Exploring Internal Emphasis for Robust Noisy RAG

Qifeng Lai[1], **Zhiguo Gong**[1*], **Usman Naseem**[2] and **Wei Wang**[3*]
[1]Department of Computer and Information Science, University of Macau, Macau, China
[2]School of Computing, Macquarie University, Sydney, Australia
[3]Engineering Research Centre of Applied Technology on Machine Translation and Artificial Intelligence, Macao Polytechnic University, Macau, China
yc47424@um.edu.mo, fstzgg@um.edu.mo, usman.naseem@mq.edu.au, ehomewang@ieee.org

Abstract

Retrieval-augmented generation (RAG) improves knowledge-intensive QA tasks by incorporating external evidence, yet retrieval remains imperfect and often returns irrelevant or misleading passages. We refer to this as Noisy RAG, where LLMs suffer from inconsistent relevance signals because relevance identification is performed by only a small subset of internal modules (e.g., retrieval heads) and can be easily confounded by other modules. Recent work attempts to enhance the denoising ability of the LLM through external emphasis, prompting the LLM to highlight helpful passages as the emphasis before answering. However, these textual identifications themselves depend on the same inconsistent signals and thus become less reliable. Our key insight is that, because relevance signals reside in only a few modules, emphasis should be applied directly within these modules rather than only at the text level. We therefore propose *selective internal emphasis* to amplify the relevance signals, implemented via a lightweight, plug-and-play signal amplifier that operates inside the LLM. The amplifier performs token- and channel-level selective emphasis during a standard RAG fine-tuning pipeline, with the base LLM frozen. Across four QA benchmarks and three LLM scales, our method consistently improves accuracy and robustness. Furthermore, generalization and interpretability analyses show that the amplifier captures retrieval-related patterns rather than only dataset-specific patterns, enabling more reliable passage-denoising.

1 Introduction

Retrieval-Augmented Generation (RAG) enhances large language models (LLMs) in knowledge-intensive question answering by incorporating external evidence. However, retrieved passages frequently contain irrelevant or misleading information (**Noisy RAG**) despite the continuous improvement of retrievers. In such cases, the LLM's reasoning can be disrupted: the model may temporarily form correct internal estimates yet still produce incorrect answers [Wu *et al.*, 2024; Sun *et al.*, 2024]. These errors, as an instance of hallucination, do not arise from a lack of knowledge for relevance identification, but can be attributed to the *inconsistency of relevance signals inside the model* [Orgad *et al.*, 2024], i.e., those signals can be confounded or overwritten by complex interactions inside the LLM.

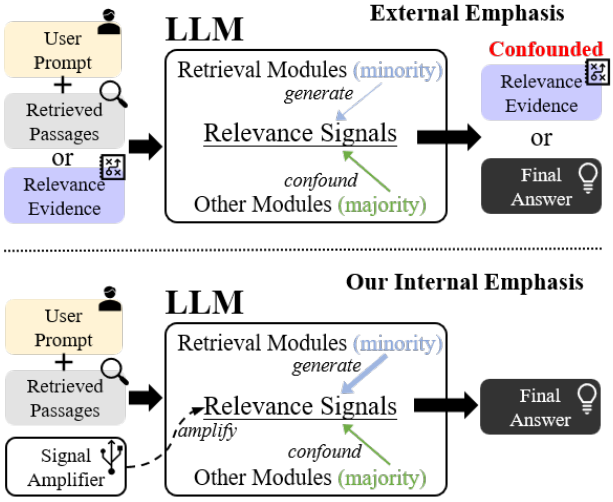

Figure 1: Comparison between our method and external emphasis approaches. Prior methods leverage the LLM to identify helpful passages and then prompt the LLM to answer questions with those identifications. However, identifications also suffer from the inconsistency of the LLM. Our internal emphasis attempts to enhance the influence of key retrieval modules, which mitigate the inconsistency, enhancing the denoising ability of the LLM.

Existing studies typically adopt *external emphasis* to enhance the denoising ability of the LLM [Wei *et al.*, 2025; Wang *et al.*, 2025; Chang *et al.*, 2025], prompting the LLM to first identify helpful passages and then answer based on that identification. Although straightforward, such identifications are themselves susceptible to errors due to the same *inconsistency*: only a small subset of attention heads (retrieval heads), and only early feed-forward modules are responsible for relevance detection inside the LLM [Geva *et al.*, 2023; Wu *et al.*, 2024; Farahani and Johansson, 2024]. In Noisy

*Corresponding authors.

RAG, we also observe that only around 10% feature units in the LLM function as passage denoising (details in §6.4). The relevance signals from these sparse modules can be easily confounded by the majority of other modules [Sun *et al.*, 2024], leading to unreliable external emphasis.

Our key insight is that **relevance identification is handled by only a small subset of modules** within the LLM, and therefore the emphasis should be **applied directly at these modules** to mitigate the inconsistency rather than only at the text level. Motivated by state-editing techniques such as knowledge editing [Ilharco *et al.*, 2023], we introduce **internal emphasis, a selective intervention directly applied to the LLM's internal states**.

Concretely, we treat the post-module intermediate states of attention and feed-forward modules as relevance signals. These signals exhibit two structural dimensions: per-token features corresponding to input token representations, and per-channel features corresponding to atomic feature units within a module. We design a lightweight signal amplifier that emphasizes these signals via token–channel gating during standard RAG fine-tuning with the base LLM frozen. Token gating prioritizes question-relevant tokens, steering internal attention toward key evidence, while channel gating amplifies channels responsible for relevance identification. Unlike mechanistic tools such as activation patching or probing [Zhang and Nanda, 2024; Belinkov, 2022], this approach requires no additional data or controlled experiments. The emphasis magnitude is bounded by variance-calibrated initialization to avoid overwhelming original representations. Empirically, our method achieves strong performance across datasets and model scales, while remaining robust and interpretable by consistently identifying retrieval-related channels and key evidence tokens. The code is available on https://github.com/xiaolai2333/InternalEmphasisForNoisyRAG.

Contributions.

