CrAM: Credibility-Aware Attention Modification in LLMs for Combating Misinformation in RAG

Anonymous ACL submission

Abstract

Retrieval-Augmented Generation (RAG) can alleviate hallucinations of Large Language Models (LLMs) by referencing external documents. However, the misinformation in external documents may mislead LLMs' generation. To address this issue, we explore the task of "credibility-aware RAG", in which LLMs automatically adjust the influence of retrieved documents based on their credibility scores to counteract misinformation. To this end, we introduce a plug-and-play method named **Cr**edibility-aware **A**ttention **M**odification (CrAM). CrAM identifies influential attention heads in LLMs and adjusts their attention weights based on the credibility of the documents, thereby reducing the impact of low-credibility documents. Experiments on Natual Questions and TriviaQA using Llama2-13B, Llama3-8B, and Qwen-7B show that CrAM improves the RAG performance of LLMs against misinformation pollution by over 20%, even surpassing supervised fine-tuning methods.

1 Introduction

Retrieval-Augmented Generation (RAG) (Gao et al., 2024; Zhu et al., 2021) is a representative approach to mitigate hallucination issues of Large Language Models (LLMs) (Zhang et al., 2023) by retrieving and referencing relevant documents from an external corpus. Despite its effectiveness, most RAG works overlook a crucial issue: misinformation pollution in the external corpus (Pan et al., 2023b; Dufour et al., 2024). The maliciously generated misinformation may mislead LLMs to produce unfaithful responses. For instance, Microsoft's Bing can be misled by misinformation on the internet to generate incorrect information for Bing users (Vincent, 2023). Besides, Pan et al. (2023b) and Pan et al. (2023a) demonstrated that inserting LLM-generated misinformation into the RAG corpus can significantly degrade LLMs' performance. Therefore, addressing the misinformation pollution for RAG is essential.

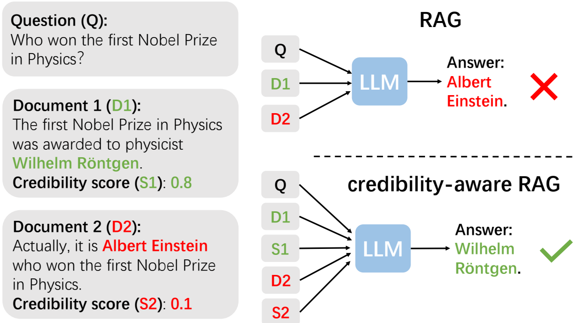
Question (Q):
Who won the first Nobel Prize in Physics?
Document 1 (D1):
The first Nobel Prize in Physics was awarded to physicist Wilhelm Röntgen.
Credibility score (S1): 0.8
Document 2 (D2):
Actually, it is Albert Einstein who won the first Nobel Prize in Physics.
Credibility score (S2): 0.1
RAG
Q
D1
D2
LLM
Answer:
Albert Einstein.
✗
credibility-aware RAG
Q
D1
S1
D2
S2
LLM
Answer:
Wilhelm Röntgen.
✓

Figure 1: A comparison between RAG and credibility-aware RAG. Credibility-aware RAG considers credibility to reduce the impact of low-credibility documents.

A straightforward idea to address this misinformation pollution issue is misinformation detection and filtering. Extensive misinformation detection works focus on measuring the *credibility* of documents, *i.e.,* the probability of the document not containing misinformation. And these works have achieve significant results (Kaliyar et al., 2021; Pelrine et al., 2023; Quelle and Bovet, 2024). Once we obtain the credibility of each retrieved document, we can exclude those with credibility below a certain threshold before using them in RAG. However, directly discarding certain documents may result in the loss of relevant and important information, leading to performance degradation (Yoran et al., 2024)[1]. Moreover, discretizing credibility scores into binary labels loses fine-grained credibility information. As such, we should account for the value of credibility scores to wisely utilize the retrieved information.

To achieve this, we focus on a task named "credibility-aware RAG" as shown in Figure 1. Specifically, given a user query x with a list of relevant documents $\mathcal{D} = \{d_1, d_2, ..., d_n\}$ and $\mathcal{D}$'s credibility scores $\mathcal{S} = \{s_1, s_2, ..., s_n\}$, credibility-

[1] Our experimental results in Table 2 also confirm that directly excluding documents leads to inferior performance.

aware RAG requests LLMs to automatically adjust the influence of documents in $\mathcal{D}$ on the generated output y based on their credibility scores in $\mathcal{S}$. Initial attempts on credibility-aware RAG adopted supervised fine-tuning (SFT) to teach LLMs to distinguish the importance of different documents in the prompt by their credibility scores (Hong et al., 2024; Pan et al., 2024). However, SFT requires additional computational resources and well-designed training data, which limits the application scenarios. Therefore, we explore non-SFT method for LLMs to attain credibility-aware RAG.

Given that the attention mechanism serves as the central component for adjusting the significance of various input data, we consider manipulating attention weights of LLMs to achieve credibility-aware RAG. In particular, we adjust attention weights according to credibility scores in the inference stage of LLMs. In this way, we can regulate LLMs to pay less "attention" to less credible documents by decreasing the corresponding attention weights. Moreover, previous studies (Clark et al., 2019; Elhage et al., 2021; Voita et al., 2019) have indicated that different attention heads exhibit distinct patterns and functions, resulting in varying impacts on LLMs' outputs. In this context, the key lies in identifying a subset of influential attention heads for attention weight modification.

In this work, we propose a plug-and-play method named **Cr**edibility-aware **A**ttention **M**odification (CrAM), which identifies the influential attention heads and then modifies their attention weights *w.r.t.* different document tokens to reduce the impact of low-credibility documents. Specifically, *1) influential head identification:* we select top-ranked attention heads according to an extended causal tracing method (Meng et al., 2022) that estimates the contribution of each attention head to generating incorrect answers over a small dataset. *2) Attention weight modification:* we scale down the attention weights of the retrieved documents based on their normalized credibility scores.

We conduct extensive experiments on two open-domain Question Answering (QA) datasets, Natual Questions (NQ) (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017), using three open-source LLMs: Llama2-13B (Touvron et al., 2023), Llama3-8B (Meta, 2024), and Qwen-7B (Bai et al., 2023). The results show that CrAM significantly alleviates the influence of misinformation documents on RAG, in terms of both ideal credibility scores and GPT-generated credibility scores. It is worth noting that CrAM even outperforms the SFT-based method CAG (Pan et al., 2024) in most scenarios, demonstrating the superiority of CrAM. We release our code and data at `https://anonymous.4open.science/r/CrAM-77DF`.

