Mitigating Knowledge Conflicts in Retrieval-Augmented Generation via Inference-Time Representation Editing

Dahyun Jung[1] **Jaehyung Seo**[2*] **Heuiseok Lim**[1*]
[1]Department of Computer Science and Engineering, Korea University
[2]Department of Computer Science and Engineering, Konkuk University
{dhaabb55,limhseok}@korea.ac.kr, seojae777@konkuk.ac.kr

Abstract

Retrieval-Augmented Generation (RAG) has been widely adopted to enable Large Language Models (LLMs) to ground their responses in external knowledge sources. Nonetheless, recent studies show that conflicts between the retrieved external knowledge and the model's parametric knowledge can lead to hallucinatory outputs, and this problem is exacerbated when the retrieved documents contain noise. In this work, we propose Conflict-Aware Representation Editing (CARE), an inference-time representation editing method for improving LLM robustness under noisy knowledge conflict settings. CARE learns a latent editing direction from intermediate representations associated with evidence-aligned and conflict-associated erroneous generations, and uses this direction to mitigate conflict-associated activation patterns during inference. We evaluate CARE across Question Answering (QA) benchmarks and LLMs, showing improvements over retrieval-based and conflict-mitigation baselines, especially under noisy retrieval conditions.[1]

1 Introduction

Recent advances in Large Language Models (LLMs) have demonstrated their remarkable capabilities across a wide range of natural language processing tasks, driven by their ability to internalize extensive world knowledge through pretraining on large-scale textual corpora [14, 45, 81]. Despite their impressive performance, LLMs suffer from limitations in reflecting up-to-date information beyond their static training cut-off, and are susceptible to errors and biases present in the training data [47, 40, 21, 56]. These challenges highlight the difficulty of fully encoding all knowledge within parametric models and underscore the necessity for real-time access to external information.

To address these limitations, Retrieval-Augmented Generation (RAG) has emerged as a widely adopted paradigm, in which external knowledge is retrieved and incorporated into the model's response generation process [19, 53]. While RAG offers a promising solution for enhancing the recency and reliability of generated content, it still faces two challenges: (1) *Knowledge Conflict*: Retrieved documents may contradict the model's parametric knowledge, leading to inconsistencies in the generated response. Such conflicts are known to induce hallucinations, wherein the model produces factually incorrect content [62, 72]. (2) *Retrieval Noise*: Due to inherent limitations of retrieval systems, retrieved passages may be irrelevant to the query. This not only hinders the model's ability to accurately assess relevance, but also introduces interference when integrating retrieved content with internal knowledge [9, 60].

*Corresponding authors.
[1]The code is available at https://github.com/ekgus9/CARE.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

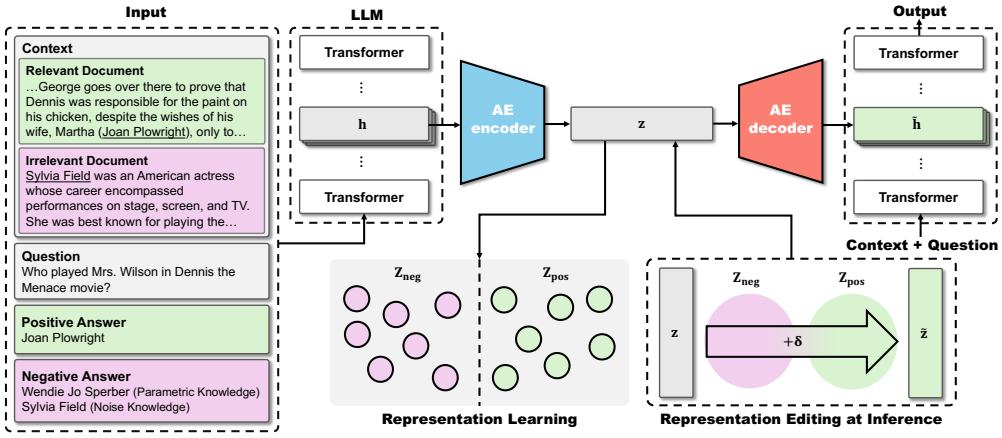


Figure 1: The overall pipeline of CARE. Retrieved context and the question are fed into an LLM to form an internal representation, and an autoencoder encodes it into a latent vector. The model is trained to distinguish latent representations associated with correct versus incorrect answers. At inference time, the learned direction vector is applied to the latent representation of a new input to edit the representation, and the modified vector is then injected back into the LLM to steer it toward a more accurate final output.

In complex scenarios, reliable RAG requires mechanisms that regulate the interaction between parametric knowledge and information retrieved from external sources. Existing approaches that focus primarily on the retrieval stage can improve the quality of retrieved evidence, but they do not directly intervene in the model representations where retrieved and parametric knowledge are integrated [5, 30, 70]. Moreover, methods that adaptively decide whether to use retrieval or adjust generation through contrastive decoding mainly operate at the retrieval-decision or output-distribution level, offering limited control over conflict-associated internal representations [2, 27, 20].

To address this, we aim to build a robust model that handles noisy knowledge conflicts by modifying LLMs' latent representations. Recent studies report that in the vast high-dimensional representation space of LLMs, different semantic features are arranged in nearly orthogonal, disentangled linear subspaces [3, 46, 24, 1]. Motivated by this line of work, we hypothesize that correct and incorrect generations under knowledge conflict can be associated with distinguishable intermediate representation patterns. Based on this hypothesis, we propose **Conflict-Aware Representation Editing (CARE)**, an inference-time representation editing method that learns a latent editing direction from intermediate representations associated with evidence-aligned and conflict-associated erroneous generations. As illustrated in Figure 1, CARE operationalizes this idea with a lightweight autoencoder trained using reconstruction and supervised contrastive objectives. The learned latent direction is applied to selected LLM layers during inference and decoded into a hidden-state correction, allowing CARE to mitigate conflict-associated activation patterns without updating the LLM parameters. This representation-level intervention improves response reliability in noisy RAG scenarios.

To validate the effectiveness of our approach, we perform comprehensive evaluations on four instruction-tuned LLMs: LLaMA-2-7B-Chat, LLaMA-2-13B-Chat [64], LLaMA-3.1-8B-Instruct [12], and Qwen-2.5-7B-Instruct [76], across five Question Answering (QA) benchmarks and two multi-hop QA datasets. CARE improves answer accuracy over retrieval-based, conflict-mitigation, and inference-time truthfulness baselines, suggesting that inference-time representation editing is an effective approach for improving robustness in noisy RAG scenarios.

2 Related Work

Knowledge Conflict LLMs can experience conflicts between their parametric knowledge and retrieved external information, which may lead to incorrect responses [39, 44, 74, 57]. To address knowledge conflicts, [44] introduces a training method that explicitly separates answers based on parametric and contextual knowledge. [69] proposes an instruction-based approach to distinguish between evidence originating from differing perspectives. [79] focuses on identifying and removing

outdated facts within LLMs to ensure compliance with up-to-date contextual information. CAD [57] enhances the effect of relevant context by amplifying the difference in output probabilities with and without context. While effective, these approaches must also account for realistic scenarios where the retrieved information may be noisy or unreliable [10, 74].

Retrieval-Augmented Generation RAG aims to enhance factual accuracy and reflect up-to-date information by integrating external document-based knowledge into language models [19, 53]. Self-RAG [2] allows an LLM to retrieve external knowledge only when necessary, and to reflect on both its generation and the retrieved results in order to improve factual accuracy and quality. Adaptive-RAG [20] dynamically toggles the use of retrieval depending on question difficulty, aiming to optimize response. SURE [30] improves consistency by summarizing retrieved documents and comparing them with candidate answers to assess supporting evidence. While these methods focus on improving input-level consistency, they do not fully resolve conflicts with the model's internal knowledge. CD^2 [27] addresses knowledge conflict in the decoding phase but does not perform adjustment of conflicts within the representation space itself.

Representation Editing Modifying the representation space of LLMs to adjust their knowledge or response behavior has recently gained significant attention [38, 32, 49]. Knowledge Neurons [8] and ROME [42] identify and manipulate specific neurons associated with factual knowledge, injecting or overwriting information directly into the model. While effective for updating long-term knowledge in pretrained models, these methods are less suited for on-the-fly knowledge adjustment during inference. ITI [35] detects and modifies representations in real-time by intervening on attention heads, and TruthX [80] explores and adjusts truthfulness signals in both attention and feed-forward networks. IRCAN [55] identifies context-sensitive neurons and adjusts their weights to encourage context-faithful generation. Building on this line of work, we leverage a trained autoencoder to extract knowledge-alignment signals from the representation space and apply them during inference time, mitigating conflicts between internal and retrieved knowledge.[2]

3 Preliminary

We consider the task of QA, where given an input query q, the goal is to select a set of relevant documents from an external corpus $D = \{d_1, d_2, \ldots, d_n\}$ and generate an answer $a = \{a_t\}_{t=1}^{m}$ based on the retrieved information. In RAG, external sources are utilized by first retrieving documents relevant to q from D and generating the response conditioned on the retrieved content [25]:

$$p_{\mathrm{RAG}} = \prod_{t=1}^{m} p_\theta(a_t \mid a_{<t}, q, R), \quad R = M_D(q), \tag{1}$$

where M_D denotes a retriever that selects relevant documents from D given the query q, and R is the resulting set of retrieved documents. $a_{<t}$ represents the partial response sequence preceding the t-th token, and θ denotes the parameters of the language model.

However, when the model's parametric knowledge conflicts with the content of the retrieved documents R, it may generate incorrect responses [62, 72]. Additionally, if the retrieved set R includes irrelevant information, the injected noise can further degrade output quality [9, 60]. In standard RAG settings, retrieved documents are treated as additional conditioning input, with limited mechanisms for detecting or resolving conflicts with the model's internal knowledge [39, 62]. This setup implicitly assumes that the model will always trust the retrieved documents, without prioritizing among potentially conflicting sources. Thus, beyond naive retrieval, there is a need for mechanisms that explicitly regulate the knowledge interaction within the model's internal representation space. Such mechanisms should serve to preemptively mitigate conflicts and guide generation toward responses grounded in the most reliable source.

4 Methodology

We aim to mitigate knowledge conflicts in RAG scenarios and promote reliable, document-grounded response generation. We introduce a conflict-aware representation adjustment mechanism, enabling

[2] An extended discussion of related work on mechanistic interpretability and additional analysis comparing our approach with prior work are provided in Appendix A.

real-time modulation of internal representations. Specifically, we train an autoencoder to reconstruct a separable latent space from intermediate representations extracted from the model layers, capturing the differences between cases where correct and incorrect answers are generated. At inference time, this learned structure adjusts representations, thereby reducing representational distortion under knowledge conflict and guiding the model toward more accurate responses.

4.1 Layer-wise Autoencoder Learning

The primary objective of the autoencoder is to map the internal representations of the LLM into a latent space via the encoder and reconstruct them through the decoder [68]. We train the autoencoder to preserve hidden-state information while encouraging representations associated with correct and incorrect generations to become more distinguishable (See Appendix B for AUC; §4.2 for δ).

To train the autoencoder, we construct training samples using existing QA datasets. For each query q, we generate two types of samples: positive and negative. Each sample shares the same system/user instructions, query, and retrieved documents; only the response differs. The positive answer a^+ corresponds to the ground-truth answer. We construct the negative answers from conflictual responses that are generated internally by the model, in order to capture its internal conflict patterns. For each negative answer a^-, we consider the following two types in equal proportions[3]:

- **Parametric Answer**: An incorrect answer generated from the model's parametric knowledge without access to retrieved evidence. We obtain these samples by prompting the model with only the question and collecting responses that do not match the ground-truth answer.
- **Noise Answer**: An incorrect answer generated even when relevant documents (i.e., documents that provide evidence supporting the ground-truth answer) are retrieved, typically caused by noisy information. That is, we collect cases where the retrieved documents contain the correct evidence, but the model misuses or overlooks it and answers incorrectly.

The autoencoder is trained independently of the LLM parameters and operates externally, tailored to the representation characteristics of each layer. At each layer, a token representation $h_t \in \mathbb{R}^d$ (where t indexes the token position in a sequence of length T) is compressed into a smaller latent vector $z_t \in \mathbb{R}^{d'}$ through a multi-layer MLP encoder, i.e., $z_t = \mathcal{E}(h_t)$, where $\mathcal{E}(\cdot)$ denotes the encoder network. This process is designed to suppress irrelevant information while preserving essential semantic features [15, 31]. As a result, representations that vary based on context are mapped to distinct regions in the latent space [8]. This separation allows the identification of knowledge conflict regions, where a single representation reflects conflicting knowledge depending on context and diverges in latent space. The decoder, also implemented as a multi-layer MLP, reconstructs $\hat{h}_t$ from z_t to match the original dimensionality of h_t.