- To our knowledge, we are the first to study Noisy RAG via **selective internal emphasis** due to the observed inconsistency of relevance signals inside LLMs under Noisy RAG.
- We propose a lightweight signal amplifier featuring two-axis gating with variance-calibrated initialization, which enhances key signals without overwhelming them, and is trained within an RAG fine-tuning pipeline while keeping the base LLM frozen.
- Across benchmarks and model scales, our approach consistently surpasses strong baselines and maintains superior performance in cross-dataset and low-data volume settings. Interpretability analyses further demonstrate that the amplifier enables LLMs to more reliably route and filter evidence in the presence of noisy passages.

2 Related Work

2.1 Retrieval Augmented Generation

Retrieval-Augmented Generation (RAG) has been commonly used to ground LLMs in external knowledge [Lewis *et al.*, 2020; Guu *et al.*, 2020; Karpukhin *et al.*, 2020]. Early work has focused on improving retrievers for recall and precision gains [Xiao *et al.*, 2025; Li *et al.*, 2024; Sohn *et al.*, 2024; Yang *et al.*, 2025b; Zhang *et al.*, 2024]. However, recent findings indicate that retrievers remain imperfect and cannot completely filter out irrelevant passages, leading to the Noisy RAG setting. To address this, recent methods introduce external emphasis strategies that reduce the influence of noisy passages during generation. For example, [Wei *et al.*, 2025; Wang *et al.*, 2025] leverage LLM-generated rationales or judgments to perform passage selection before answer generation. MAIN-RAG [Chang *et al.*, 2025] aggregates votes from multiple LLMs to filter retrieved documents without additional training. FilCo [Wang *et al.*, 2023] learns a supervised passage filter, and [Zhu *et al.*, 2024] formulates passage filtering under an information-bottleneck framework.

Despite their differences, these approaches consistently suggest that passage relevance signals are already encoded implicitly in LLMs. Yet, because these signals originate from a small subset of relevance-identification modules, they can be easily confounded by unrelated modules when surfaced in text form. This observation motivates our approach: instead of relying solely on external methods, we aim to extract and amplify these internal signals directly at the modules responsible for relevance identification.

2.2 Locating key retrieval modules

It is important to locate key modules for relevance identification before conducting the emphasis. Prior studies often employ mechanistic-interpretability techniques to locate LLM modules with specific functionality. Activation patching methods score candidate modules by measuring functional changes under patched activations [Zhang and Nanda, 2024; Heimersheim and Nanda, 2024]. ROME, as a knowledge-editing approach, finds where factual associations lie in mid-layer FFNs through causal tracing, then uses low-rank updates to change these associations [Meng *et al.*, 2022]. Patchscopes probe hidden representations by inserting them into diagnostic prompts to elicit natural-language interpretations [Ghandeharioun *et al.*, 2024]. For long-context LLMs, other studies analyze retrieval-aware attention or jointly train retrieval-specific components to suppress irrelevant content during decoding [Qiu *et al.*, 2025]. In the RAG setting, ReDeEP [Sun *et al.*, 2024] decomposes retrieval failures into those stemming from "knowledge FFNs" versus "copying heads" informed by retrieved evidence.

Although these techniques are novel and insightful, they typically require additional datasets, controlled interventions, or specialized diagnostic pipelines. Since experiments of these methods are usually not designed for Noisy RAG, these methods can perform poorly in our setting. Furthermore, iterative knowledge editing has been shown to cause forgetting [Gupta *et al.*, 2024], while probing methods can fail to transfer across datasets [Belinkov, 2022].

Motivated by these limitations, rather than explicitly locating or editing modules via external experiments, we directly learn the relevant feature units inside the LLM within a single RAG fine-tuning pipeline, while keeping the LLM frozen. This allows us to obtain a low-cost, data-driven, and accurate localization of module functionality.

3 Preliminaries

3.1 LLM Internal States

Large decoder-only LLMs consist of a stack of Transformer blocks, each with two primary submodules: multi-head self-attention (MHSA) and a feed-forward network (FFN). Each module transforms representations of tokens layer by layer with different functionalities. Recent work attempts to interpret the functionality of different modules in LLMs rather than treating them as black boxes [Ferrando *et al.*, 2024], called mechanistic analysis.

Attention module. The MHSA module uses queries, keys, and values to attend to preceding tokens; each attention head operates in parallel and may specialize in different types of information. Recent work in RAG settings finds that certain *retrieval heads* consistently attend to and copy relevant external content. Ablating these heads degrades retrieval quality and increases hallucinations, underscoring their causal role in relevance identification [Wu *et al.*, 2024; Sun *et al.*, 2024].

FFN module. FFNs implement *key–value memories* that store latent patterns capable of shaping the model's output distribution, offering insight into how factual knowledge is written into and read out of hidden states during generation [Geva *et al.*, 2021], i.e., knowledge storage. Besides, FFN blocks also contribute to determining contextual relevance and encoding factual associations. Causal-tracing and activation-patching studies identify strong early-layer FFN effects on relevance decisions, suggesting that both attention and FFN modules encode signals essential for filtering noisy passages [Zhang and Nanda, 2024; Geva *et al.*, 2023]. These FFN-stored signals may contribute directly to relevance decisions [Farahani and Johansson, 2024].

3.2 Problem Formulation

Following prior work that uses the Noisy RAG framework [Wei *et al.*, 2025; Wang *et al.*, 2025; Chang *et al.*, 2025], our objective is to improve LLMs' question-answering ability given noisy passages. Formally, let $\mathcal{Q} = \{q_1, q_2, \ldots, q_N\}$ be a collection of N queries. Given a query $q_i \in \mathcal{Q}$, a retriever produces K candidate passages:

$$\mathcal{C}_i = \{c_{i,1}, c_{i,2}, \ldots, c_{i,K}\}, \tag{1}$$

where some of the $c_{i,j}$ may be noisy. We then condition an LLM f_θ on $(q_i, \mathcal{C}_i)$ to produce an output response $\hat{y}_i$:

$$\hat{y}_i = f_\theta(y|q_i, \mathcal{C}_i). \tag{2}$$

The response should contain the answer t_i to the query q_i, despite irrelevant/noisy passages in $\mathcal{C}_i$.