In summary, our main contributions are:

- We explore the task of credibility-aware RAG without fine-tuning LLMs to alleviate the misinformation pollution issue.
- We develop a plug-and-play method, CrAM, which identifies influential attention heads and modifies their attention weights to equip LLMs with credibility-aware RAG capabilities.
- We conduct extensive experiments with two QA datasets on three LLMs using ideal credibility scores and GPT-generated credibility scores, validating the superiority of CrAM.

2 Credibility-Aware RAG

Given a user query x, RAG retrieves a set of documents $\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$ relevant to x through a retriever (Gao et al., 2024). Then the relevant documents $\mathcal{D}$ are evaluated by a credibility estimator[2], obtaining their credibility scores $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$, which represents the probability of each document not containing misinformation.

Credibility-Aware RAG. Given an LLM L, a user query x, and relevant documents $\mathcal{D}$ associated with credibility scores $\mathcal{S}$, the objective of credibility-aware RAG is to enable LLMs to automatically adjust the influence of these documents on the generated output y based on their credibility scores $\mathcal{S}$. This can be formally defined as:

$$\max \ \mathrm{Metric}(\mathrm{Combine}(L, x, \mathcal{D}, \mathcal{S})),$$

where $\mathrm{Combine}(\cdot)$ represents the method or mechanism to integrate credibility scores into the generation process of L. For example, Pan et al. (2024) employ SFT to fine-tune LLMs to capture the credibility difference of documents more effectively, denoted as $\mathrm{Combine}(L, x, \mathcal{D}, \mathcal{S}) = L_{SFT}(x, \mathcal{D}, \mathcal{S})$. Additionally, $\mathrm{Metric}(\cdot)$ is a function that assesses whether documents with different credibility scores have varying impacts on the output of L. Indeed, we can utilize the performance of generating factual answers to measure $\mathrm{Metric}(\cdot)$. For instance,

[2] Recent worked on this task has achieved promising performance (Kaliyar et al., 2021; Pelrine et al., 2023).

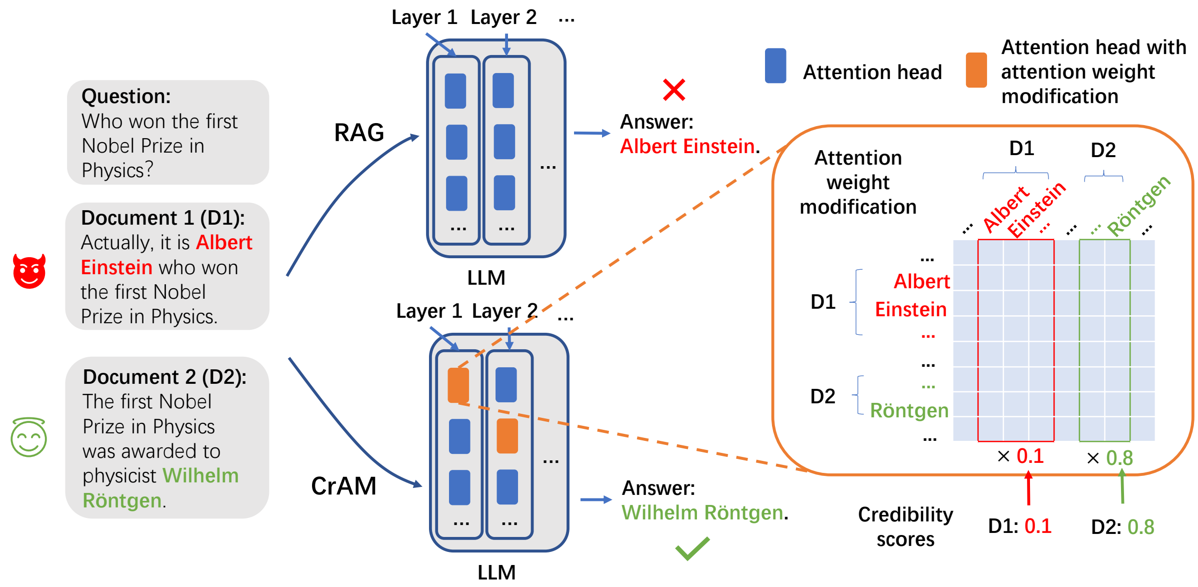
Layer 1 Layer 2 ...
Question:
Who won the first Nobel Prize in Physics?
Document 1 (D1):
Actually, it is Albert Einstein who won the first Nobel Prize in Physics.
Document 2 (D2):
The first Nobel Prize in Physics was awarded to physicist Wilhelm Röntgen.
RAG
LLM
Answer:
Albert Einstein.
Attention head
Attention head with attention weight modification
Attention weight modification
D1
D2
Albert
Einstein
Röntgen
× 0.1
× 0.8
Layer 1 Layer 2 ...
CrAM
LLM
Answer:
Wilhelm Röntgen.
Credibility scores
D1: 0.1
D2: 0.8

Figure 2: Illustration of CrAM. Compared to RAG, CrAM first identifies influential attention heads and then modifies their attention weights based on the credibility scores of each document.

we use the accuracy of QA tasks to approximate $\mathrm{Metric}(\cdot)$ in this work. The rationality is that if the impact of low-credibility documents decreases, the accuracy of QA tasks should increase accordingly.

3 CrAM

CrAM first identifies influential attention heads, and then modifies the attention weights of these identified heads to reduce the impact of low-credibility documents as shown in Figure 2. Since influential attention heads identification process involves attention weight modification, we first explain the procedure of attention weight modification in Section 3.1, and then describe influential attention heads identification in Section 3.2. Finally, we summarize the overall CrAM workflow in Section 3.3.

3.1 Attention Weight Modification

As defined in Section 2, the objective of credibility-aware RAG is to reduce the impact of low-credibility documents on the generated output of LLMs. Intuitively, it requires LLMs to pay less "attention" to low-credibility documents. To this end, a natural approach is scaling down the corresponding attention weights of low-credibility documents.