We design the training process around two objectives:

Reconstruction Loss The model should reconstruct the input representation. By minimizing the L2 distance between the original input h and the reconstructed output $\hat{h}$, we encourage the model to retain useful features in the compressed latent space. This objective is captured using a mean squared error (MSE)–based reconstruction loss:

$$\mathcal{L}_{\text{recon}} = \frac{1}{T} \sum_{t=1}^{T} \|h_t - \hat{h}_t\|_2^2. \tag{2}$$

Supervised Contrastive Loss We aim to separate the representations of correct and incorrect responses, making them distinguishable in the latent space. To achieve this, we incorporate a supervised contrastive loss. For each training sample, we maintain both positive and negative representations and encourage the model to embed positives closer together while pushing negatives further apart. This objective is formulated as:

$$\mathcal{L}_{\text{supcon}} = \sum_i -\log \frac{\sum_{z^+ \in Z_i^+} \exp\big(\text{sim}(z_i, z^+)/\tau\big)}{\sum_{z^+ \in Z_i^+} \exp\big(\text{sim}(z_i, z^+)/\tau\big) + \sum_{z^- \in Z_i^-} \exp\big(\text{sim}(z_i, z^-)/\tau\big)}, \tag{3}$$

[3]We include only negative samples for which the model response is clearly incorrect, as determined by an F1 score of 0. We further validate the composition of negative answers in Appendix E.1.

where sim denotes the cosine similarity, τ is a temperature parameter, Z_i^+ is the set of embeddings that share the same class label with z_i (excluding z_i itself), and Z_i^- is the set of embeddings that belong to different classes. The loss encourages an anchor representation z_i to be closer to its positive samples Z_i^+ while being pushed apart from negative samples Z_i^-.

The losses are combined into the final objective function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{supcon}}. \tag{4}$$

4.2 Inference-time Activation Adjustment

The trained autoencoder is incorporated into the model's forward pass during inference, enabling dynamic correction of latent representations as responses are generated. To achieve this, we identify a direction vector δ in the latent space that mitigates knowledge conflicts. Specifically, $\mathbb{Z}_{\text{pos}}$ and $\mathbb{Z}_{\text{neg}}$ denote the average latent representations of positive and negative samples, respectively, and the direction vector δ is defined as their difference:

$$\mathbb{Z}_{\text{pos}} = \frac{1}{|D_{\text{pos}}|} \sum_{z \in D_{\text{pos}}} z, \quad \mathbb{Z}_{\text{neg}} = \frac{1}{|D_{\text{neg}}|} \sum_{z \in D_{\text{neg}}} z, \quad \delta = \mathbb{Z}_{\text{pos}} - \mathbb{Z}_{\text{neg}}, \tag{5}$$

where D_{pos} and D_{neg} denote the sets of latent representations associated with positive and negative training samples, respectively.

We select the top five layers with the most well-separated latent spaces for representation adjustment[4]. For each incoming input, we extract the intermediate representation h_t from each selected layer and pass it through the encoder of the trained autoencoder. To incorporate conflict-mitigation signals, the latent vector is shifted along the learned direction δ, yielding the adjusted latent code $\tilde{z}_t = z_t + \delta$. The corrected latent representation $\tilde{z}_t$ is then passed through the decoder to produce a residual correction term, which is added back to the original representation:

$$\tilde{h}_t = h_t + \lambda \cdot \mathcal{D}(\tilde{z}_t), \tag{6}$$

where $\mathcal{D}(\cdot)$ is the decoder that maps the edited latent to a residual vector in the hidden space, and λ is a hyperparameter that controls the strength of the correction. In summary, `CARE` improves the robustness of RAG responses by adjusting the model's internal representations during inference, using learned conflict-aware signals.

5 Experiments

5.1 Experimental Setup

Datasets and Metrics To comprehensively evaluate the effectiveness of `CARE`, we conduct experiments on seven QA datasets. For standard QA, we use Natural Questions (NQ) [33], TriviaQA [28], PopQA [41], SQuAD [51], and NQ-SWAP [39]. For multi-hop QA, where reasoning over multiple documents is required, we include HotpotQA [78] and 2WikiMultiHopQA (2WIKI) [16]. We use NQ to train our method, a widely used open-domain QA benchmark consisting of questions extracted from real user search logs and covering a broad range of answer types. During training, we use only the training split of NQ[5]; for evaluation, we use only the test split. We apply a preprocessing step to ensure no overlap between training and test samples. The remaining datasets are used to assess the generalizability of our method. For all datasets, we prioritize using the official test set; if unavailable, we fall back on the evaluation split. We randomly sample 600 examples from each dataset for evaluation [30]. To measure the accuracy of generated responses, we report Exact Match (EM) and F1 scores[6]. Further details on the datasets are provided in Appendix C.1.

[4]This is measured by the area under the ROC curve (AUC), and further details are provided in Appendix B. Experimental validation of the number of selected layers is provided in §5.3.

[5]Appendix E.2 analyzes a target-trained variant that learns the editing direction from each target dataset.

[6]EM and F1 are valid for assessing response correctness and evidence integration in RAG, while finer effects of knowledge conflicts are examined in §5.3.

Table 1: Performance across QA datasets for various methods and LLM backbones. Best results per model group are in **bold**. * denotes statistical significance with $p < 0.05$ compared to With Retrieval using a paired t-test. NQ is evaluated as in-domain, while the remaining datasets are used to evaluate out-of-domain performance. Self-RAG and SPARE both use LLaMA-2-7B as their backbone. PK denotes *steer to use parametric knowledge*, and CK denotes *steer to use contextual knowledge.*

Method	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	
LLaMA-2-13B-Chat													
No Retrieval	13.33	19.27	33.67	39.88	9.67	19.52	6.67	11.11	11.33	18.42	14.50	18.79	18.01
With Retrieval	29.67	44.00	42.00	57.84	25.33	38.89	19.17	31.90	16.67	28.97	16.00	26.04	31.37
LLMLingua	26.17	39.19	40.67	54.75	24.83	37.96	15.50	28.66	16.83	28.85	16.00	24.70	29.51
Selective-Context	21.17	34.01	33.67	48.30	13.33	29.07	11.00	23.46	13.67	24.61	14.00	23.74	24.17
SURE	29.17	42.21	**50.67**	59.48	35.50	42.72	19.33	29.75	**19.17**	29.01	18.67	27.06	33.56
Opin	15.33	24.14	27.50	43.38	12.83	22.62	7.17	15.13	7.50	16.18	3.33	16.76	17.66
CAD	29.00	43.17	41.00	57.25	23.50	37.34	17.50	30.85	16.17	28.63	14.83	25.15	30.37
IRCAN	33.67	46.36	47.50	59.16	**37.83**	**44.37**	18.17	31.65	19.00	29.19	**19.50**	28.58	34.58
CARE	**34.00***	**47.09***	49.00*	**62.23***	33.50*	43.37*	**19.83**	**33.17***	18.33*	**30.26***	17.17*	**28.71***	**34.72**
LLaMA-3.1-8B-Instruct													
No Retrieval	26.67	37.53	52.17	60.15	24.17	29.17	12.00	21.39	19.00	28.06	15.83	27.18	29.44
With Retrieval	35.83	49.64	58.83	68.72	39.67	48.67	29.50	40.58	27.50	38.41	14.33	25.06	39.73
LLMLingua	34.33	47.38	58.17	67.90	37.00	44.89	27.00	38.40	27.33	38.29	**22.83**	**30.82**	39.53
Selective-Context	32.67	45.71	57.83	67.14	36.00	42.89	23.00	32.96	25.50	35.19	17.67	26.32	36.91
SURE	36.17	51.14	48.00	62.28	**41.50**	47.98	24.33	37.15	19.17	32.52	9.00	15.49	35.39
Opin	38.17	47.78	58.83	68.34	38.00	43.69	30.00	39.72	25.00	33.77	12.00	21.18	38.04
CAD	36.67	49.60	57.67	68.47	38.83	48.22	28.50	40.41	28.33	39.13	16.83	26.99	39.97
IRCAN	37.83	49.55	50.67	61.56	38.17	45.82	21.17	35.67	23.17	33.65	9.17	19.71	35.51
CARE	**38.50***	**51.31***	**59.67***	**69.96***	40.33*	**48.84**	**30.33***	**41.85***	**29.50***	**40.36**	18.33*	27.51*	**41.37**
Qwen-2.5-7B-Instruct													
No Retrieval	15.83	24.14	43.50	49.74	17.17	21.56	11.33	20.19	20.33	28.40	24.83	30.65	25.64
With Retrieval	37.00	49.12	61.67	69.91	40.17	47.59	31.17	41.61	29.17	40.13	25.00	32.06	42.05
LLMLingua	30.00	41.78	57.33	66.95	35.17	42.91	24.50	35.60	27.67	38.00	26.67	32.15	38.23
Selective-Context	31.50	44.26	56.33	64.47	34.00	40.09	21.67	32.23	25.33	35.13	24.67	31.66	36.78
SURE	**45.17**	**55.17**	62.00	69.60	**42.67**	47.67	30.67	38.78	27.00	36.14	17.67	23.60	41.35
Opin	37.50	49.00	61.00	69.22	41.50	47.97	33.00	42.17	30.17	40.30	25.67	31.99	42.46
CAD	37.00	49.00	61.33	69.61	40.33	47.81	31.00	40.87	29.67	40.51	25.33	32.63	42.09
IRCAN	36.33	47.91	60.17	68.63	40.33	47.39	29.67	40.33	29.17	39.26	24.00	31.60	41.23
CARE	40.17*	51.37*	**62.17**	**70.37**	41.67*	**48.43***	**34.17***	**44.29***	**32.00***	**42.84***	**28.00***	**35.73***	**44.27**
LLaMA-2-7B-Chat													
No Retrieval	12.50	22.04	37.33	46.40	6.33	15.09	6.33	15.09	15.33	24.08	**25.17**	**31.67**	21.45
With Retrieval	16.33	29.24	42.33	55.65	28.50	38.05	8.33	20.18	15.33	25.87	14.83	26.39	26.75
Self-RAG	**40.33**	**48.54**	41.50	56.56	22.17	34.38	16.00	28.73	15.00	28.32	12.67	26.03	30.85
CtrlA	13.72	18.28	25.25	33.94	12.76	13.16	**43.73**	**47.09**	**19.66**	27.10	21.40	27.21	25.28
SPARE_{PK}	26.67	35.41	50.33	58.70	34.83	39.48	14.33	23.33	7.00	10.13	15.67	18.49	27.86
SPARE_{CK}	4.50	7.53	9.00	12.75	1.83	3.97	0.83	2.84	17.50	24.29	20.50	27.73	11.11
ITI	0.33	7.87	2.83	14.49	2.33	11.85	0.17	6.69	0.50	5.82	0.00	7.25	5.01
TruthX	15.00	29.39	43.83	58.04	32.83	41.55	10.17	23.12	15.33	27.10	13.00	27.28	28.05
CARE	25.33*	37.94*	**53.17***	**62.49***	**40.67***	**46.51***	14.17*	26.63*	19.50*	**28.95***	21.50*	30.28*	**33.93**

Baselines To evaluate the generalizability of our approach, we consider our method using four instruction-tuned language models: LLaMA-2-7B-Chat [64], LLaMA-2-13B-Chat [64], LLaMA-3.1-8B-Instruct [12], and Qwen-2.5-7B-Instruct [76]. We evaluate thirteen baselines, categorized into four groups: (1) *Retrieval Usage*: No Retrieval, With Retrieval; (2) *RAG Methods*: Self-RAG [2], LLMLingua [22], Selective-Context [36], SURE [30], CtrlA [17]; (3) *Inference-Time Truthfulness Methods*: ITI [35], TruthX [80]; (4) *Conflict Resolution Methods*: Opin [83], CAD [57, 27], IRCAN [55], SPARE [82]. However, several baselines are tied to specific backbone models or publicly released checkpoints, which makes a fully controlled comparison across all LLM backbones infeasible. Self-RAG is based on LLaMA-2-7B, while ITI and TruthX provide publicly available variants trained for LLaMA-2-7B-Chat. Similarly, CtrlA and SPARE are limited to their supported model settings. Therefore, we evaluate these baselines under their corresponding supported backbones and report them in LLaMA-2-7B-Chat. Further details on the baselines are provided in Appendix C.2.

