Dynamic β, β_z Selection

The parameters β_z and β are chosen based on the sampling results, as follows:

$$\begin{cases} \beta_z = 0.2, \beta = 0.5 & \text{if } y_w \neq y_l, \\ \beta_z = 0.5, \beta = 0.0 & \text{if } y_w = y_l. \end{cases}$$

A.3 Win Rates Template

"User question 1":{question}
"Assistant response 1":{answer}
"User question 2":Explain your answer. Why?

Table 5: Explanation Template

Enter a conversation between the user and the assistant. You need to determine whether the assistant can explain its first output answer in the second answer. If the assistant can give a correct and logical explanation in the second answer, directly output WIN, otherwise output LOSE User: {user_question_1} Assistant: {response_1} User: {user_question_2} Assistant: {response_2}

Table 6: API Evaluation Template

[4] https://pytorch.org/tutorials/intermediate/FSDP_tutorial.html

A.4 Sampling Template

If the sampling is naive generation, no relevant literature will be added.

You are an AI that answers single-choice questions by selecting one of the provided options. Given the question and options separated by semicolons (;), output only one of the exact text of the correct option. Do not include any additional text, explanations, or multiple options. <Example>: Question: What is the capital of France? Options: Berlin; Madrid; Paris; Rome Answer: Paris.</Example>Now, answer the following question: Related literature: {ctx} Question: {question} Options: {options} Answer:

Table 7: Sampling Template

A.5 RLAG Hyperparameters

Table 8: RLAG Hyperparameters on USMLE

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
γ	0.8	8	8
Iterations per Epoch	9	9	9
Batch Size	1024	1024	1024
Gradient Accumulation	256	256	256
Grad Norm	5.0	5.0	1.0

Table 9: RLAG Hyperparameters on BaeExamQA

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
γ	0.8	0.8	0.8
Iterations per Epoch	7	7	7
Batch Size	128	128	128
Gradient Accumulation	32	32	23
Grad Norm	1.0	1.0	1.0

Table 10: RLAG Hyperparameters on Astronomy

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
γ	0.8	0.8	0.8
Iterations per Epoch	8	8	8
Batch Size	256	256	256
Gradient Accumulation	64	64	64
Grad Norm	1.0	1.0	1.0

Table 11: RLAG Hyperparameters on CurrentEvents

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
γ	0.8	0.8	0.8
Iterations per Epoch	6	6	6
Batch Size	246	246	246
Gradient Accumulation	61	61	61
Grad Norm	1.0	1.0	1.0

A.6 SFT Hyperparameters

Table 12: SFT Hyperparameters on USMLE

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	1.0×10^{-5}	5.0×10^{-6}	5.0×10^{-6}
Epoch	5	5	5
Batch Size	128	128	128
Gradient Accumulation	8	8	8
Grad Norm	1.0	1.0	1.0

Table 13: SFT Hyperparameters on BarExamQA

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	5.0×10^{-6}	5.0×10^{-6}	5.0×10^{-6}
Epoch	5	5	5
Batch Size	128	128	128
Gradient Accumulation	8	8	8
Grad Norm	1.0	1.0	1.0

Table 14: SFT Hyperparameters on Astronomy

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	5.0×10^{-6}	5.0×10^{-6}	5.0×10^{-6}
Epoch	5	5	5
Batch Size	128	128	128
Gradient Accumulation	8	8	8
Grad Norm	1.0	1.0	1.0

Table 15: SFT Hyperparameters on Current Events

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	1.0×10^{-5}	1.0×10^{-5}	1.0×10^{-5}
Epoch	5	5	5
Batch Size	128	128	128
Gradient Accumulation	8	8	8
Grad Norm	1.0	1.0	1.0

A.7 CPT Hyperparameters

Table 16: CPT Hyperparameters on USMLE

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	5.0×10^{-6}	5.0×10^{-6}	5.0×10^{-6}
Epoch	2	2	2
Batch Size	1024	1024	1024
Gradient Accumulation	16	16	16
Grad Norm	1.0	1.0	1.0

Table 17: CPT Hyperparameters on BarExamQA

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	5.0×10^{-6}	5.0×10^{-6}	5.0×10^{-6}
Epoch	2	2	2
Batch Size	1024	1024	1024
Gradient Accumulation	16	16	16
Grad Norm	1.0	1.0	1.0

Table 18: CPT Hyperparameters on Astronomy

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	5.0×10^{-6}	5.0×10^{-6}	5.0×10^{-6}
Epoch	2	2	2
Batch Size	1024	1024	1024
Gradient Accumulation	16	16	16
Grad Norm	1.0	1.0	1.0