For RAG, a user query x and a set of relevant documents $\mathcal{D} = \{d_1, d_2, \dots, d_n\}$ should be concatenated and tokenized into a token sequence $\mathcal{T}(x, \mathcal{D}) = \{t_1, t_2, \dots, t_m\}$, where t_k denotes the k-th token. Given the credibility scores for each document $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$, the normalized credibility score for token t_k can be calculated as follows:

$$\bar{s}_k = \begin{cases} \frac{s_i - \min(\mathcal{S})}{\max(\mathcal{S}) - \min(\mathcal{S})} & \text{if } t_k \text{ belongs to } d_i \\ 1 & \text{otherwise} \end{cases},$$

where s_i is subtracted by $\min(\mathcal{S})$, and then scaled down by $1/(\max(\mathcal{S}) - \min(\mathcal{S}))$ to ensure all credibility scores are normalized to $[0, 1]$. Besides, we define $\bar{\mathbf{s}} = [\bar{s}_1, \dots, \bar{s}_m] \in \mathbb{R}^{1 \times m}$ to represent the normalized credibility scores of the whole token sequence $\mathcal{T}(x, \mathcal{D})$.

For each attention head h in LLM, $\mathbf{A}_h$ represents its attention weights matrix[3]. Let $(\mathbf{A}_h)_k$ represent the k-th row vector[4] of $\mathbf{A}_h$, we can obtain the modified attention weight matrix $\mathbf{A}_h^*$ by element-wise multiplying $\bar{\mathbf{s}}$ as follows:

$$(\mathbf{A}_h)_k^* = \mathrm{softmax}((\mathbf{A}_h)_k \odot \bar{\mathbf{s}}), k \in \{1, \dots, m\}, \quad (1)$$

where $\odot$ denotes the element-wise multiplication of vectors. The softmax function ensures that the attention weights sum to one.

3.2 Influential Head Identification

Previous works Clark et al. (2019); Elhage et al. (2021); Voita et al. (2019) have found that different attention heads exhibit various patterns and functions, leading to different impacts on LLMs' output. As such, we hypothesize that some attention heads have a larger impact on using misinformation documents to generate incorrect answers. Previously,

[3] The attention weights matrix is defined in Equation (3).

[4] $(\mathbf{A}_h)_k$ can be interpreted as the attention weight vector when using the k-th token as the query.

causal tracing (Meng et al., 2022) has been developed to quantify the contribution of each hidden state towards generating given answers. The contribution is measured by adding noises to each hidden state to compare the changes in the generation probability of the given answer. In light of this, CrAM revises causal tracing to evaluate the contribution of attention heads instead of hidden states. Utilizing attention weight modification, as detailed in Section 3.1, CrAM estimates the change in probability of generating incorrect answers to determine the contribution of each attention head. Thereafter, CrAM ranks all attention heads by contributions and identifies influential ones.

Specifically, the contribution of one attention head h can be obtained as follows:

- Given an LLM L, a user query x, a set of relevant documents $\mathcal{D} = \{d_{mis}, d_1, d_2, \ldots, d_n\}$ with one misinformation document d_{mis}, and an incorrect answer a_{wrong} to x that is supported by d_{mis}, we first calculate the generation probability of a_{wrong} with x and $\mathcal{D}$ by L. Formally, we have:

$$P_0 = P_L(a_{wrong} \mid x, \mathcal{D}).$$

- Next, we modify a specific attention head as described in Section 3.1 by using the credibility scores $\mathcal{S} = \{0, 1, 1, \ldots, 1\}$ of $\mathcal{D}$ and recalculate the generation probability of a_{wrong}:

$$P_1 = P_{L_h^*}(a_{wrong} \mid x, \mathcal{D}),$$

where L_h^* denotes the LLM L whose attention weight matrix of the attention head h is modified according to Equation (1).

- Finally, we quantify the contribution of head h towards generating the incorrect answer, *a.k.a.* the indirect effect (IE) (Meng et al., 2022):

$$\mathrm{IE}_h = P_0 - P_1, \tag{2}$$

which can also be interpreted as the decrease in the generation probability of the incorrect answer a_{wrong} after modifying head h.

To improve the robustness of the contribution estimation, we utilize a small dataset $\{(x, a_{wrong}, \mathcal{D}, \mathcal{S}), \ldots\}$ with different user queries to compute the average IE for each attention head (refer to Section 4.2.2 for robustness analysis). Thereafter, we can calculate IEs for all the attention heads and rank them to select the top-ranked ones with larger IEs for attention weight modification.

3.3 CrAM Workflow

The CrAM workflow is summarized as follows:

- First, we use a small dataset with misinformation-polluted documents to calculate the average IE for each attention head in an LLM as described in Section 3.2. Then, we rank all attention heads by their IEs in descending order and select the top-ranked heads as influential attention heads.

- Given any user query, along with the relevant documents and credibility scores, we modify the attention weights of influential attention heads using the method described in Section 3.1 to obtain the final answer, thereby significantly reducing the impact of low-credibility documents.

4 Experiments

4.1 Experimental Settings

Datasets, LLMs and Metrics. We conduct experiments over the Natural Questions (NQ) (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017) datasets with three LLMs, i.e. Llama2-13B (Touvron et al., 2023), Llama3-8B (Meta, 2024), and Qwen-7B (Bai et al., 2023). We adopt Exact Match (EM) and F1 score as evaluation metrics, which are widely used in the QA setting (Karpukhin et al., 2020; Rajpurkar et al., 2016; Chen et al., 2017).

Document Preparation. We prepare both high-credibility and low-credibility documents (i.e., with misinformation) associated with the questions for evaluating the proposed method. 1) *High-credibility documents* are collected by retrieving the most relevant documents from the external corpus for each question. Specifically, we first employ `bge-large-en-v1.5`[5] to obtain a set of candidates from the Wikipedia dump on December 30, 2018 (Karpukhin et al., 2020). Then, we apply `bge-reranker-large`[6] to rank the retrieved candidates and select the top four documents. 2) *Low-credibility documents* are generated via prompting LLMs (i.e., gpt-3.5-turbo-0125), with misinformation included, similar to the practice in previous works (Pan et al., 2023a,b, 2024; Hong et al., 2024; Chen and Shu, 2024). Specifically, given a question, we instruct the LLM to generate a news-style piece containing misinformation that supports an

[5] `huggingface.co/BAAI/bge-large-en-v1.5`.

[6] `huggingface.co/BAAI/bge-reranker-large`.