Implementation Details All models are implemented using HuggingFace[7], and we utilize the following checkpoints: 'meta-llama/Llama-2-7b-chat-hf', 'meta-llama/Llama-2-13b-chat-hf', 'meta-

[7] https://huggingface.co/

llama/Llama-3.1-8B-Instruct', and 'Qwen/Qwen2.5-7B-Instruct'. Training is performed using 8 NVIDIA RTX A6000 48GB GPUs with the AdamW optimizer, 1×10^{-4} learning rate, and 10 epochs. For retrieval, we adopt E5-base-v2 [67] as the default retriever, using the 'intfloat/e5-base-v2' checkpoint. We retrieve five passages per query from the DPR version of the Wikipedia dataset (December 2018 snapshot)[8]. For additional experiments, BM25 [52] is implemented using Pyserini [37]. The latent dimension z of the autoencoder is set to one-quarter of the hidden size, following [80], and is implemented as a 2-layer MLP. The editing strength λ is tuned for each model: 0.3 for LLaMA-2-7B-Chat and LLaMA-3.1-8B-Instruct, and 0.5 for both LLaMA-2-13B-Chat and Qwen-2.5-7B-Instruct. Most baseline experiments are conducted using FlashRAG [25]. Further implementation details are provided in Appendix C.3, and the prompt templates used are provided in Appendix C.4.

5.2 Experimental Results

Table 1 presents a quantitative evaluation of CARE across various instruction-tuned LLM backbones and QA benchmarks[9]. Overall, across different LLMs, CARE consistently achieves the highest average performance in RAG settings. These results lead to the following findings:

Existing RAG optimization and conflict-mitigation methods show uneven robustness across benchmarks. Methods such as LLMLingua and Selective-Context reduce redundant context through compression or selection, but do not directly address conflicts between retrieved evidence and parametric knowledge. SURE, Self-RAG, and CtrlA achieve strong results on specific datasets, yet their gains are less consistent across benchmarks, particularly in multi-hop settings. Likewise, conflict-mitigation methods such as CAD and IRCAN can degrade under noisy retrieval, while Opin shows strong performance on Qwen-2.5-7B-Instruct but substantially weaker results on LLaMA-based models. SPARE tends to favor parametric knowledge, and truthfulness-oriented methods such as TruthX and ITI also show limited robustness in noisy, conflict-prone RAG settings. Overall, these results suggest that existing retrieval-optimization and conflict-mitigation strategies do not consistently provide robust improvements across datasets and model families.

CARE improves average performance across noisy retrieval settings. CARE achieves the highest average performance across all model families. These results suggest that the learned representation-editing direction can provide gains across different LLM backbones. Although CARE is trained only on NQ, it also improves average performance on out-of-domain datasets, indicating that the learned editing direction is not limited to the training dataset. Notably, CARE consistently improves over the standard With Retrieval baseline across all models and datasets, including single-hop and multi-hop QA benchmarks. Since these improvements are statistically significant in most cases, the results demonstrate consistent conflict-mitigation benefits in noisy RAG settings.

CARE remains effective under explicit knowledge conflict and retrieval noise. We evaluate CARE on NQ-SWAP [39], where the retrieved context supports an answer conflicting with the model's parametric knowledge. In addition, we introduce NOISY-NQ-SWAP as a stress-test setting by adding retrieved noise contexts alongside the gold context[10]. As shown in Figure 2, CARE outperforms the compared baselines across the evaluated metrics. In NOISY-NQ-SWAP, where noisy contexts are intro-

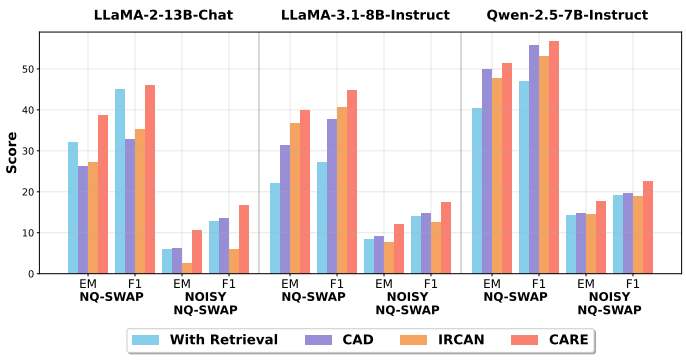


Figure 2: Performance under explicit knowledge conflict. We compare CARE with conflict-mitigation baselines (CAD, IRCAN) and With Retrieval on NQ-SWAP and NOISY-NQ-SWAP.

[8]Since our pipeline uses a standard retriever, retrieval noise naturally arises from irrelevant, insufficient, or misleading documents. We do not artificially inject synthetic noise, reflecting a standard retrieval setting.

[9]Additional experimental results are provided in Appendix D, including robustness analyses across different random seeds, evaluations with Qwen2.5-32B-Instruct [76], Gemma-3-12B-IT [29] and Qwen3-4B-Thinking [75], and experiments on additional tasks and domains.

[10]The noise contexts are constructed by retrieving the top-4 documents. To avoid unintended overlap with the original answer in NQ-SWAP, we filter out contexts containing the original answer.

Table 3: Ablation results across four QA datasets. The values in parentheses indicate performance changes relative to the full method. $\mathcal{L}_{\text{recon}}$ denotes the reconstruction loss, and $\mathcal{L}_{\text{supcon}}$ represents the supervised contrastive loss. δ is the conflict mitigation direction vector.

Method	NQ		TriviaQA		PopQA		SQuAD	
	EM	F1	EM	F1	EM	F1	EM	F1
CARE	38.50	51.31	59.67	69.96	40.33	48.84	30.33	41.85
w/o $\mathcal{L}_{\text{recon}}$	0.50 (↓38.00)	3.08 (↓48.23)	1.00 (↓58.67)	5.63 (↓64.33)	0.83 (↓39.50)	3.46 (↓45.38)	0.67 (↓29.67)	3.21 (↓38.64)
w/o $\mathcal{L}_{\text{supcon}}$	37.17 (↓1.33)	49.33 (↓1.98)	59.17 (↓0.50)	69.12 (↓0.84)	39.67 (↓0.67)	48.45 (↓0.40)	29.50 (↓0.83)	40.75 (↓1.10)
$-\delta$	28.33 (↓10.17)	43.51 (↓7.80)	48.50 (↓11.17)	62.08 (↓7.88)	28.83 (↓11.50)	41.24 (↓7.62)	24.50 (↓5.83)	36.68 (↓5.17)
Random δ	22.17 (↓16.33)	36.86 (↓14.45)	36.33 (↓23.33)	50.81 (↓19.15)	19.50 (↓20.83)	31.56 (↓17.28)	17.33 (↓13.00)	29.80 (↓12.05)

duced, CARE maintains higher performance than both conflict-mitigation baselines and With Retrieval. These results suggest that inference-time representation editing can improve robustness under explicit knowledge conflict and retrieval noise.

5.3 Analysis

We conduct a series of analysis experiments to investigate the impact of various components in our method. We use NQ and LLaMA-3.1-8B-Instruct for analyses, considering space constraints.

Transition Analysis of External Knowledge To more rigorously assess how external knowledge affects model responses, we conduct a transition-based analysis of answer correctness, as shown in Table 2. We report the average EM across NQ, TriviaQA, PopQA, and SQuAD. Retention denotes cases where the model's parametric knowledge remains correct and stable even after incorporating noisy external documents. Resolution refers to cases where the model initially fails to answer correctly but succeeds once external knowledge is added, indicating successful conflict resolution. Corruption captures cases where a correct parametric answer is overturned into an incorrect one after external knowledge is introduced, reflecting cases where retrieved context harms an already correct response. Retention score is almost identical to that of With Retrieval, indicating that, unlike other methods, it very rarely degrades performance on instances where the model already produces the correct answer. At the same time, CARE achieves the highest Resolution, demonstrating its ability to supplement incomplete parametric knowledge with retrieved information. The lowest Corruption indicates that CARE does not corrupt answers that were already correct based solely on parametric knowledge, even in the presence of retrieval-induced noise. These results suggest that CARE improves accuracy by effectively integrating knowledge while minimizing side effects. We provide a fine-grained conflict-type analysis in Appendix E.3.

Table 2: Transition-based analysis of how external knowledge affects model responses.

Method	Retention ↑	Resolution ↑	Corruption ↓
With Retrieval	**5.83**	35.13	6.17
SURE	4.92	32.58	7.08
IRCAN	5.17	31.79	6.83
CARE	5.79	**36.42**	**6.13**

Visualization of Knowledge Conflict Space Figure 3 shows the 2D projection of positive and negative sample representations using PCA. The visualization reveals that the two groups form separable clusters in the representation space, providing qualitative evidence that the learned latent space distinguishes evidence-aligned generations from conflict-associated erroneous generations. The positive samples exhibit a more compact structure, suggesting more consistent representations under evidence-aligned generation.

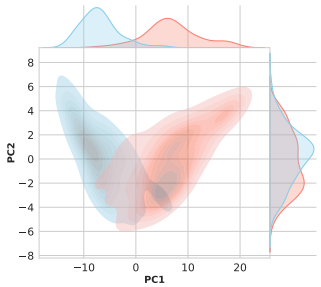


Figure 3: 2D projection of representations for positive (blue) and negative (red) samples.

Ablation Study Table 3 presents ablation studies assessing the contribution of each component within CARE[11]. Excluding the reconstruction loss $\mathcal{L}_{\text{recon}}$ results in a substantial drop in both EM and F1 across all datasets. This indicates that preserving the input-output consistency in the autoencoder is critical for learning stable internal representations. Removing the supervised contrastive loss $\mathcal{L}_{\text{supcon}}$ also leads to performance degradation, confirming that contrastive

[11] Additional ablation studies on the autoencoder are provided in Appendix E.4.

learning helps distinguish knowledge-aligned and conflicting representations, thereby enhancing representational robustness.

To evaluate the role of the direction vector δ, we reverse the direction ($-\delta$) or replace it with a random vector (Random δ). Performance drops with $-\delta$, suggesting that the method's effectiveness depends on using the correct directional adjustment in latent space. Interestingly, the random vector causes a larger performance drop than the inverse vector. While $-\delta$ steers the latent representation towards intensified knowledge conflict, reducing factual accuracy, the representation remains within the knowledge-relevant subspace, preserving the model's language generation capabilities. Conversely, a random vector displaces the latent representation into arbitrary dimensions orthogonal to the knowledge conflict. This displacement acts as noise that disrupts the underlying representational structure essential for coherent text generation.

Finally, using a controlled dataset with identical output tokens, the two conditions remain distinguishable with an AUC of 0.93, indicating that the learned representations capture conflict-sensitive information beyond lexical differences. Detailed analysis is provided in Appendix E.5.

Effect of Retrieval Settings Figure 4 compares performance under different retrieval strategies (E5 and BM25) and varying numbers of retrieved documents. E5 generally yields stronger downstream QA performance than BM25, but CARE consistently improves over the corresponding With Retrieval baseline under both retrievers. This suggests that the benefit of CARE is not tied to a specific retrieval method. Performance is relatively similar when only one document is retrieved ($k = 1$), while increasing k generally improves performance by providing additional evidence despite introducing more noisy context. Overall, these results indicate that our representation editing can improve robustness across different retrieval conditions.

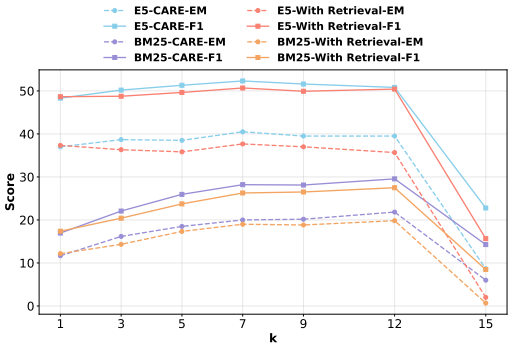


Figure 4: Performance comparison under different retrieval methods and varying numbers of retrieved documents k.

Impact of Hyperparameter Figure 5 shows the impact of different hyperparameter settings on model performance. In the left plot, increasing λ sharply decreases performance, suggesting that while moderate perturbation is beneficial, excessive intervention can degrade representation editing [43, 35, 61]. The right plot shows the effect of varying the number of layers used for representation editing. Using the top 5 most informative layers achieves the best performance, indicating that targeted intervention is sufficient to mitigate knowledge conflicts. In contrast, editing all layers degrades performance, highlighting the importance of avoiding over-intervention.