Terminology We use *token* to refer to a position of the input sequence (length T). We use *channel*, also known as a feature unit, to refer to a single unit of the hidden vector (width H). Therefore, each post-module hidden state $h_l \in R^{T\times H}$ can be represented as a matrix with rows indexed by tokens and columns indexed by channels. In attention blocks, a channel corresponds to one dimension of the hidden state after the output projection. In FFN blocks, a channel denotes one neuron's activation after the down projection. The channel gating

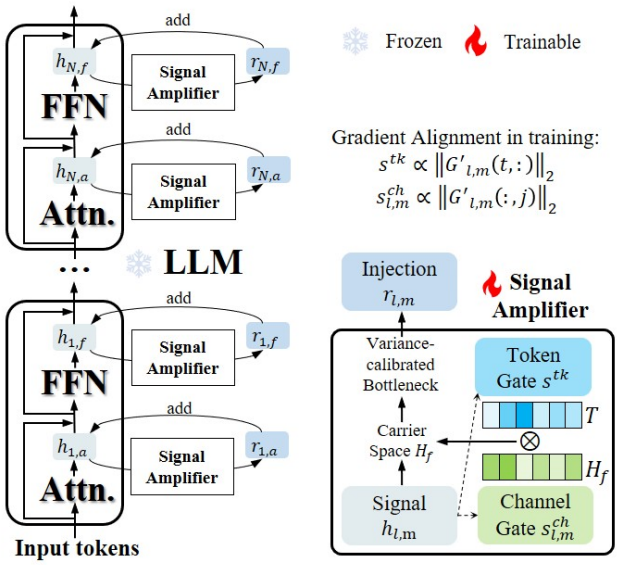

Figure 2: Overview of the signal amplifier for Noisy RAG. The amplifier reads post-module signals and selectively enhances them through token and channel gating. The combination of the carrier space and the variance-calibrated bottleneck ensures robust emphasis without overwhelming the model's original representations. During training, the two-axis gating aligns naturally with gradient guidance, thereby ensuring the effectiveness of our approach.

serves to identify and select retrieval-related modules, similar in spirit to mechanistic analysis, but it accomplishes this automatically within a single RAG fine-tuning process.

4 Method

As discussed, relevance identification signals can be confounded by other non-retrieval modules when such signals are surfaced as text and used as external emphasis [Wei *et al.*, 2025; Wang *et al.*, 2025; Chang *et al.*, 2025]. We therefore move the emphasis *inside* the LLM by operating directly on post-module hidden states, treating them as the relevance signals. We then apply selective emphasis via token–channel gating and write back a small injection. The injection magnitude is constrained by a bottlenecked linear-nonlinear mapping, preventing the internal emphasis from distorting the original distribution. Finally, we analyze why token–channel gating leads to robust learning of passage relevance patterns in the gradient view. The lightweight signal amplifier enables low-cost training. An overview is shown in Figure 2.

4.1 Internal Emphasis

Auditable intervention To realize the internal emphasis on relevance signals, we first need to consider where the signals appear. We define the signals as the post-module, pre-residual output (e.g., attention or FFN output before being added to the residual stream). This isolates a module's contribution prior to mixing with other modules. Then, the signals are emphasized and written back at the same location, making the emphasis auditable and interpretable: we can trace how emphasis on each module influences the QA performance, without

extra mechanistic experiments. Formally, $h_{l,m} \in R^{T\times H}$ are the post-module signals from a module $m \in \{\text{attn}, \text{ffn}\}$ at layer l. The emphasis is computed from $h_{l,m}$ and forms as a small injection $r_{l,m} \in R^{T\times H}$. Then we perform internal emphasis with the adding $r_{l,m}$ to the original signals:

$$\tilde{h}_{l,m} = h_{l,m} + r_{l,m}. \tag{3}$$

The process can be viewed as masked, low-rank residual modulation of $h_{l,m}$, which directly alters the original states, distinguishing it from structure-based methods, such as LoRA-like methods [Hu *et al.*, 2022] and adapters [Houlsby *et al.*, 2019]. Having localized the post-module signals $h_{l,m}$, we next specify where and how to emphasize them.

Token-channel gating As an intervention, an effective emphasis should enhance rather than overwhelm the model's original signals. Intuitively, the emphasis should be placed at the most appropriate positions. We consider two dimensions of $h_{l,m}$, i.e., T for tokens and H for channels. In Noisy RAG, question-relevant tokens in passages should be prioritized; due to the sparsity of retrieval computations, only a subset of channels, i.e., feature units, should be amplified. Naturally, we introduce rank-1 token-channel gating. The token gate $s^{\text{tk}} \in [0,1]^T$ is designed to filter out tokens that are irrelevant to the question and shared across modules, ensuring consistent token emphasis. The channel gate $s^{\text{ch}}_{l,m} \in [0,1]^{H_f}$ is specific to module m and employed to select specific channels in m for retrieval (the reason for the dimension H_f is introduced next). Formally, the rank-1 mask is defined as: $M_{l,m} = s^{\text{tk}}\big(s^{\text{ch}}_{l,m}\big)^\top \in [0,1]^{T\times H_f}$. To ensure low computational cost and easy training, we compute s^{tk} via a lightweight cross-attention, with mean question representation $\bar{h}_{l,m,q}$ as the query and passage token representations $h_{l,m,p}$ as the key and value; we leverage the linear projection with sigmoid as $s^{\text{ch}}_{l,m}$. The emphasis is expected to concentrate on the most related tokens and channels for retrieval with the mask, which further differs from input-agnostic and global-updating methods inside the LLM [Mahabadi *et al.*, 2021; Ben Zaken *et al.*, 2022; Liu *et al.*, 2022].

Magnitude constraint While rank-1 gating localizes where to emphasize, we still need to control the injection magnitude to prevent the overwhelming of original signals. We therefore introduce a bottleneck with variance-calibrated initialization to bound the injection magnitude. We first project $h_{l,m} \in R^{T\times H}$ into a carrier space with dimension $H_f < H$, inspired by [Du *et al.*, 2024], which also enables feature refinement and reduces computational cost. Denoting the projected signals as $z_{l,m} \in R^{T\times H_f}$, we apply the selective emphasis in the carrier space:

$$U_{l,m} = M_{l,m} \odot z_{l,m},$$

where $\odot$ is the element-wise product. Then, we obtain the injection with the bottleneck network:

$$r_{l,m} = W^{\text{up}}\phi(U_{l,m}W^{\text{dn}}) \in R^{T\times H}, \tag{4}$$

where ϕ is GELU activation, $W^{\text{dn}} \in R^{H_f\times d}$ and $W^{\text{up}} \in R^{d\times H}$ are projections with a bottleneck size $d \ll H$. Intuitively, a narrow bottleneck d limits the effective degrees of freedom of the update, and the spread from d to H by a linear projection keeps a small per-channel injection.