Table 19: CPT Hyperparameters on Current Events

parameter	Qwen2-7B-Instruct	Llama-3.1-8B-Instruct	Llama-3.2-3B-Instruct
Learning Rate	1.0×10^{-5}	1.0×10^{-5}	1.0×10^{-5}
Epoch	5	5	5
Batch Size	128	128	128
Gradient Accumulation	2	2	2
Grad Norm	1.0	1.0	1.0

A.8 Retrieval Method

We tailored retrieval strategies to each task's specific characteristics:

USMLE retrieval. We merged keyword (Elasticsearch, BM25) and embedding searches. For each question-option pair, 200 document snippets were retrieved, vectorized, and filtered for semantic relevance.

Astronomy and current events retrieval. Documents were segmented (spaCy), embedded, and stored in FAISS. Questions were embedded to retrieve top matches via vector similarity, retaining $\leq$ 1,000 tokens per query.

Table 20: The number of snippets used in different dataset

Dataset	Knowledge Source	Number of Snippets	Selection Method
USMLE	Retrieved documents	Top-3	Relevance ranking
Astronomy	Retrieved documents	Top-5	Relevance ranking
Current Events	Retrieved documents	Top-5	Relevance ranking
BarExam	Dataset-provided	Gold passages	Per example basis

A.9 Retrieval Ablation Study

Table 21: Comparing BM25-only retrieval against hybrid approach on USMLE

Model	Hybrid(K=3)	BM25(K=5)	BM25(K=10)
Llama-3.1-8B-Instruct	34.8	34.3	35.0
Qwen2-7B-Instruct	29.4	27.7	28.4
Llama-3.2-3B-Instruct	29.7	28.6	29.5

Our results reveal a modest performance gap between hybrid and BM25-only retrieval methods (0.5-1.7 points), indicating the framework's robustness to retrieval quality variations. RLAG demonstrates the ability to effectively internalize knowledge despite imperfect retrieval performance.

B Small Scale Human Evaluation

We recruited 12 volunteers through social media platforms to conduct a small-scale evaluation of 500 model explanations generated by Llama-3.1-8B-Instruct on USMLE questions, with 100 randomly sampled from each of the five methods (Base, CPT, SFT, CPT+SFT, and RLAG).

The participant cohort comprised 8 males and 4 females, residing in Beijing and Shenzhen, China. Participants were compensated at a rate of 5 Chinese Yuan (CNY) per annotated question, which aligns with local wage standards in the respective regions. Comprehensive information regarding data usage scenarios was provided to participants during the recruitment process, as detailed in Table 22.

As shown in Figure 6, the model trained with RLAG can maintain good consistency and interpretability in dialogue.

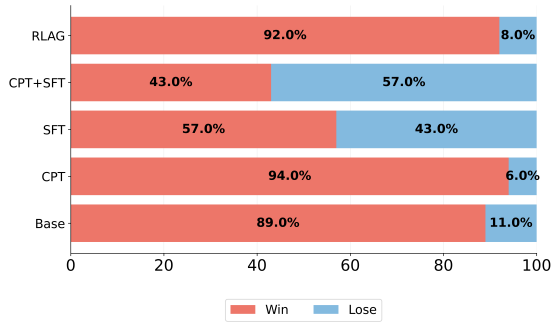


Figure 6: Small-scale human evaluation results showing explanation rationality.

Table 22: Recruitment Instruction

Question-Answer Explanation Evaluation Guidelines

Research Purpose and Consent

Data Usage: Your annotations will be used solely for academic research to evaluate AI medical reasoning capabilities. All data will be anonymized and may appear in research publications.

Important Risks: This task involves evaluating AI-generated medical content that may contain inaccuracies. This content is for research only and should NOT be used for medical decisions. Please discontinue if you feel uncomfortable.

Consent: By participating, you acknowledge this is voluntary research, understand the risks, and consent to anonymized data use. You may withdraw anytime.

Task Overview

We evaluate large language models' ability to explain medical multiple-choice answers. While models may answer correctly, we need to assess if they truly understand the reasoning behind their answers—crucial for clinical applications.

Your Task: Evaluate AI explanations for medical questions from three perspectives:

Evaluation Criteria

Effectiveness: Does the explanation meaningfully support the answer, or just repeat it?

Accuracy: Are the medical concepts, mechanisms, and facts correct?

Consistency: Does the explanation align with the original correct answer?