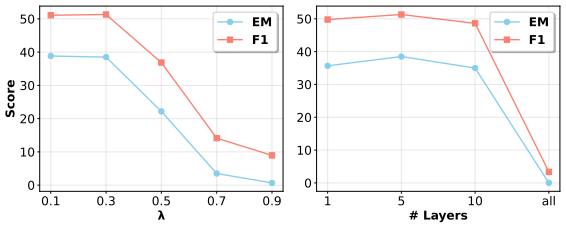


Figure 5: Performance variation with respect to λ (left) and the number of edited layers (right). λ denotes the strength of representation editing, while # Layers indicates the number of layers in which editing is applied.

Distinguishing Conflict-Associated Errors from Generic Prediction Errors To distinguish conflict-associated failures from generic incorrectness, we compare incorrect generations with and without parametric-contextual conflict. Their representations remain highly separable (AUC = 0.96), indicating that the learned representation captures information beyond correctness alone. We further learn an editing direction from non-conflict errors using the same procedure. As

Table 4: Comparison of editing directions on conflict-focused and standard QA settings.

Editing Direction	NQ-SWAP EM	NQ-SWAP F1	NOISY-NQ-SWAP EM	NOISY-NQ-SWAP F1	NQ EM	NQ F1
With Retrieval	22.00	27.28	8.50	13.93	35.83	49.64
Non-conflict Error Direction	27.00	39.86	8.70	11.24	37.16	49.94
CARE Direction	**39.83**	**44.89**	**12.17**	**17.43**	**38.50**	**51.31**

shown in Table 4, CARE substantially outperforms this direction on the conflict-focused NQ-SWAP and NOISY-NQ-SWAP benchmarks, while showing only a marginal advantage on standard NQ. This contrast suggests that CARE's effect is particularly associated with conflict-related failures rather than generic error correction.

Efficiency and Qualitative Analyses We provide efficiency and qualitative analyses in Appendix E.6 and Appendix F, respectively. These analyses show that CARE introduces only modest inference overhead while producing more reliable answers under noisy RAG settings.

6 Conclusion

In this work, we proposed CARE, an inference-time representation editing method for improving the robustness of RAG under noisy knowledge conflict settings. CARE trains an autoencoder-based module to learn a latent editing direction from intermediate representations associated with correct and incorrect generations, and applies this direction to selected LLM layers during inference. Experimental results across multiple LLMs and QA benchmarks show that CARE improves answer accuracy over retrieval-based, conflict-mitigation, and inference-time truthfulness baselines, particularly under noisy retrieval conditions. These findings suggest that representation-level intervention is a promising direction for improving reliable, knowledge-grounded response generation in RAG systems.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2021R1A6A1A03045425). This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (RS-2024-00398115, Research on the reliability and coherence of outcomes produced by Generative AI). This work was supported by the Commercialization Promotion Agency for R&D Outcomes (COMPA) grant funded by the Korea government (Ministry of Science and ICT) (2710096072). This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (IITP-2026-RS-2026-25615817, AI Star Fellowship Support Program).

References

[1] Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

[2] Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International conference on learning representations*, volume 2024, pages 9112–9141, 2024.

[3] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29, 2016.

[4] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2, 2023.

[5] Chi-Min Chan, Chunpu Xu, Ruibin Yuan, Hongyin Luo, Wei Xue, Yike Guo, and Jie Fu. RQ-RAG: Learning to refine queries for retrieval augmented generation. In *First Conference on Language Modeling*, 2024.

[6] Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. Inside: Llms' internal states retain the power of hallucination detection. 2024.

[7] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[8] Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[9] Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. Bias and unfairness in information retrieval systems: New challenges in the llm era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 6437–6447. ACM, August 2024.

[10] Yibing Du, Antoine Bosselut, and Christopher D. Manning. Synthetic disinformation attacks on automated fact verification systems. In *Proceedings of the Thirty-Sixth AAAI Conference on Artificial Intelligence*, 2022.

[11] Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*, 2025.

[12] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models, 2024.

[13] Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in neural information processing systems*, 36:44123–44279, 2023.

[14] Adi Haviv, Ido Cohen, Jacob Gidron, Roei Schuster, Yoav Goldberg, and Mor Geva. Understanding transformer memorization recall through idioms. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 248–264, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[15] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[16] Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In Donia Scott, Nuria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[17] Liu Huanshuo, Hao Zhang, Zhijiang Guo, Jing Wang, Kuicai Dong, Xiangyang Li, Yi Quan Lee, Cong Zhang, and Yong Liu. CtrlA: Adaptive retrieval-augmented generation via inherent control. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12592–12618, Vienna, Austria, July 2025. Association for Computational Linguistics.

[18] Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[19] Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43, 2023.

[20] Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7036–7050, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

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

[22] Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LLMLingua: Compressing prompts for accelerated inference of large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13358–13376, Singapore, December 2023. Association for Computational Linguistics.

[23] Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. 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 1658–1677, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[24] Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In *Forty-first International Conference on Machine Learning*, 2024.

[25] Jiajie Jin, Yutao Zhu, Zhicheng Dou, Guanting Dong, Xinyu Yang, Chenghao Zhang, Tong Zhao, Zhao Yang, and Ji-Rong Wen. Flashrag: A modular toolkit for efficient retrieval-augmented generation research. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 737–740, 2025.

[26] Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. PubMedQA: A dataset for biomedical research question answering. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China, November 2019. Association for Computational Linguistics.

[27] Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, Xiaojian Jiang, Jiexin Xu, Li Qiuxia, and Jun Zhao. Tug-of-war between knowledge: Exploring and resolving knowledge conflicts in retrieval-augmented language models. In *Proceedings of the 2024 joint international conference on computational linguistics, language resources and evaluation (LREC-COLING 2024)*, pages 16867–16878, 2024.

[28] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[29] Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report, 2025.

[30] Jaehyung Kim, Jaehyun Nam, Sangwoo Mo, Jongjin Park, Sang-Woo Lee, Minjoon Seo, Jung-Woo Ha, and Jinwoo Shin. Sure: Summarizing retrievals using answer candidates for open-domain QA of LLMs. In *The Twelfth International Conference on Learning Representations*, 2024.

[31] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[32] Lingkai Kong, Haorui Wang, Wenhao Mu, Yuanqi Du, Yuchen Zhuang, Yifei Zhou, Yue Song, Rongzhi Zhang, Kai Wang, and Chao Zhang. Aligning large language models with representation editing: A control perspective. *Advances in Neural Information Processing Systems*, 37:37356–37384, 2024.

[33] 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. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019.

[34] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, 2017.

[35] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36:41451–41530, 2023.

[36] Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. Compressing context to enhance inference efficiency of large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6342–6353, Singapore, December 2023. Association for Computational Linguistics.

[37] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. Pyserini: An easy-to-use python toolkit to support replicable ir research with sparse and dense representations, 2021.

[38] Sheng Liu, Haotian Ye, Lei Xing, and James Zou. In-context vectors: Making in context learning more effective and controllable through latent space steering. *arXiv preprint arXiv:2311.06668*, 2023.

[39] Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[40] Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A. Smith. Time waits for no one! analysis and challenges of temporal misalignment. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5944–5958, Seattle, United States, July 2022. Association for Computational Linguistics.

[41] 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 Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada, July 2023. Association for Computational Linguistics.

[42] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

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

[44] Ella Neeman, Roee Aharoni, Or Honovich, Leshem Choshen, Idan Szpektor, and Omri Abend. DisentQA: Disentangling parametric and contextual knowledge with counterfactual question answering. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10056–10070, Toronto, Canada, July 2023. Association for Computational Linguistics.

[45] OpenAI. Gpt-4 technical report, 2023.

[46] Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. *Advances in Neural Information Processing Systems*, 36:66727–66754, 2023.

[47] Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, Online, June 2021. Association for Computational Linguistics.

[48] Minhu Park, Hongseok Oh, Eunkyung Choi, and Wonseok Hwang. Lrage: Legal retrieval augmented generation evaluation tool, 2025.

[49] Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo Maria Ponti, and Shay Cohen. Spectral editing of activations for large language model alignment. *Advances in Neural Information Processing Systems*, 37:56958–56987, 2024.

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

[51] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In Jian Su, Kevin Duh, and Xavier Carreras, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[52] Stephen Robertson, Hugo Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389, 2009.

[53] Ohad Rubin and Jonathan Berant. Retrieval-pretrained transformer: Long-range language modeling with self-retrieval. *Transactions of the Association for Computational Linguistics*, 12:1197–1213, 2024.

[54] Jaehyung Seo, Hyeonseok Moon, and Heuiseok Lim. The impact of negated text on hallucination with large language models. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13565–13583, Suzhou, China, November 2025. Association for Computational Linguistics.

[55] Dan Shi, Renren Jin, Tianhao Shen, Weilong Dong, Xinwei Wu, and Deyi Xiong. IRCAN: Mitigating knowledge conflicts in LLM generation via identifying and reweighting context-aware neurons. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[56] Haizhou Shi, Zihao Xu, Hengyi Wang, Weiyi Qin, Wenyuan Wang, Yibin Wang, Zifeng Wang, Sayna Ebrahimi, and Hao Wang. Continual learning of large language models: A comprehensive survey, 2024.

[57] Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. Trusting your evidence: Hallucinate less with context-aware decoding. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 783–791, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[58] Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Nikul Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. Layer by layer: Uncovering hidden representations in language models. In *Forty-second International Conference on Machine Learning*, 2025.

[59] Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. ASQA: Factoid questions meet long-form answers. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[60] Hongjin SU, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han yu Wang, Liu Haisu, Quan Shi, Zachary S Siegel, Michael Tang, Ruoxi Sun, Jinsung Yoon, Sercan O Arik, Danqi Chen, and Tao Yu. BRIGHT: A realistic and challenging benchmark for reasoning-intensive retrieval. In *The Thirteenth International Conference on Learning Representations*, 2025.

[61] Hao Sun, Huailiang Peng, Qiong Dai, Xu Bai, and Yanan Cao. Layernavigator: Finding promising intervention layers for efficient activation steering in large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

[62] Hexiang Tan, Fei Sun, Wanli Yang, Yuanzhuo Wang, Qi Cao, and Xueqi Cheng. Blinded by generated contexts: How language models merge generated and retrieved contexts when knowledge conflicts? 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 6207–6227, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[63] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, 2018.

[64] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models, 2023.

[65] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

[66] 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. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30553–30571, Vienna, Austria, July 2025. Association for Computational Linguistics.

[67] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training, 2024.

[68] Yasi Wang, Hongxun Yao, and Sicheng Zhao. Auto-encoder based dimensionality reduction. *Neurocomputing*, 184:232–242, 2016.

[69] Yike Wang, Shangbin Feng, Heng Wang, Weijia Shi, Vidhisha Balachandran, Tianxing He, and Yulia Tsvetkov. Resolving knowledge conflicts in large language models. In *First Conference on Language Modeling*, 2024.

[70] Zhepei Wei, Wei-Lin Chen, and Yu Meng. InstructRAG: Instructing retrieval-augmented generation via self-synthesized rationales. In *The Thirteenth International Conference on Learning Representations*, 2025.

[71] Zhaofeng Wu, Xinyan Velocity Yu, Dani Yogatama, Jiasen Lu, and Yoon Kim. The semantic hub hypothesis: Language models share semantic representations across languages and modalities. In *The Thirteenth International Conference on Learning Representations*, 2025.

[72] Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations*, 2024.

[73] Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. Benchmarking retrieval-augmented generation for medicine. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6233–6251, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[74] Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for LLMs: A survey. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8541–8565, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[75] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[76] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report, 2025.

[77] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze D Gui, Ziran W Jiang, Ziyu Jiang, et al. Crag-comprehensive rag benchmark. *Advances in Neural Information Processing Systems*, 37:10470–10490, 2024.

[78] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[79] Michael Zhang and Eunsol Choi. Mitigating temporal misalignment by discarding outdated facts. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14213–14226, Singapore, December 2023. Association for Computational Linguistics.

[80] Shaolei Zhang, Tian Yu, and Yang Feng. TruthX: Alleviating hallucinations by editing large language models in truthful space. 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 8908–8949, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[81] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2025.

[82] Yu Zhao, Alessio Devoto, Giwon Hong, Xiaotang Du, Aryo Pradipta Gema, Hongru Wang, Xuanli He, Kam-Fai Wong, and Pasquale Minervini. Steering knowledge selection behaviours in LLMs via SAE-based representation engineering. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5117–5136, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.

[83] Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. Context-faithful prompting for large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14544–14556, Singapore, December 2023. Association for Computational Linguistics.