We further propose to bound $r_{l,m}$ via variance calibration for bottleneck initialization. Denoting $L_\phi \approx 1$ as the Lipschitz constant of ϕ, the injection admits the operator–Frobenius bound[1]:

$$\|r_{l,m}\|_F \le L_\phi \|W^{\text{up}}\|_2 \|W^{\text{dn}}\|_2 \|U_{l,m}\|_F.$$

We initialize the bottleneck with variance-calibrated Gassian to keep $\|W^{\text{up}}\|_2\|W^{\text{dn}}\|_2$ small:

$$\begin{cases} W^{dn}_{ij} \sim \mathcal{N}\big(0, \sigma^2_{dn}\big), & \sigma_{dn} = \frac{\alpha_{dn}}{\sqrt{H_f+\sqrt{d}}}, \\ W^{up}_{ij} \sim \mathcal{N}\big(0, \sigma^2_{up}\big), & \sigma_{up} = \frac{\alpha_{up}}{\sqrt{H+\sqrt{d}}}. \end{cases}$$

By standard spectral concentration for Gaussian matrices, this yields $E\|W^{\text{dn}}\|_2 \approx \alpha_{dn}$ and $E\|W^{\text{up}}\|_2 \approx \alpha_{up}$, thereby, the expected amplification of the bottleneck satisfies:

$$E\big[\|W^{\text{dn}}\|_2\|W^{\text{up}}\|_2\big] \approx \alpha_{dn}\alpha_{up}. \tag{5}$$

where $\alpha_{dn}\alpha_{up} := C_{init}$ is a dimension-agnostic constant. In addition, $\|U_{l,m}\|_F$ is bounded by the two-axis gating that has values between $[0,1]$. Together, we have $\|r_{l,m}\|_F$ with safe and trainable magnitude, which is more robust than setting upper bounds manually.

4.2 First-Order Gradient Alignment

Since the amplifier is trained within the Noisy RAG pipeline, we leverage the standard cross-entropy as the training loss:

$$\mathcal{L} = E_{(q,C,y)}\big[-\log p(y|q,\mathcal{C})\big]. \tag{6}$$

With the base LLM frozen, $\mathcal{L}$ backpropagates to each emphasis point $h_{l,m}$. The loss gradient $G_{l,m} = \nabla_{h_{l,m}}\mathcal{L} \in R^{T\times H}$ quantifies emphasizing on which tokens and channels at module m are most sensitive to error. Consider a small injection $r_{l,m}$, a first-order Taylor expansion gives $\mathcal{L}(h_{l,m} + r_{l,m}) \approx \mathcal{L}(h_{l,m}) + G_{l,m}\cdot r_{l,m}$. Then, we have:

$$\begin{aligned} \Delta\mathcal{L} &\approx \big\langle G_{l,m}, r_{l,m}\big\rangle, \\ &= \Big\langle G_{l,m}, W^{\text{up}}\phi\big(M_{l,m}\odot z_{l,m}W^{\text{dn}}\big)\Big\rangle, \\ &= \Big\langle W^{\text{up}\top}(G_{l,m}\odot z_{l,m})W^{\text{dn}}, M_{l,m}\Big\rangle, \end{aligned} \tag{7}$$

where $\langle A, B\rangle = \sum_{t,j} A_{t,j}B_{t,j}$ is the Frobenius inner product. We denote $G'_{l,m} := W^{\text{up}\top}(G_{l,m}\odot z_{l,m})W^{\text{dn}}$ as the effective gradient, which integrates the actual loss sensitivity for the emphasis on all tokens and channels. Since $M_{l,m} = s^{\text{tk}}\big(s^{\text{ch}}_{l,m}\big)^\top$, the row-wise norms $\|G'_{l,m}(t,:)\|_2$ drive token-gate updates, and the column-wise norms $\|G'_{l,m}(:,j)\|_2$ drive channel-gate updates:

$$\begin{aligned} s^{\text{tk}} &\propto \Pi_{[0,1]}\big(\|G'_{l,m}(t,:)\|_2\big), \\ s^{\text{ch}}_{l,m} &\propto \Pi_{[0,1]}\big(\|G'_{l,m}(:,j)\|_2\big). \end{aligned} \tag{8}$$

Therefore, the emphasis is aligned with the directions that most reduce cross-entropy, which means the amplifier learns passage-relevance patterns by tuning the gates. That explains why our method has better cross-dataset and low-data-volume robustness (see §6.2)

[1]We bound the two linear projections with spectral norms to measure worst-case amplification.

Dataset	Train	Test	Retriever	Recall@5	Label Form
PopQA	12,868	1,399	Contriever	68.7	Short
TriviaQA	78,785	11,313	Contriever	73.5	Short
NaturalQuestions	79,168	3,610	DPR	68.8	Short
ASQA	4,353	948	GTR	82.2	Long

Table 1: Dataset statistics.

5 Experiment Settings

We focus on denoising under standard open-domain QA and evaluate on four relevant benchmarks: PopQA [Mallen *et al.*, 2023], TriviaQA [Joshi *et al.*, 2017], Natural Questions (NQ) [Kwiatkowski *et al.*, 2019], and ASQA [Stelmakh *et al.*, 2022]. We adopt the off-the-shelf datasets from [Wei *et al.*, 2025]. Each example comprises a question, a ground-truth answer, and five top-ranked Wikipedia passages retrieved with retrievers such as GTR [Ni *et al.*, 2022] and DPR [Karpukhin *et al.*, 2020]). Dataset statistics are summarized in Table 1. Following prior works [Asai *et al.*, 2024; Wei *et al.*, 2025], we report the official correctness metric (`str-em`) for ASQA and accuracy (`acc`), i.e., whether the gold answer appears in the generation, for other datasets.