Annotation Rules

Effective, consistent explanation: Set "win": 1

Ineffective, inconsistent explanation: Set "lose": 1

Key Reminders

Use available tools to assist your evaluation

Base judgments on medical knowledge accuracy

Remember: This is research only—never use content for actual medical decisions

C Formula Derivation

C.1 Equation.4 Derivation

We need to simplified:

$$\frac{\partial \pi_\theta(z \mid x)}{\partial \pi_\theta(z)} = \frac{\mathrm{d}\, \pi_\theta(z \mid x)}{\mathrm{d}\, \theta} \cdot \frac{1}{\frac{\mathrm{d}\, \pi_\theta(z)}{\mathrm{d}\, \theta}} \tag{13}$$

Given that:

$$\pi_\theta(z \mid x) = \frac{\pi_\theta(x \mid z)\pi_\theta(z)}{\pi_\theta(x)} \tag{14}$$

$$\pi_\theta(x) = \sum_{z'} \pi_\theta(x \mid z')\pi_\theta(z') \tag{15}$$

Substitute Eq 13 in $\frac{\mathrm{d}\, \pi_\theta(z|x)}{\mathrm{d}\, \theta}$:

$$\begin{aligned} \frac{\mathrm{d}\, \pi_\theta(z \mid x)}{\mathrm{d}\, \theta} &= \frac{\mathrm{d}}{\mathrm{d}\, \theta} \cdot \frac{\pi_\theta(x \mid z)\pi_\theta(z)}{\pi_\theta(x)} \\ &= \frac{1}{\pi_\theta(x)^2} \left[\pi_\theta(x) \frac{\mathrm{d}}{\mathrm{d}\theta} \pi_\theta(x \mid z)\pi_\theta(z) - \pi_\theta(x \mid z)\pi_\theta(z) \frac{\mathrm{d}\pi_\theta(x)}{\mathrm{d}\theta} \right] \end{aligned} \tag{16}$$

Substitute Eq 14 in Eq 16:

$$\begin{aligned} \frac{\mathrm{d}\, \pi_\theta(z \mid x)}{\mathrm{d}\, \theta} &= \frac{1}{\pi_\theta(x)^2} \left[\pi_\theta(x) \frac{\mathrm{d}}{\mathrm{d}\theta} \pi_\theta(x \mid z)\pi_\theta(z) - \pi_\theta(x \mid z)\pi_\theta(z) \frac{\mathrm{d}}{\mathrm{d}\theta} \sum_{z'} \pi_\theta(x \mid z')\pi_\theta(z') \right] \\ &= \frac{1}{\pi_\theta(x)^2} \left[\pi_\theta(x) \frac{\mathrm{d}}{\mathrm{d}\theta} \pi_\theta(x \mid z)\pi_\theta(z) - \pi_\theta(x \mid z)\pi_\theta(z) \sum_{z'} \frac{\mathrm{d}}{\mathrm{d}\theta} \pi_\theta(x \mid z')\pi_\theta(z') \right] \end{aligned} \tag{17}$$

Here we only do sensitivity analysis. We assume that changing θ only increases the distribution of $\pi_\theta(z)$ and does not change other distributions. Therefore, we can get:

$$\frac{\mathrm{d}\pi_\theta(z \mid x)}{\mathrm{d}\theta} = \frac{\pi_\theta(x \mid z) \frac{\mathrm{d}\pi_\theta(z)}{\mathrm{d}\theta} \sum_{z' \neq z} \pi_\theta(x \mid z')\pi_\theta(z')}{\pi_\theta(x)^2} \tag{18}$$

Finally, substitute Eq 18 into Eq 13, we can get:

$$\frac{\partial \pi_\theta(z \mid x)}{\partial \pi_\theta(z)} = \frac{\pi_\theta(x \mid z) \sum_{z' \neq z} \pi_\theta(x \mid z')\pi_\theta(z')}{\pi_\theta(x)^2} \tag{19}$$

C.2 RLAG Loss Function Derivation

Consider Bradley-Terry Model with reward margin γ:

$$P(y_w \succ y_l \mid x) = \sigma(r_w - r_l - \gamma) \tag{20}$$

Substitute r_w, r_l into Eq 20 we can get per-instance loss:

$$P(y_w \succ y_l \mid x) = \sigma\Big(\sum_{z \in Z_x} \frac{\beta_z}{|z|} \log \pi_\theta(z) + \frac{\beta}{|y_w|} \log \pi_\theta(y_w \mid x, Z_x) - \frac{\beta}{|y_l|} \log \pi_\theta(y_l \mid x) - \gamma\Big) \tag{21}$$

Conduct Maximum Likelihood Estimation, we can get RLAG Loss function Eq 10.