Appendix

Contents

A Additional Related Work **17**

B ROC–AUC Estimation **17**

C Experimental Details **18**

C.1 Datasets 18

C.2 Baselines 19

C.3 Implementation Details 19

C.4 Prompt Templates 21

D Additional Experimental Results **21**

D.1 Robustness Across Random Seeds 21

D.2 Scalability Across LLMs 22

D.3 Generalization to Additional Tasks and Domains 23

E Additional Analysis **24**

E.1 Effect of Negative Sample Composition 24

E.2 Impact of Training Data 24

E.3 Conflict Type Analysis 25

E.4 Analysis of `CARE` Design Choices 25

E.5 Robustness against Lexical Differences 26

E.6 Efficiency Analysis 26

E.7 Layer-wise AUC of Latent Space Separation 27

E.8 Effect of Training Dataset Size 27

E.9 Effect of Applying CAD 28

F Case Study **28**

G Limitations and Broader Impacts **28**

A Additional Related Work

Mechanistic Interpretability Recent mechanistic interpretability work has leveraged sparse dictionary learning (SDL) and Sparse Autoencoders (SAEs) to decompose internal LLM representations into interpretable, monosemantic features [4, 18, 11]. These approaches typically rely on overcomplete latent spaces and strong sparsity regularization to disentangle complex activation patterns. Building on this line of work, SPARE [82] further demonstrated that SAE-derived latent features can be directly manipulated to steer knowledge-selection behaviors in LLMs, highlighting the controllability and interpretability enabled by sparse latent structures. Although CARE also employs an autoencoder architecture to analyze representation space, its purpose differs fundamentally from that of SAEs. Instead of pursuing unsupervised feature decomposition for interpretability, we utilize a compressed, dense latent space trained with supervised contrastive signals. This enables us to manipulate global directions associated with knowledge conflict, prioritizing the robustness and controllability of model behavior.

Comparison with Related Work While our work is related to various RAG and representation-editing approaches in that it aims to mitigate knowledge conflicts caused by imperfect retrieval, it differs markedly in both goals and design.

- ASTUTE RAG [66]: Building on the observation that incorrect or even adversarial content is inevitably introduced by imperfect retrieval, ASTUTE RAG repeatedly integrates internal knowledge and external documents in a source-aware manner and selects the final answer based on reliability. In contrast, CARE is not tied to any particular RAG algorithm; instead, it directly intervenes in the intermediate representations of the LLM after retrieval has already been performed, weakening conflict signals between parametric knowledge and document-based knowledge via representation editing. Thus, ASTUTE RAG operates at the high-level design of RAG strategies, whereas CARE performs fine-grained adjustments in the representation space of a given LLM, making the two approaches complementary.
- CtrlA [17]: Although CtrlA also approaches RAG from a representation perspective, its goal is *Adaptive RAG*, namely deciding when to trigger retrieval. CtrlA extracts honesty and confidence directions from the LLM's internal representations and uses them to control whether and when retrieval should be performed, proposing an inherent control framework. In contrast, CARE applies a *Robust RAG* perspective: it assumes that retrieval has already been performed and may have injected noise, and then corrects the model's internal state after the fact.
- SPARE [82]: SPARE leverages SAEs to extract sparse features in the LLM's latent space that correspond to parametric and contextual knowledge, and then modulates these features to control which knowledge source is emphasized. SPARE and CARE both manipulate latent representations, but differ in two key aspects. (i) SPARE focuses on adjusting knowledge selection at the level of interpretable sparse features, whereas CARE learns conflict patterns directly in a compressed latent space and moves representations along directions that alleviate those conflicts. (ii) SPARE requires relatively high computational cost for training SAEs and extracting interpretable features, while CARE employs a lightweight autoencoder architecture centered on a 1D direction vector, making it easier to deploy across different models and domains.

B ROC–AUC Estimation

Centroid-based Scoring Given an arbitrary latent vector z, we define a centroid-based score:

$$f(z) = \frac{\|z - \mathbb{Z}_{\text{neg}}\|_2}{\|z - \mathbb{Z}_{\text{pos}}\|_2 + \|z - \mathbb{Z}_{\text{neg}}\|_2}. \tag{7}$$

This score takes values in $[0, 1]$, where larger values indicate that z is closer to the positive centroid.

ROC and AUC Let $\{f(z_i^+)\}_{i=1}^m$ and $\{f(z_j^-)\}_{j=1}^n$ denote the scores for positive and negative samples, respectively. For a threshold $\gamma \in \mathbb{R}$, define

$$\text{TPR}(\gamma) = \frac{1}{m}\sum_{i=1}^{m} \mathbf{1}[f(z_i^+) \geq \gamma], \qquad \text{FPR}(\gamma) = \frac{1}{n}\sum_{j=1}^{n} \mathbf{1}[f(z_j^-) \geq \gamma]. \tag{8}$$

The ROC curve is the set $\{(\mathrm{FPR}(\gamma), \mathrm{TPR}(\gamma)) : \gamma \in \mathbb{R}\}$, and AUC is

$$\mathrm{AUC} = \int_0^1 \mathrm{TPR}\, d(\mathrm{FPR}). \tag{9}$$

Empirical Estimation In practice, AUC is computed via the Mann–Whitney statistic:

$$\widehat{\mathrm{AUC}} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \Big(\mathbf{1}[f(z_i^+) > f(z_j^-)] + \tfrac{1}{2} \mathbf{1}[f(z_i^+) = f(z_j^-)] \Big). \tag{10}$$

C Experimental Details

C.1 Datasets

We use the same set of datasets as those employed in FlashRAG, all of which are publicly available via Hugging Face[12]. Detailed descriptions of the datasets used in our experiments are provided below:

- **Natural Questions (NQ)** [33]: A large-scale QA dataset consisting of real user queries collected from Google Search and associated Wikipedia documents. The questions are naturally phrased and diverse, and the answers are extracted from specific paragraphs or sentences within the documents. Because of its emphasis on long-form and context-rich questions, NQ serves as a strong benchmark for evaluating a model's reading comprehension ability.
- **TriviaQA** [28]: A QA dataset featuring quiz-style questions paired with evidence documents from the web. The questions are not synthetic but are collected from actual trivia quizzes, requiring complex reasoning and substantial background knowledge to answer correctly.
- **PopQA** [41]: A large-scale open-domain QA dataset designed to assess factual knowledge based on Wikidata. It primarily covers commonsense and factual questions across diverse domains such as entertainment, history, and sports.
- **SQuAD** [51]: A widely used QA benchmark focused on paragraph-level reading comprehension over Wikipedia articles. Answers are extracted as continuous text spans from a single paragraph. Its high-quality human-annotated questions and answers make it a standard for evaluating the base performance of QA models.
- **HotpotQA** [78]: A multi-hop QA dataset that requires reasoning over multiple supporting documents. In addition to the answer, supporting facts are provided to facilitate the evaluation of multi-step reasoning and explainability in model predictions.
- **2WikiMultiHopQA (2WIKI)** [16]: A multi-hop QA dataset that involves reasoning across multiple Wikipedia articles. Answering a question typically requires integrating information from at least two distinct documents, making it suitable for assessing complex relational and world knowledge reasoning.
- **NQ-SWAP** [39]: NQ-SWAP is a knowledge conflict benchmark derived from NQ, where gold entities in the retrieved documents are intentionally swapped to create contradictions with the model's parametric knowledge.

We randomly sample 5,000 instances from the original training dataset for evaluation. The evaluation aims to identify negative samples by selecting instances with an F1 score of 0. When the NQ training dataset is used, evaluation with the LLaMA-3.1-8B-Instruct model yields 766 instances. We provide an analysis of performance under different training dataset sizes in Appendix E.8.

License We are committed to ethical research practices and ensuring the reproducibility of our work. To this end, we present the licensing information for all datasets utilized in our experiments. These include Natural Questions (CC BY-SA 3.0), TriviaQA (Apache License 2.0), PopQA (MIT License), SQuAD (CC BY-SA 4.0), HotpotQA (CC BY-SA 4.0), 2WikiMultiHopQA (Apache License 2.0), and NQ-SWAP (MIT License). Our use of these datasets is in full compliance with their terms, as all aforementioned licenses expressly permit their application in academic research.

[12] `https://huggingface.co/datasets/RUC-NLPIR/FlashRAG_datasets`

Table 5: Performance of two CAD variants across six QA datasets with respect to varying values of the weight parameter α.

Method	α	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
		EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	
[27]	1	**36.67**	**49.60**	**57.67**	**68.47**	**38.83**	**48.22**	**28.50**	40.41	**28.33**	**39.13**	16.83	26.99	**39.97**
	0.5	34.33	48.59	57.50	68.34	38.67	48.05	28.33	**40.69**	27.50	38.69	**18.00**	**27.48**	39.68
[57]	1	36.16	47.62	55.66	65.72	36.33	44.87	25.33	37.02	21.00	32.42	7.83	20.36	35.86
	0.5	35.66	47.12	56.33	66.49	36.83	45.16	25.66	36.93	21.50	32.55	8.33	20.43	36.08

C.2 Baselines

We provide a detailed description of each adopted baseline method below:

- **No Retrieval**: A baseline setting where the model generates responses solely based on its parametric knowledge, without using any external documents.
- **With Retrieval**: A standard RAG approach where the retrieved documents R are directly appended to the prompt.
- **Self-RAG** [2]: A self-reflective RAG pipeline in which the model rewrites the query, determines whether retrieval is necessary, and refines the answer via critic, grounding, and refinement over retrieved evidence. The implementation uses 'selfrag/selfrag_llama2_7b.'
- **LLMLingua** [22]: A prompt compression method that minimizes semantic loss during compression, thereby reducing the possibility of knowledge conflicts caused by irrelevant information and promoting consistent response generation. Following the paper, we use 'meta-llama/Llama-2-7b-hf' as the model for refinement.
- **Selective-Context** [36]: A context filtering method that removes parts of the LLM input with low self-information, reducing context length and improving processing efficiency without degrading performance. We use 'openai-community/gpt2' as the model for refinement.
- **SURE** [30]: Summarizes retrieved documents and compares them with candidate answers to select the response that best aligns with the given context, thereby resolving conflicting knowledge.
- **ITI** [35]: A truthfulness-oriented inference-time intervention method that identifies attention heads associated with truthfulness via linear probes and shifts their activations along truthful directions during generation; in our setup, we use 'likenneth/honest_llama2_chat_7B.'
- **CtrlA** [17]: An adaptive RAG framework that leverages honesty and confidence directions in a model's internal representations to decide whether and when to trigger retrieval.
- **TruthX** [80]: An inference-time representation editing method designed to reduce hallucinations, which uses an autoencoder to map LLM representations into a truthful latent space and edits them along a truthful direction. Our implementation uses 'ICTNLP/Llama-2-7b-chat-TruthX.'
- **Opin** [83]: A prompting strategy that reformulates a question into one asking for the speaker's opinion, encouraging the model to prioritize contextual interpretation.
- **CAD** [57, 27]: A contrastive decoding approach that adjusts the difference in token probabilities between generation with and without the given context, guiding the model to produce content grounded in the context rather than relying on prior knowledge, thus mitigating knowledge conflicts.
- **IRCAN** [55]: Identifies neurons that are highly sensitive to contextual information, especially when it conflicts with pre-trained knowledge, via the integrated gradients technique, and reweights the selected neurons to amplify their influence during generation. The paper emphasizes that combining CAD mitigates such conflicts; accordingly, in Appendix E.9, we apply CAD to IRCAN and `CARE` to verify its effect.
- **SPARE** [82]: Steers LLM behavior by manipulating sparse SAE features that disentangle parametric and contextual knowledge, and in our setup, the SAE is instantiated with the publicly released pretrained checkpoint 'yuzhaouoe/Llama2-7b-SAE'.

C.3 Implementation Details

To ensure reproducibility, we fix the random seed to 2024 for all experiments. The layer configurations used in our main experiments are as follows: for LLaMA-2-7B-Chat, we use layers 22, 23, 24, 25, and

Table 6: Prompt template used for No Retrieval. The model is instructed to answer the question solely based on its internal knowledge without access to any retrieved context.

[SYSTEM PROMPT]
Answer the question based on your own knowledge. Only give me the answer and do not output any other words.

[USER PROMPT]
Question: `{question}`
Answer:

Table 7: Prompt template used for With Retrieval, LLMLingua, Selective-Context, CAD, IRCAN, and `CARE`. The model is instructed to answer the question based on the retrieved passages, which are explicitly included in the prompt.