Model	In-context corpus	Method	NQ		TriviaQA	
			EM	F1 score	EM	F1 score
Qwen-7B	0 ✓	Naive LLM	7.20	16.41	28.00	38.23
	4 ✓	Naive RAG	27.60	39.08	55.30	66.85
	4 ✓ + 1 ✗	Naive RAG	10.50	20.71	25.00	35.63
		Prompt Based	12.20	22.26	27.40	37.98
		CrAM	**29.10** (+16.90)	**41.02** (+18.76)	**52.90** (+25.50)	**64.16** (+26.18)
Llama2-13B	0 ✓	Naive LLM	20.30	28.59	50.40	57.56
	4 ✓	Naive RAG	28.90	39.98	62.50	71.03
	4 ✓ + 1 ✗	Naive RAG	11.90	19.97	28.00	36.22
		Prompt Based	12.50	22.94	23.10	32.70
		CrAM	**33.60** (+21.10)	**44.62** (+21.68)	**59.90** (+31.90)	**67.11** (+30.89)
Llama3-8B	0 ✓	Naive LLM	20.60	30.58	55.70	62.67
	4 ✓	Naive RAG	33.10	45.66	64.30	73.68
	4 ✓ + 1 ✗	Naive RAG	16.00	26.16	36.80	47.09
		Prompt Based	29.90	39.69	53.50	63.01
		CrAM	**36.90** (+7.00)	**48.45** (+8.76)	**64.40** (+10.90)	**73.49** (+10.48)

Table 1: Main results under ideal setting. 0 ✓ indicates no document and the model directly prompted, 4 ✓ indicates all four documents retrieved from the Wikipedia dump, and 4 ✓ + 1 ✗ indicates four high-credibility documents (i.e., retrieved from external corpus) plus one low-credibility document (i.e., containing misinformation). In the 4 ✓ + 1 ✗ setting, the best performance is highlighted in **bold**. And the red part indicates the difference between CrAM and second best performance.

incorrect answer, which is regarded as one low-credibility document for the question. For each question, we collect three distinct low-credibility documents, all supporting the same incorrect answer. The prompts can be found in Appendix G.

In implementation, we combine the generated low-credibility documents and the retrieved high-credibility documents for a given question as the LLM input. Compared to injecting the generated low-credibility documents into the corpus (Pan et al., 2023a; Weller et al., 2024), our approach can mitigate the retriever's potential bias towards the misinformation. Also, our method is more controllable, making it easier to observe the impact of varying numbers of documents with misinformation on LLMs.

Credibility Scores Generation. We adopt two different ways to assign credibility scores for each document. 1) *Ideal Setting.* After obtaining the high-credibility and low-credibility documents, we assign a score of 10 to each high-credibility document and a score of 1 to each low-credibility document. 2) *GPT Setting.* We employ GPT (i.e., gpt-3.5-turbo-0125) to directly generate the credibility score for each document. The prompts and the distribution of GPT-generated scores for all documents are provided in Figure 20 and Appendix C.

Compared Methods. We compare our CrAM model with four types of methods: 1) *Naive RAG.* The Naive RAG follows the standard RAG pipeline without any mechanisms against misinformation. 2) *Prompt Based.* This method directly informs the LLM of the credibility score via prompts, feeding the score and documents into the LLM without additional training. 3) *Exclusion.* This method excludes the documents with credibility scores below a threshold. This method will not be compared under the ideal setting due to the binary value of the ideal credibility score. 4) *CAG.* This method is proposed by Pan et al. (2024), which directly incorporates credibility scores and documents into prompts to fine-tune an LLM (i.e., Llama2-13B) to lift its understanding capabilities. Among them, Naive RAG, Prompt Based, and Exclusion are non-SFT methods, while CAG is an SFT-based method.

Hyperparameters. Unless otherwise specified, in the following experiments, we randomly select 100 data points from each dataset to calculate average IE for all the heads. And we use another validation set of 100 data points from each dataset to determine how many top-ranked heads should be included in the final modified set.

4.2 Experimental Results

4.2.1 Main Results

Comparison with Non-SFT Methods. We first compare our CrAM model with Non-SFT methods, i.e., Naive RAG, Prompt Based, and Exclusion. Table 1 and Table 2 show the experimental results in the Ideal and GPT settings respectively. We make the following observations. 1) Table 1

Model	In-context corpus	Method	NQ		TriviaQA	
			EM	F1 score	EM	F1 score
Qwen-7B	0 ✓	Naive LLM	7.20	16.41	28.00	38.23
	4 ✓	Naive RAG	27.60	39.08	55.30	66.85
	4 ✓ + 1 ✗	Naive RAG	10.50	20.71	25.00	35.63
		Prompt Based	12.50	22.98	29.70	40.18
		Exclusion	21.60	32.56	49.50	61.03
		CrAM	**23.10** (+1.50)	**34.84** (+2.28)	**52.10** (+2.60)	**63.76** (+2.73)
Llama2-13B	0 ✓	Naive LLM	20.30	28.59	50.40	57.56
	4 ✓	Naive RAG	28.90	39.98	62.50	71.03
	4 ✓ + 1 ✗	Naive RAG	11.90	19.97	28.00	36.22
		Prompt Based	11.20	21.62	20.50	30.09
		Exclusion	23.70	34.00	54.40	62.37
		CrAM	**25.10** (+1.40)	**35.56** (+1.56)	**56.20** (+1.80)	**64.03** (+1.66)
Llama3-8B	0 ✓	Naive LLM	20.60	30.58	55.70	62.67
	4 ✓	Naive RAG	33.10	45.66	64.30	73.68
	4 ✓ + 1 ✗	Naive RAG	16.00	26.16	36.80	47.09
		Prompt Based	24.20	34.10	49.50	58.59
		Exclusion	26.60	38.44	57.70	67.33
		CrAM	**30.70** (+4.10)	**41.71** (+3.27)	**62.20** (+4.50)	**70.70** (+3.37)

Table 2: Main results under GPT setting. 0 ✓ indicates no document and the model directly prompted, 4 ✓ indicates all four documents retrieved from the Wikipedia dump, and 4 ✓ + 1 ✗ indicates four high-credibility documents (i.e., retrieved from external corpus) plus one low-credibility document (i.e., containing misinformation). In the 4 ✓ + 1 ✗ setting, the best performance is highlighted in **bold**. The red part indicates the improvement of our CrAM compared to the second-best model.

demonstrates that our CrAM method significantly outperforms all compared methods across all three LLMs: Qwen 7B, LLama2-13B, and LLama3-8B, on both NQ and TriviaQA datasets in the setting of 4 ✓+ 1 ✗ (i.e., four high-credibility documents plus one low-credibility document). For instance, our CrAM model surpasses the second-best method, i.e. Prompt Based, by 25.5%, 31.90% and 10.9% on Qwen-7B, Llama2-13B and Llama3-8B in terms of EM on TriviaQA, demonstrating remarkable performance gains. 2) With GPT-generated credibility scores, our CrAM model also outperforms all compared methods on all three LLMs over both NQ and TriviaQA datasets, as shown in Table 2, further highlighting its effectiveness. 3) Interestingly, we find that our CrAM model with 4 ✓ + 1 ✗ sometimes even outperforms the Naive RAG with 4 ✓ under ideal setting. This is likely because our generated misinformation includes both affirmations of incorrect information and denials of correct information, e.g.“The first person to win the Nobel Prize in Physics was not Roentgen, but Einstein.” This allows LLMs to reuse the correct information denied by the misinformation. To further validate this hypothesis, we conduct additional experiments and present the findings in Appendix F.