We consider three LLMs of different scales: Llama-3.2-1B-Instruct [Grattafiori *et al.*, 2024], Qwen3-4B-Instruct-2507 [Yang *et al.*, 2025a], and Llama-3-8B-Instruct [Grattafiori *et al.*, 2024], denoted *Llama-1B*, *Qwen-4B*, and *Llama-8B* respectively. Full-parameter FT, LoRA [Hu *et al.*, 2022], IA3 [Liu *et al.*, 2022], FILCO [Wang *et al.*, 2023], Self-RAG [Asai *et al.*, 2024], RetRobust [Yoran *et al.*, 2023], and InstructRAG [Wei *et al.*, 2025] are baselines. FILCO, Self-RAG, and InstructRAG are implemented via LoRA at q, k, v, o projections as a similar PEFT regime for comparison. IA3 is operated at k, v, and down projections in all attention and FFN modules. We reproduce baselines using official code releases and report their best performance after hyperparameter search. We generate the answer using Greedy Decode for all methods. We set H_f=1024/1280/1536 and d=256/384/512 for *Llama-1B*, *Qwen-4B*, and *Llama-8B*, and the trainable parameter count is around 2.5% of the base LLM. We set $\alpha_{up} = \alpha_{dn} = 1$. Following [Wei *et al.*, 2025], we prompt LLMs to generate answers with the form: `<Retrieved Passages>. Based on the provided information and your own knowledge, answer the question: <Question>.` Ours+Rationale is used in all secondary experiments, where rationale is supervision text indicating which retrieved passages are relevant [Wei *et al.*, 2025].

6 Experimental Results

6.1 Main Results

The main results are summarized in Table 2. Across three LLM backbones and four datasets, our method achieves stable state-of-the-art performance. *Ours+Rationale* attains the best score in 9 out of 12 settings, while *Ours*♠ achieves the top results in the remaining three. These outcomes suggest that our approach is both accurate and robust.

The signal amplifier trained only on ASQA (*Ours*♠) generalizes well to other datasets. On PopQA, TriviaQA, and NQ, the ASQA-trained amplifier consistently outperforms per-dataset training, with average gains of 3.3, 5.9, and 3.4 points across the three LLMs. We attribute this behavior to the supervision signal: ASQA provides long-form answers whose semantics align with passage relevance, enabling the amplifier to learn general relevance patterns rather than dataset-specific cues. In contrast, datasets with short labels offer weaker guidance.

LLM	Method	Datasets			
		ASQA (str-em)	PopQA (acc)	TriviaQA (acc)	NQ (acc)
Llama-1B	Full Fine-tuning	33.0	55.5	64.6	47.9
	LoRA Fine-tuning	34.0	59.8	63.6	47.1
	IA3 Fine-tuning	26.6	50.8	58.4	41.4
	FILCO-LoRA	27.6	40.9	56.6	39.1
	Self-RAG-LoRA	29.1	49.8	60.7	42.7
	RetRobust	31.3	54.9	63.1	45.3
	InstructRAG-LoRA	31.7	58.5	65.2	49.6
	Ours♠	**36.2**	60.4	63.8	50.3
	Ours+Rationale	34.4	**63.0**	**67.2**	**53.5**
Qwen-4B	Full Fine-tuning	37.9	61.2	69.8	50.4
	LoRA Fine-tuning	41.3	59.9	69.8	52.0
	IA3 Fine-tuning	35.3	56.1	64.2	47.9
	FILCO-LoRA	30.1	50.1	68.1	43.5
	Self-RAG-LoRA	34.1	54.6	68.8	45.5
	RetRobust	38.3	59.3	69.9	50.0
	InstructRAG-LoRA	40.2	60.3	70.9	52.1
	Ours♠	**44.6**	63.8	73.1	**58.8**
	Ours+Rationale	44.2	**65.8**	**73.9**	58.2
Llama-8B	Full Fine-tuning*	43.8	61.0	73.9	56.6
	LoRA Fine-tuning	42.6	62.6	74.0	55.7
	Self-RAG-LoRA	37.8	55.1	71.1	43.9
	IA3 Fine-tuning	42.9	63.6	72.1	56.2
	FILCO-LoRA	31.1	52.9	70.3	43.7
	RetRobust*	40.5	56.5	71.5	54.2
	InstructRAG-LoRA	43.7	64.5	75.3	60.1
	Ours♠	43.6	64.7	74.5	58.1
	Ours+Rationale	**44.7**	**66.8**	**76.4**	**61.7**

Table 2: Main results (%). Best and second-best results are shown in bold and underlined. Results with * are reported by [Wei *et al.*, 2025]. Ours♠ indicates training the signal amplifier only on ASQA without rationales; Ours+Rationale is trained or tested per dataset with that dataset's own rationales [Wei *et al.*, 2025] (details in §6.1).

To examine this explanation, we incorporate explicit rationales during training, forming *Ours+Rationale*. The results show that adding rationales usually brings further improvements over *Ours*♠ and gives the strongest overall performance. This supports our claim that the signal amplifier benefits most from supervision signals that highlight relevant passages. The improvement on ASQA is smaller, as its long-form labels already provide similar relevance supervision.

6.2 Robustness Results

Intuitively, learning general relevance patterns should transfer better across Noisy RAG datasets. Figure 3 shows the cross-dataset results. Our method displays strong and steady robustness across datasets. Although out-of-domain scores are naturally lower than each dataset's in-domain baseline, the off-diagonal average reaches 57.4%, which is 10.2 points

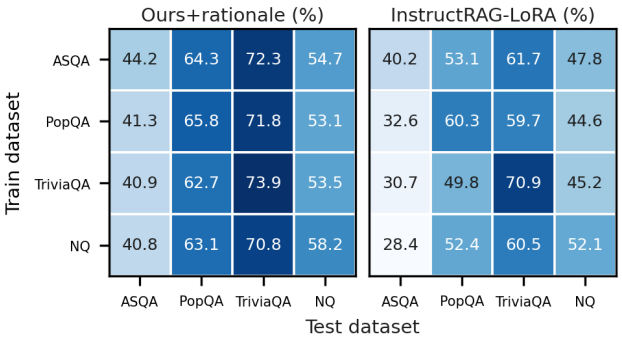

Figure 3: Cross-dataset generalization on Qwen-4B. Off-diagonal cells indicate out-of-domain transfer.