[SYSTEM PROMPT]
Answer the question based on the given passage. Only give me the answer and do not output any other words. The following are given passages: `{retrieval passages}`

[RETRIEVAL PASSAGES PROMPT]
Doc 1 (Title: `{title}`) `{content}`
Doc 2 (Title: `{title}`) `{content}`
Doc 3 (Title: `{title}`) `{content}`
Doc 4 (Title: `{title}`) `{content}`
Doc 5 (Title: `{title}`) `{content}`

[USER PROMPT]
Question: `{question}`
Answer:

26; for LLaMA-3.1-8B-Instruct, we utilize layers 3, 12, 18, 20, and 31; for LLaMA-2-13B-Chat, we select layers 5, 12, 24, 32, and 37; and for Qwen-2.5-7B-Instruct, we use layers 19, 21, 23, 24, and 27. The selection criterion (i.e., the plot of per-layer AUC scores) is provided in Appendix E.7. Training was conducted with LLaMA-3.1-8B-Instruct on an NVIDIA RTX A6000 48GB GPU, requiring less than 25 minutes per layer. Inference is conducted on a single GPU with a maximum input length of 2048 tokens and a maximum output length of 32 tokens[13].

For the LLMLingua setting, we use LLaMA-2-7B [64][14] to compute perplexity and apply LongLLMLingua [23] as the compressor with a compression ratio of 0.55. Selective-Context uses GPT-2 [50][15] to measure perplexity, with a compression ratio of 0.5.

There are two primary variants of CAD: the original formulation [57] and a more recent version specifically designed for noisy contexts [27]. As shown in Table 5, the noise-aware CAD consistently demonstrates superior robustness. We adopt the noise-aware version as our primary baseline for all subsequent evaluations. The CAD follows the objective function:

$$\log p_\theta(a_i \mid a_{<i}, q, R) - \alpha \log p_\theta(a_i \mid a_{<i}, q), \tag{11}$$

where $p_\theta(a_i \mid a_{<i}, q, R)$ denotes the probability of generating token a_i conditioned on the retrieved documents $R = M_D(q)$, and $p_\theta(a_i \mid a_{<i}, q)$ is the probability based solely on the internal language model without external context. The hyperparameter α controls the extent to which the influence of the internal model is penalized. Table 5 presents the performance differences across values of α. The best average performance is achieved when $\alpha = 1$, which we adopt as the default configuration in this work.

We train and evaluate IRCAN using the official implementation on github[16]. Because of out-of-memory issues, training is conducted on 8 NVIDIA SXM4 A100 80GB GPUs. The model is trained on the same dataset as used in our method for fair comparison.

[13]LLaMA-2-13B-Chat tends to generate overly long responses; therefore, we set the maximum output length to 64 tokens.
[14]meta-llama/Llama-2-7b-hf
[15]openai-community/gpt2
[16]`https://github.com/danshi777/IRCAN`

Table 8: Prompt template used for Opin. This template encourages the model to answer the question from the perspective of a fictional character "Bob," using retrieved passages attributed to Bob as supporting context.

[SYSTEM PROMPT]
Answer the question based on the given passage. Only give me the answer and do not output any other words. Bob said, `{retrieval passages}`

[RETRIEVAL PASSAGES PROMPT]
Doc 1 (Title: `{title}`) `{content}`
Doc 2 (Title: `{title}`) `{content}`
Doc 3 (Title: `{title}`) `{content}`
Doc 4 (Title: `{title}`) `{content}`
Doc 5 (Title: `{title}`) `{content}`

[USER PROMPT]
Question: `{question}` in Bob's opinion?
Answer:

Table 9: Performance with different random seeds (mean ± standard deviation).

	NQ		TriviaQA		PopQA	
Method	EM	F1	EM	F1	EM	F1
No Retrieval	24.28 ± 1.70	35.27 ± 1.60	55.22 ± 2.19	62.74 ± 1.83	21.83 ± 1.66	28.13 ± 0.77
With Retrieval	36.94 ± 1.04	49.98 ± 0.73	61.78 ± 2.38	70.42 ± 1.74	40.50 ± 1.06	49.90 ± 1.10
SURE	37.39 ± 0.87	48.71 ± 1.77	56.06 ± 5.72	64.85 ± 1.87	**43.22** ± 1.22	48.44 ± 0.33
IRCAN	36.50 ± 2.37	48.53 ± 1.94	51.56 ± 1.03	62.27 ± 1.38	37.56 ± 1.36	45.83 ± 0.69
CARE	**40.55** ± 1.77	**52.47** ± 1.35	**62.50** ± 2.09	**70.92** ± 1.34	**43.22** ± 2.11	**50.60** ± 1.26

	SQuAD		HotpotQA		2WIKI	
Method	EM	F1	EM	F1	EM	F1
No Retrieval	12.39 ± 0.67	22.59 ± 1.06	17.83 ± 0.85	26.88 ± 0.91	16.28 ± 0.52	25.73 ± 1.53
With Retrieval	31.44 ± 1.56	43.37 ± 2.29	28.17 ± 0.94	38.42 ± 0.80	14.33 ± 0.41	24.90 ± 1.15
SURE	27.00 ± 2.00	38.45 ± 1.52	21.56 ± 1.70	30.22 ± 2.25	6.94 ± 1.47	10.59 ± 3.47
IRCAN	21.56 ± 0.34	36.84 ± 0.96	23.72 ± 0.78	33.55 ± 1.17	9.78 ± 0.48	20.55 ± 0.59
CARE	**33.67** ± 2.46	**44.71** ± 2.13	**29.05** ± 0.63	**39.36** ± 0.72	**17.72** ± 0.64	**26.86** ± 1.09

C.4 Prompt Templates

This section presents the full prompt templates used throughout our experiments. We primarily adopt the prompt format proposed in FlashRAG [25]. Below, we detail the prompts used for each method:

- **No Retrieval**: See Table 6
- **With Retrieval**, **LLMLingua**, **Selective-Context**, **ITI**, **TruthX**, **CAD**, **IRCAN**, **SPARE**, and `CARE`: See Table 7
- **Opin**: See Table 8
- **Self-RAG**, **SURE**, and **CtrlA**: We use the prompts provided in the original paper.

We observe that the LLaMA-2-13B-Chat [64] model tends to generate additional explanations beyond the desired answer. To mitigate this behavior, we append the instruction "Only give me the answer and do not output any other words." at the end of the user prompt for this model.

D Additional Experimental Results

D.1 Robustness Across Random Seeds

To assess the robustness of our method with respect to random seeds, we repeat all experiments with three different random seeds (7, 42, 2024) and report the mean ± standard deviation over these runs in Table 9. Across datasets and metrics, the variance is relatively small, indicating that the behavior of all methods is reasonably stable under different seeds. In terms of overall performance,

Table 10: Performance on Qwen2.5-32B-Instruct and Gemma-3-12B-IT.

Method	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	
Qwen2.5-32B-Instruct													
No Retrieval	35.69	25.67	64.71	58.33	27.54	23.50	24.54	14.67	34.05	25.67	33.15	27.17	32.89
With Retrieval	50.29	38.00	71.58	62.67	50.72	43.00	43.12	32.17	45.92	34.00	36.12	27.83	44.62
SURE	52.78	40.50	67.83	61.50	49.00	43.83	39.38	31.83	38.93	29.33	25.63	20.17	41.73
CAD	50.45	37.67	72.24	63.17	50.66	42.50	42.51	31.00	47.09	34.67	36.76	28.00	44.73
CARE	**54.18**	**43.50**	**72.87**	**65.00**	**51.04**	**44.00**	**45.77**	**36.50**	**47.11**	**35.67**	**37.49**	**30.67**	**46.98**
Gemma-3-12B-IT													
No Retrieval	27.15	37.99	53.17	63.03	21.67	25.25	10.83	17.96	21.67	28.07	25.83	30.14	30.23
With Retrieval	40.50	54.25	57.56	66.12	37.51	44.93	20.45	30.36	25.66	31.21	24.17	29.27	38.50
SURE	34.61	55.57	52.55	60.61	43.33	48.33	17.55	27.46	14.17	21.27	17.00	17.52	34.16
CAD	41.08	55.38	63.84	70.49	45.70	50.65	21.08	30.64	30.38	38.31	23.33	28.10	41.58
CARE	**44.59**	**56.22**	**64.17**	**72.89**	**49.70**	**53.72**	**24.03**	**34.49**	**30.48**	**39.14**	**33.59**	**33.63**	**44.72**

Table 11: Performance on Qwen3-4B-Thinking, a Large Reasoning Model.

Method	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	
No Retrieval	15.95	13.33	7.17	21.67	4.67	20.00	14.32	11.67	11.04	20.00	6.57	26.67	14.42
With Retrieval	18.26	40.00	25.50	46.67	18.44	38.33	21.56	**21.67**	**20.60**	28.33	8.21	26.67	26.19
CARE	**23.60**	**50.00**	**40.07**	**56.67**	**29.38**	**56.67**	**22.69**	**21.67**	20.26	**30.00**	**14.48**	**28.33**	**32.82**

CARE consistently outperforms With Retrieval on average. These results demonstrate that the gains brought by CARE are not tied to a particular random seed, but instead constitute a reliably reproducible improvement over existing retrieval-augmented QA baselines.

D.2 Scalability Across LLMs

Effectiveness on 32B-Scale Models To verify whether the efficacy of CARE scales to larger language models, we conduct additional experiments using Qwen2.5-32B-Instruct [76][17]. Table 10 presents the evaluation results across six open-domain QA benchmarks. Due to computational constraints, baselines requiring additional training were excluded; instead, we focus our comparison on training-free conflict resolution methods, namely SURE and CAD. The results demonstrate that CARE consistently improves upon the With Retrieval, confirming its effectiveness even at the 32B parameter scale. Furthermore, while baselines often exhibit instability, leading to performance degradation compared to standard retrieval, CARE maintains robust, consistent performance gains across all evaluated benchmarks.

Effectiveness Across Model Families To examine whether CARE generalizes beyond the Qwen and LLaMA model families, we additionally evaluate it on Gemma-3-12B-IT [29][18]. As shown in Table 10, CARE improves the average score from 38.50 with standard retrieval to 44.72, while also outperforming the training-free baselines SURE and CAD. Moreover, CARE consistently improves both EM and F1 across all six benchmarks compared with With Retrieval. These results suggest that the effectiveness of CARE is not limited to a specific model family, but transfers to a different pretrained model family and scale.

Effectiveness on Large Reasoning Models To assess whether CARE also benefits models explicitly designed for multi-step reasoning, we further evaluate it on Qwen3-4B-Thinking [75][19], a representative Large Reasoning Model (LRM). Table 11 reports results on six open-domain QA benchmarks. Adding CARE raises the average score to 32.82. These results indicate that CARE is complementary to

[17] Qwen/Qwen2.5-32B-Instruct
[18] google/gemma-3-12b-it
[19] Qwen/Qwen3-4B-Thinking-2507

Table 12: Performance on CRAG across different backbone models.

Method	LLaMA-2-13B-Chat			LLaMA-3.1-8B-Instruct			Qwen-2.5-7B-Instruct		
	Acc.	Hall.	Miss.	Acc.	Hall.	Miss.	Acc.	Hall.	Miss.
No Retrieval	0.1390	0.8610	0.0000	0.1508	0.8489	0.0004	0.1075	0.8917	0.0007
With Retrieval	0.1574	0.7291	0.1135	0.1138	0.3370	0.5491	0.1042	0.3854	0.5103
LLMLingua	0.1522	0.8041	0.0436	0.0939	0.3004	0.6057	0.0976	0.3489	0.5536
Selective-Context	0.1341	0.7635	0.1024	0.0787	0.2890	0.6323	0.0735	0.3574	0.5691
SURE	0.0695	0.4819	0.4486	0.1208	0.6859	0.1933	0.1345	0.8296	0.0358
Opin	0.1611	0.8389	0.0000	0.1548	0.8444	0.0007	0.1559	0.8433	0.0007
CAD	0.0824	0.3728	0.5447	0.0946	0.3511	0.5543	0.0916	0.3732	0.5351
IRCAN	0.1086	0.8218	0.0696	0.1556	0.8441	0.0004	0.0898	0.8374	0.0728
CARE	**0.1782**	0.6915	0.1303	**0.1574**	0.3392	0.5034	**0.1626**	0.3766	0.4608

Table 13: Generalization of CARE across additional QA settings, non-QA tasks, and specialized domains, including multi-hop QA (MuSiQue), long-form QA (ASQA), relation extraction (ZSRE), fact-checking (FEVER), biomedical QA (PubMedQA), and legal reasoning (LegalBench).

Method	MuSiQue	ASQA	ZSRE	FEVER	PubMedQA	LegalBench
No Retrieval	3.33	8.59	10.80	73.33	46.67	**72.26**
With Retrieval	5.33	28.71	37.84	90.00	55.00	71.07
CARE	**6.50**	**31.75**	**45.97**	**92.00**	**55.56**	71.97

the built-in reasoning capabilities of LRMs, enabling them to use retrieved evidence more reliably rather than naively trusting it.