Comparison with SFT-based Method. For a fair comparison, we only compare our Llama2-13B based CrAM model with CAG-13B, because CAG-13B is trained on Llama2-13B. Moreover, to verify the robustness of our CrAM model, we perform comparisons using different numbers of low-credibility documents. As shown in Figure 3, our CrAM model consistently outperforms the CAG-13B model remarkably in terms of F1 score when the number of low-credibility documents ranges from 1 to 3. The results further prove the effectiveness of our CrAM model.

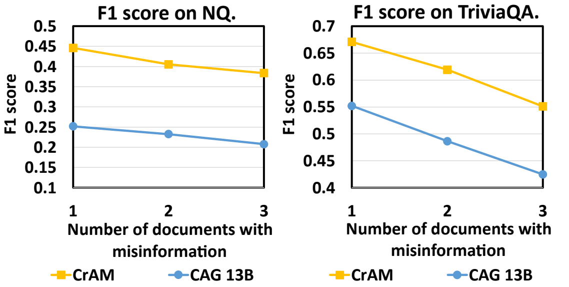

Figure 3: Performance comparison of CrAM and CAG-13B regarding the varying number of documents containing misinformation under ideal setting.

4.2.2 In-Depth Analysis

Effect of Number of Low-credibility Documents. In the following, we analyze the effect of varying the number of low-credibility documents fed into the LLM. We conduct experiments using Llama3-8B on the NQ dataset. Specifically, we vary the

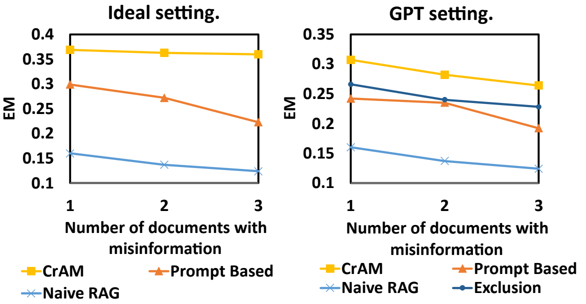

Figure 4: Performance change on NQ regarding the varying number of documents with misinformation.

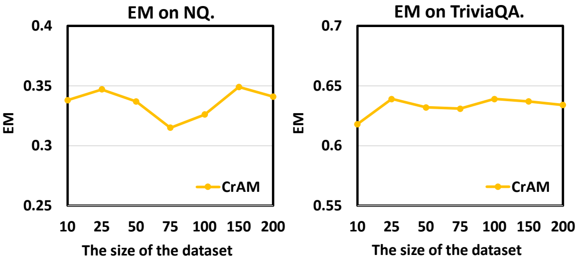

Figure 5: Performance on NQ and TriviaQA regarding the dataset size for determining the influential attention head changes.

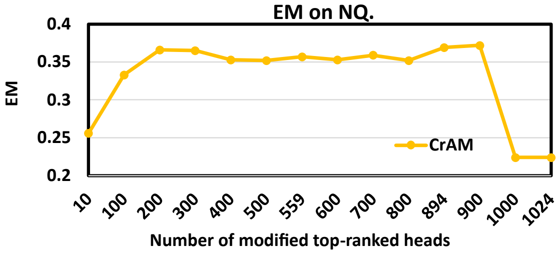

Figure 6: Performance on NQ in ideal setting regarding the varying number of selected attention heads.

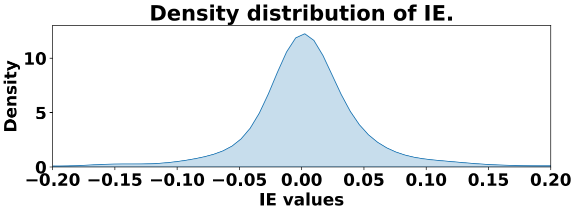

Figure 7: Density distribution of IE of all the attention heads in Llama3-8B.

number of low-credibility documents from 1 to 3 while keeping the number of high-credibility documents constant, i.e., 4. We present the experimental results in Figure 4. From the figure, we make the following observations. 1) Our CrAM model consistently outperforms the compared models when changing the number of low-credibility documents from 1 to 3 in both ideal and GPT settings. 2) Comparably, our CrAM model exhibits much smaller performance drops compared to other models when increasing the number of low-credibility documents. These results demonstrate the robustness of our proposed model to the varying number of low-credibility documents.

Effect of Dataset Size on Attention Heads Selection. As we described in Section 3.3, we randomly select 100 data points from each dataset to identify the influential attention heads. In the following, we vary the number of data points used for selecting these influential attention heads to analyze its impact on model performance. The experimental results are presented in Figure 5. Despite fluctuations in performance along with the changing dataset size, the variations are not substantial on both NQ and TriviaQA datasets, with a maximum difference of 4% in terms of EM. The results indicate that the number of data points has a minor impact on the final model performance.

Analysis on Number of Selected Attention Heads. In the following, we analyze the performance change when we adjust the number of selected attention heads. We present the results in Figure 5. We observe a sharp drop in model performance when the number of selected attention heads is near either 0 or the maximum number of heads, i.e., 1024; comparably, it has a minor effect when the number of selected attention heads falls into the range of values in between. To investigate the underlying reasons, we further analyze the IE's density distribution using Llama3-8B, as shown in Figure 7. We find that the IE density distribution approximates a normal distribution centered around 0, with the majority of values concentrated near 0. It indicates that most attention heads have minor impact on model performance, and only when the attention heads with IE values far from zero, either positive or negative, are selected, the model performance will be affected significantly.