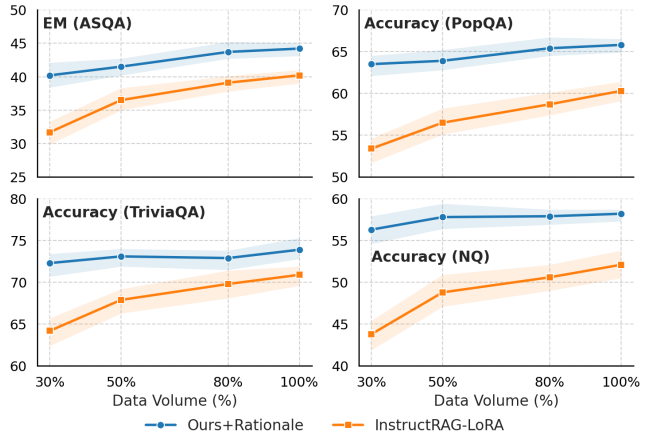

Figure 4: Performance under varying training data volumes. We sample different training samples with five different seeds.

higher than InstructRAG-LoRA. This improvement appears consistently across all out-of-domain settings, suggesting that our approach is less affected by dataset-specific quirks. Learning relevance patterns also requires less training data than learning dataset-specific templates. To check this, we measure performance under different training data sizes, as shown in Figure 4. Across all four datasets, *Ours+Rationale* keeps strong performance even with only 30% of the data and increases slowly as more data is added. In comparison, InstructRAG-LoRA shows a steeper growth curve but starts from a much lower point. At 30% data, our method already leads by a clear margin, and it maintains this advantage even when using the full dataset. The two experiments show that our method mainly **learns passage relevance instead of dataset-specific patterns**, leading to strong performance, and data-efficiency under Noisy RAG.

6.3 Ablation Study

Ablation results in Table 3 show that using both components yields the strongest performance, suggesting that channel gating and token-level attention complement each other. This also indicates that interventions on LLM internal states must be applied in a selective manner; otherwise, they may distort the model's original activation distribution. We also find that the effect of channel gating becomes more pronounced as model size increases. For larger LLMs such as Llama-8B, removing channel gating leads to a noticeably larger performance drop compared with smaller models. This aligns with the view that larger models develop more specialized internal functions and with prior findings that conditional gating mechanisms help improve capacity and generalization at scale [Shazeer *et al.*, 2017; Fedus *et al.*, 2022].

Models	Methods	ASQA	PopQA	TriviaQA	NQ
Llama-1B	with both	**34.4**	**63.0**	**67.2**	**53.5**
	w/o channel gates	↓1.1	↓2.1	↓5.4	↓1.4
	w/o token attention	↓1.7	↓3.0	↓6.5	↓2.9
Qwen-4B	with both	**44.2**	**65.8**	**73.9**	**58.2**
	w/o channel gates	↓2.6	↓2.8	↓3.5	↓2.4
	w/o token attention	↓4.5	↓3.4	↓4.8	↓4.2
Llama-8B	with both	**44.7**	**66.8**	**76.4**	**61.7**
	w/o channel gates	↓4.4	↓5.0	↓7.8	↓6.8
	w/o token attention	↓6.6	↓9.4	↓9.2	↓7.1

Table 3: Ablation results (%). ↓ means the performance drop. We implement *w/o channel gates* and *w/o token attention* by setting all values of channels and tokens in the amplifier to be the mean of the original values.

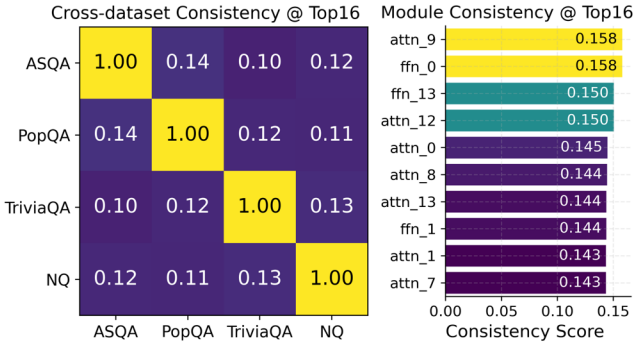

Figure 5: Cross-dataset consistency of channel importance at Top-16 channels in Llama-1B. Left: pairwise Jaccard similarities averaged over modules between datasets. Right: module consistency (Jaccard averaged over all dataset pairs; Top-10 modules shown).

6.4 Interpretablity Analysis

Module Consistency To check whether our method truly identifies the modules responsible for filtering relevant passages, we run a cross-dataset consistency experiment with a simple idea: if these modules are tied to an intrinsic relevance-identification function rather than dataset-specific cues, then the Top-k important channels should largely overlap across different Noisy RAG datasets. For each module in *Llama-1B*, we compute channel importance on each dataset, measure the Jaccard similarity between the Top-k channel sets, and then average the results across modules (Figure 5). The left panel shows that the dataset-wise Jaccard similarity at Top-16 ranges from 0.10 to 0.14 across all dataset pairs. This is 25×–35× higher than the random baseline (around 0.0039 for H=2048, k=16, with an expected random overlap

Qwen-4B	ASQA	PopQA	TriviaQA	NQ
Original	**44.2**	**65.8**	**73.9**	**58.2**
Disable Top 10% channels	↓2.5	↓2.4	↓2.9	↓3.9
Disable Top 50% channels	↓2.7	↓3.0	↓3.3	↓5.1
Switch Top-Bottom 10% channels	↓9.8	↓12.1	↓7.0	↓11.8

Table 4: Results (%) of channel intervention. Disabling certain channels means setting the values of the channels to zero. *Switch Top-Bottom 10% channels* means that the values of the top 10% are switched to the bottom 10% and vice versa.