D.3 Generalization to Additional Tasks and Domains

Additional QA Settings We further evaluate CARE on CRAG [77], MuSiQue [65], and ASQA [59] to examine its generalization to more challenging RAG settings beyond the QA benchmarks used in the main experiments. CRAG reflects a realistic RAG setting where retrieved evidence can be noisy or insufficient. As shown in Table 12, CARE achieves the highest accuracy across all three evaluated backbones. We also evaluate on MuSiQue, a multi-hop QA benchmark designed to reduce shortcut-based reasoning. As shown in Table 13, CARE improves performance over standard retrieval from 5.33 to 6.50. Finally, we evaluate on ASQA, which requires synthesizing information from multiple retrieved passages into long-form answers. Table 13 shows that CARE improves ROUGE-L from 28.71 with standard retrieval to 31.75. These results suggest that the learned representation-editing direction remains effective across noisy retrieval, multi-hop reasoning, and long-form generation settings.

Non-QA Tasks We examine whether CARE, trained only on QA data, can transfer to unseen RAG-style tasks beyond the QA benchmarks used in the main experiments. To this end, we evaluate CARE on ZSRE [34] for relation extraction and FEVER [63] for fact-checking, using EM and ACC as evaluation metrics, respectively. As shown in Table 13, CARE outperforms the standard With Retrieval baseline on both benchmarks. These results suggest that the learned representation-editing direction is not limited to the specific answer format of the QA training data, and can provide benefits in other RAG-style tasks where retrieved information may interact with the model's parametric knowledge.

Specialized Domain To examine whether the conflict-aware directions learned from a single QA corpus (NQ) extend to domains requiring specialized expertise, we additionally evaluate CARE on PubMedQA [26] and LegalBench [13] without any domain-specific fine-tuning. PubMedQA is a biomedical research QA dataset whose possible answers are yes/no/maybe, reflecting the veracity of question statements grounded in scientific literature. We conduct evaluation under the MedRAG [73] framework, retrieving from the MedCorp corpus and reporting the accuracy of model predictions. LegalBench comprises a variety of legal reasoning tasks. It is evaluated within the LRAGE [48] framework, which performs retrieval over Pile-of-Law, a legal source provided therein. In these experiments, we utilize BM25 and LLaMA-3.1-8B-Instruct to retrieve the top 5 documents.

Table 14: Performance versus changes in the negative-sample distribution. Balanced denotes an equal mixture of parametric and noise samples, while Parametric-only and Noise-only indicate distributions consisting exclusively of parametric or noise negatives, respectively.

Setting	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	
Balanced (50% / 50%)	**38.50**	**51.31**	**59.67**	69.96	**40.33**	**48.84**	**30.33**	**41.85**	**29.50**	40.36	**18.33**	**27.51**	**41.37**
Parametric-only (100% / 0%)	35.83	50.14	58.33	70.00	37.83	47.72	29.00	41.46	28.17	40.63	17.83	27.48	40.37
Noise-only (0% / 100%)	36.00	50.09	58.67	**70.48**	37.50	47.69	29.00	41.48	28.67	**40.86**	**18.33**	27.68	40.54

Table 15: Effect of training data used to learn the editing direction. Target-trained denotes a variant where the editing direction is learned from the training split of each target dataset.

Method	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	
CARE	38.50	51.31	59.67	69.96	40.33	48.84	30.33	41.85	29.50	40.36	18.33	27.51	41.37
Target-trained	38.50	51.31	60.67	69.43	43.17	50.00	31.67	42.22	29.83	39.93	18.33	27.34	41.87

As shown in Table 13, CARE yields a modest improvement on PubMedQA, increasing accuracy from 55.00 with simple retrieval to 55.56. This suggests that the learned editing direction does not substantially harm performance even in a specialized biomedical domain, although the gain is limited. On LegalBench, retrieval lowers the base model performance from 72.26 to 71.07, indicating that retrieved legal passages can introduce noise or domain-specific mismatch that interferes with generation. In this setting, CARE recovers much of the lost accuracy while using the same retriever. These results suggest that inference-time representation editing can partially mitigate the harmful effects of noisy or mismatched retrieved evidence in specialized domains.

E Additional Analysis

E.1 Effect of Negative Sample Composition

We ablate the composition of negative answers by varying the proportion of Parametric vs. Noise negatives while holding all other settings fixed. Concretely, we compare three configurations: a balanced mix ($50\%/50\%$), parametric-only ($100\%/0\%$), and noise-only ($0\%/100\%$). As summarized in Table 14, the balanced configuration achieves the best average performance, outperforming both parametric-only and noise-only. At the dataset level, a similar trend holds. Meanwhile, the noise-only setting attains the best F1 on TriviaQA and HotpotQA, and a slightly higher F1 on 2WIKI. These results suggest that the two negative types provide complementary supervision signals: parametric negatives better regularize entity-centric and short-form QA, whereas noise negatives sharpen robustness to retrieval noise and multi-hop reasoning. If CARE were trained only along a single dimension of truthfulness, such a complementary effect would not arise. This supports our claim that it is designed to handle two forms of conflict. Overall, the balanced mixture confers the most consistent gains across benchmarks, indicating that aligning the representation against both parametric confounders and retrieval-induced noise is crucial for generalization in RAG.

E.2 Impact of Training Data

We further examine how the training data used to learn the editing direction affects CARE. In the main experiments, CARE learns the editing direction from NQ and applies the same direction to all evaluation datasets. To assess whether target-specific supervision provides additional benefits, we compare this default setting with a target-trained variant, where the editing direction is learned from the training split of each target dataset. Table 15 shows that the target-trained variant yields only modest improvements over the default NQ-trained CARE direction. The average score increases from 41.37 to 41.87, with gains mainly observed on PopQA and SQuAD. However, the improvements are not uniform across all metrics: TriviaQA, HotpotQA, and 2WIKI show slightly lower F1 scores despite comparable or improved EM scores. These results suggest that the NQ-trained direction captures transferable conflict-mitigation signals, while target-specific training data can provide small additional gains when available.

Table 17: Ablation study on CARE design choices. **Latent Dimension** $d/2$ denotes a setting where the autoencoder latent dimension is expanded to $d/2$ instead of the default, **Without LeakyReLU** denotes a purely linear autoencoder without any nonlinear activations, and **Upper 5 Layers** denotes a setting where interventions are applied only to the top 5 layers instead of the original layer configuration.

Method	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	
CARE	38.50	51.31	59.67	69.96	40.33	48.84	30.33	41.85	29.50	40.36	18.33	27.51	41.37
Latent Dimension $d/2$	33.16	47.64	51.50	63.39	34.00	44.21	23.83	34.70	20.67	33.01	11.00	18.95	34.67
Without LeakyReLU	8.33	20.55	25.00	40.07	11.66	27.65	8.33	18.39	16.67	22.32	10.00	14.48	18.62
Upper 5 Layers	39.50	51.32	59.83	69.67	41.16	48.95	30.66	41.79	30.33	40.10	17.83	27.22	41.53

E.3 Conflict Type Analysis

We further analyze whether CARE mitigates different types of conflict-related failures. For this analysis, we focus on samples where the retrieved documents contain the correct answer, but the model still generates an incorrect response. We categorize these failures into two types. A failure is categorized as *Parametric* if the generated answer matches the answer produced without retrieved context, suggesting that the model falls back on its parametric knowledge. Otherwise, we categorize it as *Noise*, indicating that the error is likely related to noisy or improperly integrated retrieved context. Lower values indicate fewer errors. Table 16 reports the average error rates across six datasets. SURE can reduce noise errors on some datasets, such as PopQA and 2WIKI, likely due to its document summarization and filtering mechanism. However, this comes with a substantial increase in parametric failures on average, suggesting that filtering retrieved evidence may also weaken useful context and cause the model to fall back to parametric knowledge. In contrast, CARE reduces both error types simultaneously. Compared with With Retrieval, CARE lowers the average Parametric error from 10.28 to 8.91 and the average Noise error from 37.58 to 33.76. This suggests that CARE does not simply suppress one source of error at the cost of another, but provides a more balanced conflict-mitigation effect.

Table 16: Conflict type analysis averaged across six datasets. Lower values indicate fewer errors.

Method	Parametric ↓	Noise ↓
With Retrieval	10.28	37.58
SURE	17.61	38.04
CARE	**8.91**	**33.76**

E.4 Analysis of CARE Design Choices

In this section, we investigate how the key design choices of CARE affect its performance. Table 17 presents ablation results on (i) the autoencoder latent dimension, (ii) the use of nonlinearity in the autoencoder, and (iii) the choice of layers where the intervention is applied.

The autoencoder in CARE requires an appropriately compressed latent space. We use one quarter of the hidden size ($d/4$) as the latent dimension and compare it to a setting where the latent dimension is increased to $d/2$. The results show that this larger-latent setting consistently degrades overall performance. A plausible explanation is that an overly large latent space preserves complex signals and noise that are not directly related to knowledge conflict, thereby introducing interference during conflict mitigation. By contrast, the stronger compression at $d/4$ effectively filters out such unnecessary factors and retains only the information needed to distinguish conflict states.

The encoder/decoder MLPs of CARE use the LeakyReLU activation function. When we remove all nonlinearities and replace the autoencoder with a purely linear one, the average performance drops sharply from 41.37 to 18.62. This suggests that simple linear projection is insufficient to capture the latent structure required to distinguish between knowledge conflict states. Thus, nonlinearity plays a crucial role in isolating and correcting conflict signals.

For layer selection, we first compute an AUC score for each layer and choose the layer with the highest score for intervention. If multiple layers share the highest score, we randomly select one among them. To validate the robustness of this choice, we compare our original layer selection with a variant that intervenes on the top 5 layers ranked by AUC (layers 18, 20, 28, 29, 31). The difference in average performance between the two settings is only 0.16, and the overall trends are very similar. This suggests that what matters is not the highly specialized functional role of a particular layer,

Table 18: Comparison of average retrieval, inference, and total latency (in seconds) across different methods.

Method	Retrieval Time	Inference Time	Total Time
No Retrieval	-	0.3585	0.3585
With Retrieval	1.4054	0.4677	1.8731
LLMLingua	2.1858	0.4266	2.6124
Selective-Context	1.6948	0.4430	2.1378
SURE	1.4054	6.0109	7.4163
Opin	1.4054	0.5264	1.9318
CAD	1.4054	1.8400	3.2454
IRCAN	1.3755	1.0695	2.4450
CARE	1.4054	0.4232	1.8286

but rather that each chosen layer provides a strong signal for distinguishing between conflict and non-conflict states.

E.5 Robustness against Lexical Differences

To examine whether the learned representation contains conflict-sensitive information beyond lexical differences in the generated responses, we conduct an additional controlled analysis using sample pairs with identical output tokens. Specifically, we construct 600 sample pairs extracted from NQ-SWAP, where the LLM's final generated output tokens are strictly identical. These pairs are divided into two configurations. In the first half, the output tokens are identical, but the retrieved contexts differ. For instance, in both conditions, the model outputs the same parametric answer (e.g., "Paris"). However, in the Positive condition, the retrieved document supports this answer, whereas in the Negative condition, the retrieved document provides contradictory evidence. In the second half, the output tokens are identical, but the queries differ. Here, the Positive condition represents a correct answer derived from accurately retrieved documents, while the Negative condition represents an erroneous answer induced by noisy retrieved documents. By fixing the output string across these scenarios, we completely eliminate any lexical or semantic differences that could act as confounding variables for the autoencoder. As shown in the ROC curve in Figure 6, the two conditions remain highly separable even when the generated output tokens are strictly controlled, achieving an AUC of 0.93. This result provides evidence that the observed latent separation cannot be explained solely by lexical or output-level differences and that the learned representations contain information sensitive to the alignment between generated answers and retrieved evidence.

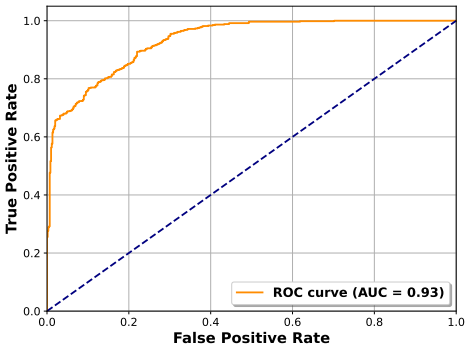


Figure 6: ROC curve of the autoencoder for sample pairs with identical output tokens.