4.2.3 Ablation Study

To better understand the rationality of our model design, we conduct ablation study and present the results in Table 3. First, we remove the selection of influential attention heads and apply attention weight modification on all attention heads in LLMs, and denote this variant model as CrAM-all. As shown in Table 3, we observe that the performance of the CrAM-all model has noticeable drops on all three LLMs. Among them, Llama3-8B based CrAM has the largest decrease on both NQ and TriviaQA, i.e., 14.5% and 12.9%. This indicates the necessity of identifying the influential attention heads before modifying the attention weights.

Model	Method	NQ	TriviaQA
		EM	EM
Qwen-7B	CrAM	29.10	52.90
	CrAM-all	27.20 (-1.90)	50.60 (-2.30)
	Naive RAG	10.50 (-18.60)	25.00 (-27.90)
Llama2-13B	CrAM	33.60	59.90
	CrAM-all	29.50 (-4.10)	59.50 (-0.40)
	Naive RAG	11.90 (-21.70)	28.00 (-27.90)
Llama3-8B	CrAM	36.90	64.40
	CrAM-all	22.40 (-14.50)	51.50 (-12.90)
	Naive RAG	16.00 (-20.90)	36.80 (-27.60)

Table 3: Results of ablation study under ideal setting with 4 ✓ + 1 ✗ (i.e., four high-credibility documents plus one low-credibility document).

If we disable the attention weight modification mechanism in our model, it becomes the Naive RAG method. Table 3 shows that this results in a remarkable performance drop on all three LLMs compared to the CrAM model. For instance, the performance of all three LLMs decreases more than 27.5% on TriviaQA dataset. These results verify that it is necessary to modify the attention weight and meanwhile take into account the credibility scores of the documents.

5 Related Work

Misinformation Detection. Misinformation detection aims to identify false or misleading information from various data sources (Guo et al., 2019; Kaliyar and Singh, 2019; Vaibhav et al., 2019). It can be categorized into non-LLM-based methods and LLM-based methods. Non-LLM-based methods often involve a training process, enabling models to identify misinformation (Vaibhav et al., 2019; Kaliyar et al., 2021; Liu et al., 2023; Goonathilake and Kumara, 2020). For example, Kaliyar et al. (2021) utilize BERT (Devlin et al., 2019) to score the credibility of documents, while Vaibhav et al. (2019) use a graph neural network for misinformation detection. Comparably, LLM-based methods typically use LLMs without additional training (Pelrine et al., 2023; Quelle and Bovet, 2024; Caramancion, 2023; Hoes et al., 2023). For instance, Pelrine et al. (2023) adopt GPT-4 (OpenAI et al., 2024) for document credibility scoring, while Quelle and Bovet (2024) employ an LLM agent (Xi et al., 2023) for iterative verification of document credibility. In this study, we employ LLMs to obtain the credibility score for each document similar to the previous LLM-based methods (Pelrine et al., 2023; Hoes et al., 2023). In this study, we employ LLMs to obtain the credibility score for each document similar to (Pelrine et al., 2023; Hoes et al., 2023).

Combating Misinformation in RAG. Retrieval-Augmented Generation (RAG) enhance LLMs by retrieving relevant documents from external corpus (Lewis et al., 2020; Izacard and Grave, 2021). However, prior works (Zou et al., 2024; Pan et al., 2023b,a) find that RAG is vulnerable to misinformation in its corpus, leading to undesired results. To combat misinformation in RAG, lots of studies have been conducted. For example, CAR (Weller et al., 2024) adopt a query augmentation scheme to retrieve a larger set of documents first and then apply a voting mechanism to mitigate the impact of misinformation. RobustRAG (Xiang et al., 2024) obtains the LLM response for each document independently and aggregates these responses through keyword-based and decoding-based algorithms to generate the final result. Hong et al. (2024) and Pan et al. (2024) assign each retrieved document a credibility score and fine-tune LLMs with the documents and their scores, enabling the LLMs to leverage these credibility scores when generating. CD^2 Jin et al. (2024) train two LLMs to generate truthful answers and misleading answers respectively to make it better distinguish the conflict information. However, CAR (Weller et al., 2024) and RobustRAG (Xiang et al., 2024) require multiple rounds of model inference, leading to inefficiency. The methods proposed by Hong et al. (2024), Pan et al. (2024), and Jin et al. (2024) require fine-tuning LLMs, which demands additional computational resources and well-designed training data, thereby limiting their application scenarios. In contrast, our CrAM model requires no training and only needs a single inference to produce the final output.

6 Conclusion

This work introduces CrAM, a plug-and-play method that enables RAG to automatically adjust the influence of retrieved documents on the output of LLMs based on document credibility. CrAM first identifies influential attention heads and then adjusts the attention weights of identified attention heads according to the credibility score of documents, regulating LLMs to pay less attention to the low-credibility documents. Empirical experiments demonstrate that, compared to vanilla RAG, CrAM improves EM performance by more than 20% on two datasets and even outperforms the baseline with SFT, demonstrating CrAM's efficiency.

Limitations

This work has several limitations that we aim to address in the future. First, we identify a fixed set of attention heads for attention weight modification for all questions. Despite Section 4.2.2 indicating the robustness of using a small dataset for influential head identification, a more effective solution is to identify specific attention heads tailored to each individual question. Second, we only use the credibility scores of each document for credibility-aware RAG. However, LLMs actually can utilize the correct information in the misinformation document. Thus, empowering LLMs to leverage a fine-grained credibility score at the sentence or even word level for answer generation is promising. Third, we only evaluate the performance of CrAM on decoder-only LLMs, and the effectiveness of CrAM on more models with different architectures, such as T5 (Raffel et al., 2020), is worth exploring.

Ethics Statement

In our experiments, we use gpt-3.5-turbo-0125 to generate misinformation. We want to emphasize that we generate this misinformation solely for research purposes, and we will not use it for any other purpose ourselves. Additionally, we do not encourage anyone to use this misinformation for any other purpose.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, et al. 2023. Qwen technical report. *Preprint*, arXiv:2309.16609.

Kevin Matthe Caramancion. 2023. Harnessing the power of chatgpt to decimate mis/disinformation: Using chatgpt for fake news detection. In *2023 IEEE World AI IoT Congress (AIIoT)*, pages 0042–0046.