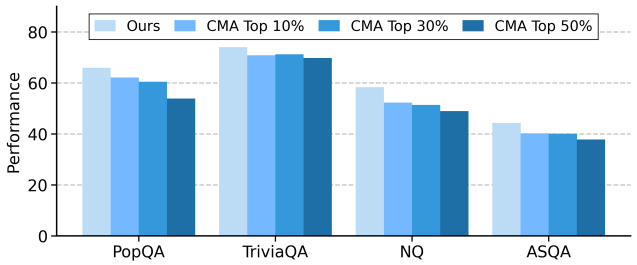

Figure 6: Comparison with activation patching via CMA. We emphasize top attention and feed-forward modules selected by CMA without channel gates, while keeping other settings the same.

of ≈ 0.125 channels). The right panel shows the per-module viewpoint: the most stable modules reach a Jaccard of 0.158, which corresponds to about four overlapping channels out of sixteen on average[2].

Channel Intervention We then intervene in the channels and test the effectiveness of channel selection. Given the trained amplifier, we record its selection of channels of 1k samples in the training set, conduct different interventions, and test it on the testing set. The results are presented in Table 4. From the results, we can find that disabling top channels does reduce the performance, even only the top 10%, indicating that our method enables accurate channel selection. Additionally, if we *Switch Top-Bottom 10% channels*, the performance drops sharply, which further indicates the importance of selective emphasis on channel dimension. Notably, the difference between disabling 10% and 50% channels is not very large, illustrating the sparsity of effective channels for retrieval. This finding aligns with the conclusion that LLMs' retrieval modules are sparse.

Comparison with CMA To further assess whether our channel-selection method is reliable, we compare it against a standard mechanistic-interpretability approach: activation patching with causal mediation analysis (CMA). In brief, CMA works by feeding both a normal input and a counterfactual input to the model, collecting the intermediate representations for the counterfactual, and then replacing the corresponding states in the normal run at a chosen layer. The resulting "corrupted" output is then compared with the original output to estimate the causal influence of that layer. In our setup, we prompt the model to select helpful passages, run CMA to compare the selections under normal and corrupted

[2]For two Top-k sets of equal size, Jaccard $= o/(2k - o)$, so $o \approx 32 \cdot \mathrm{J}/(1 + \mathrm{J})$; with k=16 and $\mathrm{J} = 0.158$, this gives $o \approx 4.3$.

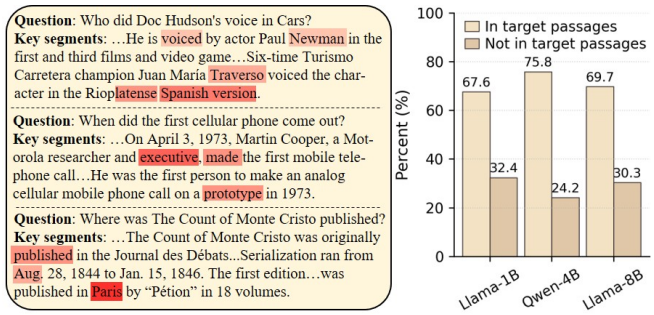

Figure 7: **Left**: three example cases. Key segments are those segments in the passages that contain the gold answer. Tokens with higher relevance scores from the signal amplifier's token attention appear in deeper red. **Right**: the share of top-scoring tokens that fall *inside* the key segments compared with those that fall *outside*.

conditions, and derive the contribution of each attention and feed-forward module. We only remove the sigmoid activation in our amplifier and get rid of explicit channel gating for a fair comparison (the parameter volume in the amplifier remains the same). Figure 6 shows the results. They confirm that our channel-selection method identifies the important components accurately, and they also highlight a limitation of CMA: layer-level analysis tends to be coarse-grained and cannot pinpoint the finer functional structure that channel selection can capture.

Token-level interpretability We examine whether the relevance scores produced by the token-level attention actually highlight tokens that serve as evidence for the correct answer (Figure 7). In the example cases, the signal amplifier consistently points to key tokens—often overlapping directly with the answer spans. Although the highest-scoring token is not always what a human would consider semantically important (e.g., punctuation or articles may occasionally receive high scores), the overall score distribution still concentrates around the answer-bearing region. This trend aligns with the bar plot: approximately 68–76% of the top-scoring tokens fall inside the target passages across different models. These results suggest that placing more emphasis on question-relevant tokens helps the LLM locate and use the correct evidence under Noisy RAG settings.

7 Conclusion

Noisy retrieved passages can lead to retrieval-signal inconsistency inside the LLM. Unlike existing methods that rely on external emphasis, we instead emphasize these inconsistent signals directly where they occur, a strategy we call internal emphasis. We introduce a lightweight signal amplifier, based on token–channel gating and variance-calibrated initialization, which selectively strengthens relevance-related internal signals. Comparison results denote the robust performance of our method. Interpretability analyses further confirm that our approach consistently identifies meaningful retrieval patterns and amplifies internal signals aligned with question-relevant content. Future work will explore richer internal representations and extensions to more challenging settings, such as explicit multi-hop QA and noisier retrieval regimes.

Acknowledgements

The Science and Technology Development Fund, Macau SAR (0126/2024/RIA2, 001/2024/SKL, 0002/2025/EQP), CG-2026, GDST (2020B1212030003, 2023A0505030013), Nansha District (2023ZD001), MYRG-GRG2023-00186-FST-UMDF, MYRG-GRG2025-00234-FST.

References

[Asai *et al.*, 2024] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. 2024.

[Belinkov, 2022] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Comput. Linguist.*, 48(1):207–219, 2022.

[Ben Zaken *et al.*, 2022] Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–9, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[Chang *et al.*, 2025] Chia-Yuan Chang, Zhimeng Jiang, Vineeth Rakesh, Menghai Pan, Chin-Chia Michael Yeh, Guanchu Wang, Mingzhi Hu, Zhichao Xu, Yan Zheng, Mahashweta Das, and Na Zou. Main-rag: Multi-agent filtering retrieval-augmented generation. In *Proc. ACL 63 (Long Papers)*, pages 2607–2622, 2025.

[Du *et al.*, 2024] Xuefeng Du, Chaowei Xiao, and Yixuan Li. Haloscope: Harnessing unlabeled llm generations for hallucination detection. In *Advances in Neural Information Processing Systems*, 2024.

[Farahani and Johansson, 2024] Mehrdad Farahani and Richard Johansson. Deciphering the interplay of parametric and non-parametric memory in retrieval-augmented language models. *arXiv*, 2024.

[Fedus *et al.*, 2022] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.*, 23:1–39, 2022.