E.6 Efficiency Analysis

Table 18 reports average runtime, decomposed into retrieval and inference latency. CARE achieves a total inference time of 1.83s, nearly identical to the baseline With Retrieval (1.87s), showing that the coordination mechanism introduces negligible latency. This efficiency arises because CARE does not spend additional time generating evasive or uncertain responses, but instead produces clear and direct answers to the query. By contrast, compression-based methods (LLMLingua, Selective-Context) introduce pre-processing latency, whereas CAD and IRCAN incur higher inference costs due to the complexity of post-retrieval reasoning. SURE shows the largest slowdown, with an average total time exceeding 7s. Overall, CARE achieves high accuracy while maintaining efficiency comparable to standard RAG, striking a favorable balance between reliability and deployment cost.

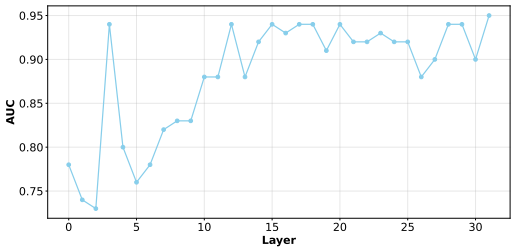


Figure 7: Layer-wise AUC scores for latent space separation. Higher AUC indicates better discriminative capability of the representations at each transformer layer.

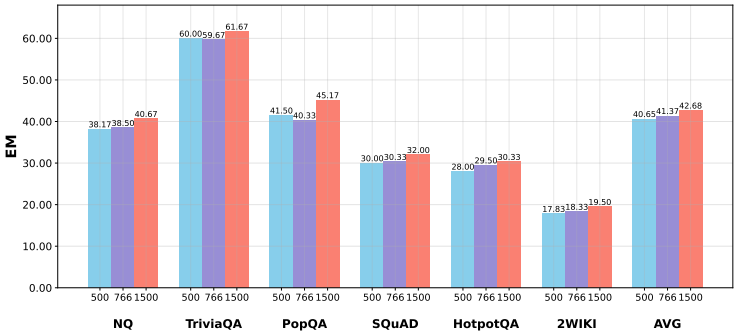


Figure 8: Performance comparison across different benchmarks according to the size of the training dataset.

E.7 Layer-wise AUC of Latent Space Separation

To analyze how well the latent representations are separated at each layer, we compute AUC for latent space separation across transformer layers. Figure 7 visualizes the AUC values measured at each layer. In the early layers, the AUC scores are relatively low and unstable, indicating limited class separation. However, up to approximately the 12th layer, AUC scores steadily increase, suggesting improved separability of representations. After the 12th layer, AUC values generally remain above 0.9, reflecting stable and high separability. This suggests that representations in the middle-to-late layers of the model contain richer information necessary for separating classes. These layers are known to encode high-level semantic features and factuality-related signals [7, 71, 58], and are believed to play a central role in integrating information under conflicting conditions. Our findings are consistent with prior empirical studies [6, 54], which report that layers most important for hallucination detection tend to be concentrated in the middle-to-late parts of the network.

E.8 Effect of Training Dataset Size

We analyze the sensitivity of CARE to the amount of training data by varying the number of training instances. Figure 8 shows the performance of CARE across multiple benchmarks when trained with 500, 766, and 1,500 instances.

Compared to our default setting of 766 training instances, reducing the data to 500 instances results in only a small performance drop. This suggests that CARE can learn a useful representation-editing direction from a relatively small number of training examples. Consequently, CARE can achieve significant conflict mitigation even in resource-constrained environments where data collection is costly. Conversely, increasing the training data to 1,500 instances yields an average improvement of 1.31 points across benchmarks. The largest gain is observed on PopQA, which contains many long-tail entity questions. One possible explanation is that additional training examples expose the autoencoder to a broader range of entity-level and context-dependent failure patterns, leading to a more effective editing direction. Overall, these results suggest that CARE is not overly sensitive to moderate reductions in training data and can further benefit from additional training examples.

Table 19: Performance with CAD additionally applied to IRCAN and CARE.

Method	NQ		TriviaQA		PopQA		SQuAD		HotpotQA		2WIKI		AVG
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	
IRCAN	37.83	49.55	50.67	61.56	38.17	45.82	21.17	35.67	23.17	33.65	9.17	19.71	35.51
IRCAN-CAD	41.17	51.36	57.17	65.35	41.00	46.99	26.33	36.59	24.50	32.59	13.00	19.77	37.99
CARE	38.50	51.31	59.67	**69.96**	40.33	48.84	30.33	41.85	29.50	**40.36**	18.33	**27.51**	41.37
CARE-CAD	**43.00**	**53.39**	**61.50**	69.88	**44.50**	**50.81**	**31.83**	**41.88**	**29.83**	39.29	**18.67**	27.14	**42.64**

E.9 Effect of Applying CAD

We observe in the original IRCAN paper that combining CAD markedly mitigates conflicts. Accordingly, as shown in Table 19, we empirically evaluate the performance of attaching CAD to both IRCAN and CARE. As a result, integrating CAD at the decoding stage yields a significant improvement in overall accuracy. Compared to CARE, CARE-CAD achieves an average absolute gain of 1.27, surpassing the prior methods IRCAN and IRCAN-CAD by 7.13 and 4.65, respectively. Across datasets, performance generally improves on all benchmarks, confirming the beneficial effect of introducing CAD. However, adding CAD entails a trade-off of increased inference time.

F Case Study

To better understand model behavior under challenging queries, Table 20 presents two representative cases illustrating how different methods respond in the presence of internal and external knowledge conflicts. In the first example, confusion arises between internal knowledge (Rab C. Nesbitt) and retrieved documents that frequently mention the boat name "The Vital Spark." Some models return hallucinated or misleading answers such as The Vital Spark or Rab C. Nesbitt. In contrast, CARE correctly identifies Para Handy by integrating two key pieces of textual evidence: "The Tales of Para Handy, starring Gregor Fisher" and "captain of the puffer 'Vital Spark'," effectively resolving the ambiguity. The second example focuses on confusion induced by external knowledge. While some models misinterpret the acronym "COBRA" and respond with Room or Civil, CARE successfully identifies Cabinet by reasoning over the retrieved phrase "Cabinet Office Briefing Rooms", which explicitly defines the acronym. These cases demonstrate that CARE is robust against knowledge conflicts and can reliably extract answers grounded in retrieved evidence, even under noisy or misleading conditions.

G Limitations and Broader Impacts

While CARE improves robustness in retrieval-augmented QA by mitigating conflicts between parametric and retrieved knowledge, it still has the following limitations.

- CARE is primarily trained on datasets where retrieval-induced confusion in long contexts constitutes a major source of error. In addition, because we use only responses with $F1 = 0$ to construct high-precision negative samples, CARE may not fully capture subtle blending errors in which correct and incorrect information are partially mixed. Consequently, its gains may be more limited in settings dominated by straightforward factual errors or such mixed errors. Future work could explore adaptive negative-sample construction tailored to different domains and error types.
- CARE applies a single, pre-learned direction vector uniformly to all inputs at inference time. While this global intervention generally provides stable performance improvements, it can introduce slight distortions when the model's parametric knowledge is already correct. A promising extension is to make the intervention strength adaptive to the estimated level of knowledge conflict.
- The selection of intervention layers and intervention strength currently relies on empirical procedures such as AUC-based layer scoring and grid search. CARE also requires access to intermediate hidden states, making it difficult to apply directly to API-only models that do not expose internal representations. Developing more automated and broadly applicable intervention mechanisms is an important direction for future work.
- Our experiments primarily focus on retrieval-augmented QA, with limited validation in broader generation settings. Moreover, CARE is not explicitly designed to assess source reliability or arbitrate

Table 20: Case study on two challenging examples involving knowledge conflict.

Question 1	**Played on TV by Gregor Fisher, whose boat was called The Vital Spark?**
Document 1	(Title: The Vital Spark) The Vital Spark was a BBC Scotland television series set in the western isles of Scotland in the 1930s, based on the Para Handy books by Neil Munro. It starred Roddy McMillan as Peter...
Document 2	(Title: The Tales of Para Handy) The series followed the "Vital Spark's" adventures around the coastal waters of west Scotland and the various schemes that Para Handy would get himself and his crew involved in...
Document 3	(Title: Para Handy - Master Mariner) several dramatizations of the "Para Handy" stories, being followed by "The Vital Spark" in 1965-6 (remade in 1973-4) starring Roddy McMillan in the title role, and by **"The Tales of Para Handy" in 1994-5, starring Gregor Fisher.** Season 1 Para Handy - Master Mariner Para Handy - Master Mariner is a Scottish television series set in the western isles of Scotland in the 1930s, based on the Para Handy books by Neil Munro. It starred Duncan Macrae as **Peter "Para Handy" MacFarlane, captain of the puffer "Vital Spark"**...
Document 4	(Title: Vital Spark) Gregor Fisher in the lead role alongside Sean Scanlan as Dougie, Andrew Fairlie as Sunny Jim and Rikki Fulton as Dan McPhail. This ran for two series, a total of nine episodes, in 1994 and 1995...
Document 5	(Title: Vital Spark) Master Mariner, which ran from 1959–60, starred Duncan Macrae (Para Handy), Roddy McMillan (The Mate), and John Grieve (Dan MacPhail, the engineer). Six episodes were made...
Ground-truth Answer	Para Handy
No Retrieval	Rab C. Nesbitt. ×
With Retrieval	The Vital Spark ×
SURE	Para Handy ✓
IRCAN	The Vital Spark ×
CARE	**Para Handy** ✓
Question 2	**In the context of UK government, for what does the C stand in the acronym COBRA?**
Document 1	(Title: Civil Contingencies Secretariat) Civil Contingencies Committee, often informally referred to as COBRA from the name of the room used, is a forum for ministers and senior officials to discuss...
Document 2	(Title: Cabinet Office Briefing Rooms) committee after the 7 July 2005 London bombings and at other intervals from 2005 to 2007, was highly critical of its workings in his book "The Terrorist Hunters". A single photo of one of the rooms in COBR was released in 2010 in response to a Freedom of Information Act request. Cabinet Office Briefing Rooms The **Cabinet Office Briefing Rooms (COBR, often mistakenly referred to as COBRA)** are a group of meeting rooms in the Cabinet Office at 70 Whitehall, often used for different committees which co-ordinate the actions of bodies within **the Government of the United Kingdom** in response...
Document 3	(Title: Cabinet Office Briefing Rooms) **The Cabinet Office Briefing Rooms (COBR, often mistakenly referred to as COBRA)** are a group of meeting rooms in the Cabinet Office at 70 Whitehall, often...
Document 4	(Title: The Cobra (novel)) The Cobra is a 2010 thriller novel by Frederick Forsyth about the international cocaine trade. In it, an unnamed Obama-like (said to have a wife named 'Michelle' and...
Document 5	(Title: COBRA (avant-garde movement)) COBRA (or CoBrA) was a European avant-garde movement active from 1948 to 1951. The name was coined in 1948 by Christian Dotremont from the initials of...
Ground-truth Answer	Cabinet
No Retrieval	Cabinet ✓
With Retrieval	There is no clear answer in the given passages. ×
SURE	Civil ×
IRCAN	Room ×
CARE	**Cabinet** ✓

among multiple retrieved documents that provide mutually conflicting information. Extending CARE to more diverse generation settings and inter-document conflict resolution is an important direction for future work.

This work raises no specific ethical issues. It excludes human and animal subjects as well as any personal or sensitive data. The datasets utilized are publicly available and legally licensed for academic research. The modeling approaches and experimental methods strictly adhere to community-recognized ethical standards and are not designed for harmful, malicious, or discriminatory outcomes. Accordingly, we believe this research upholds the principles of responsible AI and entails no foreseeable risks of misuse.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the contributions and scope of the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitation of our work in Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: While this work does not provide a formal theoretical analysis, it is based on the assumption that knowledge-conflict settings can be associated with distinguishable patterns in the model's intermediate representations. We support this assumption through extensive empirical evidence.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide a comprehensive and detailed experimental setup in Section 5.1 and Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: All experiments are conducted on publicly available datasets, and we release our code along with detailed instructions to enable reproduction of the results reported in this paper.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide experimental details in Section 5.1 and Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report statistical significance information in Table 1. We also provide additional robustness analyses in Appendix D.1 to further support the reliability of the reported trends.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the GPU resources used for training and evaluation in Section 5.1, and provide runtime measurements in Appendix E.6.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in our paper conforms with the NeurIPS Code of Ethics in every respect.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the broader impacts of our work in Appendix G.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly credit the creators of all existing assets used in this work through citations.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We have released the code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs are a core component of both the proposed method and the experiments. We describe the evaluated LLM backbones, checkpoints, and implementation details in Section 5.1.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.