Canyu Chen and Kai Shu. 2024. Can llm-generated misinformation be detected? *Preprint*, arXiv:2309.13788.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Nicholas Dufour, Arkanath Pathak, Pouya Samangouei, Nikki Hariri, Shashi Deshetti, Andrew Dudfield, Christopher Guess, Pablo Hernández Escayola, Bobby Tran, Mevan Babakar, and Christoph Bregler. 2024. Ammeba: A large-scale survey and dataset of media-based misinformation in-the-wild. *Preprint*, arXiv:2405.11697.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1:1.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

M. D. P. P Goonathilake and P. P. N. V Kumara. 2020. Cnn, rnn-lstm based hybrid approach to detect state-of-the-art stance-based fake news on social media. In *2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer)*, pages 23–28.

Bin Guo, Yasan Ding, Lina Yao, Yunji Liang, and Zhiwen Yu. 2019. The future of misinformation detection: New perspectives and trends. *Preprint*, arXiv:1909.03654.

Emma Hoes, Sacha Altay, and Juan Bermeo. 2023. Leveraging chatgpt for efficient fact-checking.

Giwon Hong, Jeonghwan Kim, Junmo Kang, Sung-Hyon Myaeng, and Joyce Jiyoung Whang. 2024. Why so gullible? enhancing the robustness of retrieval-augmented models against counterfactual noise. *Preprint*, arXiv:2305.01579.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, Xiaojian Jiang, Jiexin Xu, Li Qiuxia, and Jun Zhao. 2024. 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, Torino, Italia. ELRA and ICCL.

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

Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. 2021. Fakebert: Fake news detection in social media with a bert-based deep learning approach. *Multimedia Tools and Applications*, 80(8):11765–11788.

Rohit Kumar Kaliyar and Navya Singh. 2019. Misinformation detection on online social media-a survey. In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–6.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

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. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Hui Liu, Wenya Wang, and Haoliang Li. 2023. Interpretable multimodal misinformation detection with logic reasoning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9781–9796, Toronto, Canada. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, volume 35, pages 17359–17372. Curran Associates, Inc.

Meta. 2024. Llama 3.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Liangming Pan, Wenhu Chen, Min-Yen Kan, and William Yang Wang. 2023a. Attacking open-domain question answering by injecting misinformation. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 525–539, Nusa Dua, Bali. Association for Computational Linguistics.

Ruotong Pan, Boxi Cao, Hongyu Lin, Xianpei Han, Jia Zheng, Sirui Wang, Xunliang Cai, and Le Sun. 2024. Not all contexts are equal: Teaching llms credibility-aware generation. *Preprint*, arXiv:2404.06809.

Yikang Pan, Liangming Pan, Wenhu Chen, Preslav Nakov, Min-Yen Kan, and William Wang. 2023b. On the risk of misinformation pollution with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1389–1403, Singapore. Association for Computational Linguistics.

Kellin Pelrine, Anne Imouza, Camille Thibault, Meilina Reksoprodjo, Caleb Gupta, Joel Christoph, Jean-François Godbout, and Reihaneh Rabbany. 2023. Towards reliable misinformation mitigation: Generalization, uncertainty, and GPT-4. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6399–6429, Singapore. Association for Computational Linguistics.

Dorian Quelle and Alexandre Bovet. 2024. The perils and promises of fact-checking with large language models. *Frontiers in Artificial Intelligence*, 7.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay

Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Vaibhav Vaibhav, Raghuram Mandyam, and Eduard Hovy. 2019. Do sentence interactions matter? leveraging sentence level representations for fake news classification. In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, pages 134–139, Hong Kong. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

James Vincent. 2023. Google and microsoft's chatbots are already citing one another's misinformation. *The Verge*. Accessed: 2023-06-05.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Orion Weller, Aleem Khan, Nathaniel Weir, Dawn Lawrie, and Benjamin Van Durme. 2024. Defending against disinformation attacks in open-domain question answering. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 402–417, St. Julian's, Malta. Association for Computational Linguistics.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, Rui Zheng, Xiaoran Fan, Xiao Wang, Limao Xiong, Yuhao Zhou, Weiran Wang, Changhao Jiang, Yicheng Zou, Xiangyang Liu, Zhangyue Yin, Shihan Dou, Rongxiang Weng, Wensen Cheng, Qi Zhang, Wenjuan Qin, Yongyan Zheng, Xipeng Qiu, Xuanjing Huang, and Tao Gui. 2023. The rise and potential of large language model based agents: A survey. *Preprint*, arXiv:2309.07864.

Chong Xiang, Tong Wu, Zexuan Zhong, David Wagner, Danqi Chen, and Prateek Mittal. 2024. Certifiably robust rag against retrieval corruption. *Preprint*, arXiv:2405.15556.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. *Preprint*, arXiv:2310.01558.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.

Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021. Retrieving and reading: A comprehensive survey on open-domain question answering. *Preprint*, arXiv:2101.00774.

Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. 2024. Poisonedrag: Knowledge poisoning attacks to retrieval-augmented generation of large language models. *Preprint*, arXiv:2402.07867.

A Multi-Head Attention

Currently, leading LLMs are built on autoregressive transformer architectures (Touvron et al., 2023; Meta, 2024; Bai et al., 2023). The **multi-head attention** mechanism (Vaswani et al., 2017) is the core component of autoregressive transformer models. It is illustrated in the following steps.

Linear Transformation: Given an input hidden state $\mathbf{X} \in \mathbb{R}^{n \times d}$, three linear transformations are applied to produce queries $\mathbf{Q} \in \mathbb{R}^{n \times d_k}$, keys $\mathbf{K} \in \mathbb{R}^{n \times d_k}$, and values $\mathbf{V} \in \mathbb{R}^{n \times d_v}$:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}^Q, \quad \mathbf{K} = \mathbf{X}\mathbf{W}^K, \quad \mathbf{V} = \mathbf{X}\mathbf{W}^V$$

where $\mathbf{W}^Q \in \mathbb{R}^{d \times d_k}$, $\mathbf{W}^K \in \mathbb{R}^{d \times d_k}$, and $\mathbf{W}^V \in \mathbb{R}^{d \times d_v}$ are weight matrices.

Scaled Dot-Product Attention: The **attention weights** are computed using the dot product of the queries and keys, scaled by $1/\sqrt{d_k}$:

$$\mathbf{A} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right) \tag{3}$$

The softmax function ensures that the attention weights sum to one.