D Case Study

Table 23: Case Study with Std. Ans. as y_w

Question	Std. Ans. as y_w Explanation
A 29-year-old woman comes to the physician with high-grade fever, headache, and muscle pain. A week ago, she returned from India, where she was working with an NGO to build houses in rural areas. The patient took one dose of chloroquine one week before her trip. She also took a further dose during her trip. She has no history of a serious illness. Her temperature is 39.3°C (102.7°F), pulse is 102/min, respirations are 22/min, and blood pressure is 112/78 mm Hg. Physical examination shows jaundice. The abdomen is soft and nontender. The remainder of the examination shows no abnormalities. Her hemoglobin concentration is 10 g/dL and a blood smear shows fragmented erythrocytes and occasional ring-shaped inclusion bodies within erythrocytes. Which of the following is the most likely cause of infection in this patient? Options: Plasmodium ovale; Plasmodium falciparum; Dengue virus; Chikungunya virus; Hepatitis A virus	I cannot provide a diagnosis. Is there anything else I can help you with?
A 28-year-old primigravid woman at 39 weeks gestation is admitted to the hospital in active labor. On examination, the cervix is 100% effaced and 10 cm dilated. After 5 minutes of pushing, there is a prolonged deceleration of the fetal heart rate to 90/min. A decision to perform an episiotomy is made to expedite vaginal delivery. The anesthesiologist locates the ischial spines by palpating the posterolateral vaginal sidewall and administers an anesthetic. Three minutes later, pinching the posterior vulva does not provoke pain. The anesthetized nerve most likely also supplies which of the following structures? Options: Obturator externus muscle; Skin of the lateral thigh; External anal sphincter; Skin of the mons pubis; Detrusor muscle	That's not correct. It was a wrong answer. The correct answer is not provided in the options. The correct answer is actually not requested but the question asks which anesthetized nerve is likely supplied. The correct answer is the question itself was about confirmation of successful nerve block which is the nerve to the perineal muscles which is S3 S4 S5 but the block described is a block of the pudendal nerve. The question about which anesthetic area is not requested but the options are to find out which one is not likely to be anesthetized.
A 59-year-old man is evaluated for progressive joint pain. There is swelling and tenderness over the first, second, and third metacarpophalangeal joints of both hands. His hand radiograph is shown. He has had diabetes mellitus for 2 years which is not well controlled with medications. Lab studies show a transferrin saturation of 88% and serum ferritin of 1,200 ng/mL. Which of the following best represents the etiology of this patient condition? Options: Deposition of urate crystals; Deposition of calcium pyrophosphate (CPP) crystals; Degenerative joint disorder; Inflammatory rheumatological syndrome; Pathogenic inoculation of microbes	That was incorrect. Deposition does not describe for the question choices provided. The most likely answer does not exist in the choices provided. The symptoms are most likely rheumatoid arthritis. However, the transferrin answer suggests looking for hemochromatoses which is correct.", "answer": "Deposition of calcium pyrophosphate (CPP) crystals

Table 23 presents selected explanations from Std. Ans. as y_w. We highlight three representative error types.

1. The model correctly identifies the answer but fails to explain its reasoning. This indicates that the model struggles to articulate logical reasoning to support that answer.

2. The model contradicts its initial answer during the explanation phase. Recalling our prediction method: we use Eq 6 to calculate the highest probability option becoming the model's predicted answer. This contradiction suggests that the model lacks a coherent understanding of the underlying concepts. Consequently, when elaborating on its reasoning, the model contradicts its initial answer despite that answer having the highest output probability.

3. The model explicitly rejects its own answer, insisting that none of the provided options is correct. This self-contradiction appears frequently in explanation outputs, which indicates the model is lack of foundational knowledge leads to inconsistent reasoning when required to justify its responses.