[Ferrando *et al.*, 2024] Javier Ferrando, Gabriele Sarti, Arianna Bisazza, and Marta R. Costa-jussà. A primer on the inner workings of transformer-based language models. *arXiv*, 2024.

[Geva *et al.*, 2021] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. *arXiv*, 2021.

[Geva *et al.*, 2023] Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *Proc. EMNLP 2023*, pages 12216–12235, 2023.

[Ghandeharioun *et al.*, 2024] Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. Patchscopes: A unifying framework for inspecting hidden representations of language models. In *Proc. ICML*, 2024.

[Grattafiori *et al.*, 2024] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, and et al. The llama 3 herd of models, 2024.

[Gupta *et al.*, 2024] Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. Model editing at scale leads to gradual and catastrophic forgetting. *arXiv*, 2024.

[Guu *et al.*, 2020] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training. In *Proc. ICML*, pages 3929–3938. PMLR, 2020.

[Heimersheim and Nanda, 2024] Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching, 2024.

[Houlsby *et al.*, 2019] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp, 2019.

[Hu *et al.*, 2022] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[Ilharco *et al.*, 2023] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *ICLR 2023*, 2023.

[Joshi *et al.*, 2017] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proc. ACL 55 (Long Papers)*, pages 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[Karpukhin *et al.*, 2020] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proc. EMNLP 2020 (EMNLP)*, pages 6769–6781, Online, November 2020. Association for Computational Linguistics.

[Kwiatkowski *et al.*, 2019] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *TACL*, 7:452–466, 2019.

[Lewis *et al.*, 2020] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020.

[Li *et al.*, 2024] Xinze Li, Sen Mei, Zhenghao Liu, Yukun Yan, Shuo Wang, Shi Yu, Zheni Zeng, Hao Chen, Ge Yu, Zhiyuan Liu, et al. Rag-ddr: Optimizing retrieval-augmented generation using differentiable data rewards. *arXiv*, 2024.

[Liu *et al.*, 2022] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning, 2022.

[Mahabadi *et al.*, 2021] Rabeeh Karimi Mahabadi, James Henderson, and Sebastian Ruder. Compacter: Efficient low-rank hypercomplex adapter layers, 2021.

[Mallen *et al.*, 2023] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proc. ACL 61 (Long Papers)*, pages 9802–9822, Toronto, Canada, July 2023. Association for Computational Linguistics.

[Meng *et al.*, 2022] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. In *NeurIPS 2022*, 2022.

[Ni *et al.*, 2022] Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith B. Hall, Ming-Wei Chang, and Yinfei Yang. Large dual encoders are generalizable retrievers. In *Proc. EMNLP 2022*, pages 9844–9855, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[Orgad *et al.*, 2024] Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. Llms know more than they show: On the intrinsic representation of llm hallucinations. *arXiv*, 2024.

[Qiu *et al.*, 2025] Yifu Qiu, Varun Embar, Yizhe Zhang, Navdeep Jaitly, Shay B. Cohen, and Benjamin Han. Eliciting in-context retrieval and reasoning for long-context large language models. In *Findings of ACL 2025*, 2025.

[Shazeer *et al.*, 2017] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv*, 2017.

[Sohn *et al.*, 2024] Jiwoong Sohn, Yein Park, Chanwoong Yoon, Sihyeon Park, Hyeon Hwang, Mujeen Sung, Hyunjae Kim, and Jaewoo Kang. Rationale-guided retrieval augmented generation for medical question answering. *arXiv*, 2024.

[Stelmakh *et al.*, 2022] Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. Asqa: Factoid questions meet long-form answers. In *Proc. EMNLP 2022*, pages 8273–8288, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[Sun *et al.*, 2024] Zhongxiang Sun, Xiaoxue Zang, Kai Zheng, Yang Song, Jun Xu, Xiao Zhang, Weijie Yu, and Han Li. Redeep: Detecting hallucination in retrieval-augmented generation via mechanistic interpretability. *arXiv*, 2024.

[Wang *et al.*, 2023] Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md Rizwan Parvez, and Graham Neubig. Learning to filter context for retrieval-augmented generation. *arXiv preprint arXiv:2311.08377*, 2023.

[Wang *et al.*, 2025] Fei Wang, Xingchen Wan, Ruoxi Sun, Jiefeng Chen, and Sercan O" Arik. Astute RAG: Overcoming imperfect retrieval augmentation and knowledge conflicts for large language models. *ACL*, 2025.

[Wei *et al.*, 2025] Zhepei Wei, Wei-Lin Chen, and Yu Meng. InstructRAG: Instructing retrieval-augmented generation via self-synthesized rationales. In *The Thirteenth ICLR*, 2025.

[Wu *et al.*, 2024] Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. *arXiv*, 2024.

[Xiao *et al.*, 2025] Liang Xiao, Wen Dai, Shuai Chen, Bin Qin, Chongyang Shi, Haopeng Jing, and Tianyu Guo. Retrieval-augmented generation by evidence retroactivity in llms. *arXiv*, 2025.

[Yang *et al.*, 2025a] An Yang, Anfeng Li, Baosong Yang, and et al. Qwen3 technical report, 2025.

[Yang *et al.*, 2025b] Shiping Yang, Jie Wu, Wenbiao Ding, Ning Wu, Shining Liang, Ming Gong, Hengyuan Zhang, and Dongmei Zhang. Quantifying the robustness of retrieval-augmented language models against spurious features in grounding data. *arXiv*, 2025.

[Yoran *et al.*, 2023] Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. Making retrieval-augmented language models robust to irrelevant context, 2023.

[Zhang and Nanda, 2024] Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *ICLR*, 2024.

[Zhang *et al.*, 2024] Xiaoming Zhang, Ming Wang, Xiaocui Yang, Daling Wang, Shi Feng, and Yifei Zhang. Hierarchical retrieval-augmented generation model with rethink for multi-hop question answering. *arXiv*, 2024.

[Zhu *et al.*, 2024] Kun Zhu, Xiaocheng Feng, Xiyuan Du, Yuxuan Gu, Weijiang Yu, Haotian Wang, Qianglong Chen, Zheng Chu, Jingchang Chen, and Bing Qin. An information bottleneck perspective for effective noise filtering on retrieval-augmented generation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1069, Bangkok, Thailand, August 2024. Association for Computational Linguistics.