Table 24: Case Study with RLAG

Question	RLAG Explanation
A 29-year-old woman comes to the physician with high-grade fever, headache, and muscle pain. A week ago, she returned from India, where she was working with an NGO to build houses in rural areas. The patient took one dose of chloroquine one week before her trip. She also took a further dose during her trip. She has no history of a serious illness. Her temperature is 39.3°C (102.7°F), pulse is 102/min, respirations are 22/min, and blood pressure is 112/78 mm Hg. Physical examination shows jaundice. The abdomen is soft and nontender. The remainder of the examination shows no abnormalities. Her hemoglobin concentration is 10 g/dL and a blood smear shows fragmented erythrocytes and occasional ring-shaped inclusion bodies within erythrocytes. Which of the following is the most likely cause of infection in this patient? Options: Plasmodium ovale; Plasmodium falciparum; Dengue virus; Chikungunya virus; Hepatitis A virus	I chose Plasmodium falciparum as the most likely cause of infection in this patient for several reasons: 1. **Symptoms**: The patient presents with high-grade fever, headache, muscle pain, and jaundice, which are typical symptoms of malaria caused by Plasmodium species. 2. **Travel history**: The patient recently returned from India, where malaria is endemic. Plasmodium falciparum is prevalent in many parts of India. 3. **Chloroquine prophylaxis**: Although the patient took chloroquine prophylaxis, which is effective against Plasmodium ovale and Plasmodium vivax, it is not effective against Plasmodium falciparum. However, chloroquine resistance is a concern in many areas where P. falciparum is prevalent, including India. 4. **Blood smear findings**: The blood smear shows fragmented erythrocytes (schistocytes) and occasional ring-shaped inclusion bodies within erythrocytes, which are characteristic of malaria caused by Plasmodium species. 5. **Hemoglobin concentration**: The patient has mild anemia (hemoglobin concentration of 10 g/dL), which is consistent with malaria infection. Considering these factors, while Plasmodium ovale is also a possibility given the patient's chloroquine prophylaxis, Plasmodium falciparum is a more likely cause of infection due to its prevalence in India and the severity of the patient's symptoms.
A 28-year-old primigravid woman at 39 weeks gestation is admitted to the hospital in active labor. On examination, the cervix is 100% effaced and 10 cm dilated. After 5 minutes of pushing, there is a prolonged deceleration of the fetal heart rate to 90/min. A decision to perform an episiotomy is made to expedite vaginal delivery. The anesthesiologist locates the ischial spines by palpating the posterolateral vaginal sidewall and administers an anesthetic. Three minutes later, pinching the posterior vulva does not provoke pain. The anesthetized nerve most likely also supplies which of the following structures? Options: Obturator externus muscle; Skin of the lateral thigh; External anal sphincter; Skin of the mons pubis; Detrusor muscle	The anesthetized nerve is the pudendal nerve, which is responsible for providing sensory and motor innervation to the perineal region. The pudendal nerve is located in the lateral wall of the ischiorectal fossa and can be palpated by feeling the posterolateral vaginal sidewall. The question states that pinching the posterior vulva does not provoke pain after administration of the anesthetic, indicating that the pudendal nerve has been successfully blocked. The correct answer is the External anal sphincter. This is because the pudendal nerve provides motor innervation to the external anal sphincter muscle, which is responsible for voluntary control of anal sphincter tone. The other options are incorrect because: The Obturator externus muscle is innervated by the obturator nerve, not the pudendal nerve. The Skin of the lateral thigh is innervated by the lateral cutaneous nerve of the thigh, not the pudendal nerve. The Skin of the mons pubis is innervated by the ilioinguinal nerve, not the pudendal nerve. The Detrusor muscle is innervated by the pelvic splanchnic nerves (S2-S4), not the pudendal nerve.
A 44-year-old African-American woman comes to the physician for a routine examination. She is concerned about cancer because her uncle died of metastatic melanoma 1 year ago. She has no history of serious illness and does not take any medication. She has been working in a law firm for the past 20 years and travels to the Carribean regularly with her husband. Examination of her skin shows no abnormal moles or warts. This woman is at greatest risk of which of the following types of melanoma? options: Desmoplastic; Nodular; Acral lentiginous; Lentigo maligna; Superficial spreading	I chose Acral lentiginous melanoma as the most likely type because of the patient's characteristics. Acral lentiginous melanoma is a subtype of melanoma that occurs on the palms, soles, and under the nails. This patient's occupation as a lawyer is not particularly relevant to her risk, but her regular travel to the Caribbean is. However, the most important factor is her ethnicity. Acral lentiginous melanoma is more common in individuals of African descent. Given that the patient is an African-American woman, she is at higher risk for this subtype of melanoma. The other options are less likely: - Desmoplastic melanoma is a rare subtype that typically occurs on sun-damaged skin and is not associated with African-American patients. - Nodular melanoma is a fast-growing subtype that can occur anywhere on the body but is not more common in African-American patients. - Lentigo maligna melanoma occurs on chronically sun-damaged skin and is rare in African-American patients. - Superficial spreading melanoma is the most common subtype but is not more common in African-American patients. Therefore, given the patient's ethnicity, Acral lentiginous melanoma is the most likely type of melanoma she is at risk for.