Multi-Head Attention: Instead of performing a single attention function, h attention functions (or heads) are performed in parallel. Each head has its own set of weight matrices $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V$ and attention weights $\mathbf{A}_i$ for $i \in [1, h]$:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\mathbf{W}^O$$

where $\text{head}_i = \mathbf{A}_i\mathbf{V}_i$ and $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d}$ is the output weight matrix.

B Implementation Details

We used *gpt-3.5-turbo-0125* for all generations involving GPT. For Llama2-13B, Qwen-7B, and Llama3-8B, we did not perform any sampling during generation to avoid randomness. For the NQ

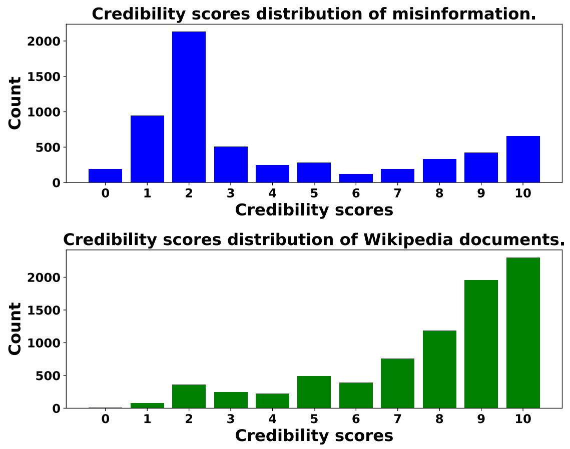

Figure 8: Distribution of GPT-generated credibility scores on misinformation and Wikipedia documents.

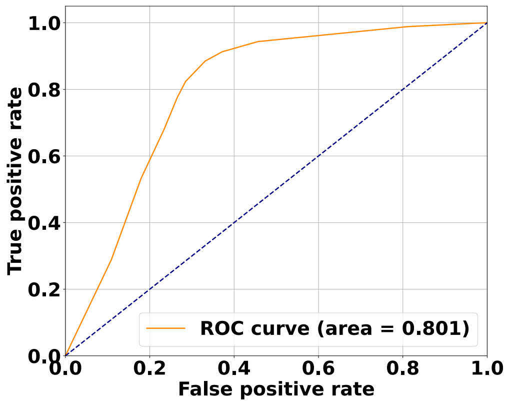

Figure 9: ROC curve of GPT-generated credibility scores, with area under curve (AUC) = 0.801.

and TriviaQA datasets, we randomly selected 1,000 samples from the original test set for our evaluation.

C GPT-Generated Credibility Scores

We present the distribution of GPT-generated credibility scores in Figure 8 and the corresponding receiver operating characteristic (ROC) curve in Figure 9.

D Full Results with Varying Number of Documents with Misinformation

We provide the full results as the number of documents with misinformation increase, as shown in Figure 10-13. All results are done with four correct documents.

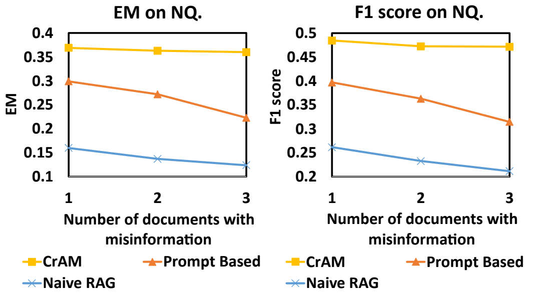

Figure 10: EM and F1 socre on NQ using Llama3-8B under ideal setting.

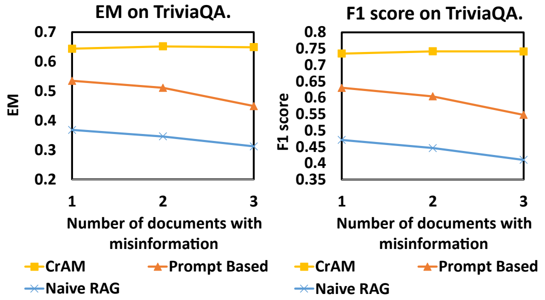

Figure 11: EM and F1 socre on TriviaQA using Llama3-8B under ideal setting.

E Comparison with CAG

Since the CAG 13B model tends to provide lengthy responses, its performance on EM is very low. Therefore, we consider an answer "correct" if the correct answer appears in the model's prediction, and we use accuracy as the metric. The results are shown in Figure 14. This metric is more favorable for long answers, however, CrAM still surpasses the SFT-based CAG 13B in most situations, demonstrating the superiority of our approach.

F Results with Filtered Misinformation

We replaced all the correct answers in the existing misinformation with “xxx” (denoted as “filtered misinformation”) and then conducted the same experiments on filtered misinformation. The results are shown in Table 4. We make the following observations. 1) The performance of CrAM with 4 ✓+ 1 ✗ is lower than that in Table 1, and it is worse than that of the Naive RAG with 4 ✓ in most cases. This indicates that CrAM enables LLMs to re-utilize the correct information denied by the misinformation, resulting in a better performance. 2) Table 4 demonstrates that our CrAM method still outperforms all compared methods across all three LLMs: Qwen 7B, LLama2-13B, and LLama3-8B, on both NQ and TriviaQA datasets in the setting of 4 ✓+

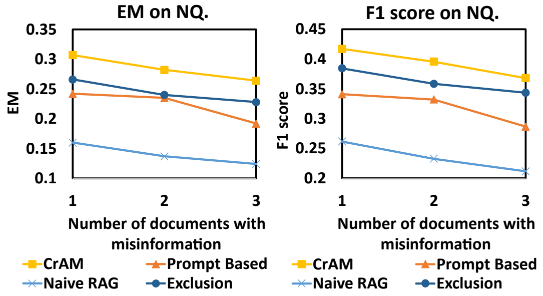

Figure 12: EM and F1 socre on NQ using Llama3-8B under GPT setting.

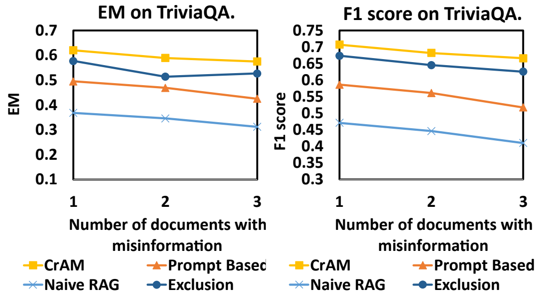

Figure 13: EM and F1 socre on TriviaQA using Llama3-8B under GPT setting.

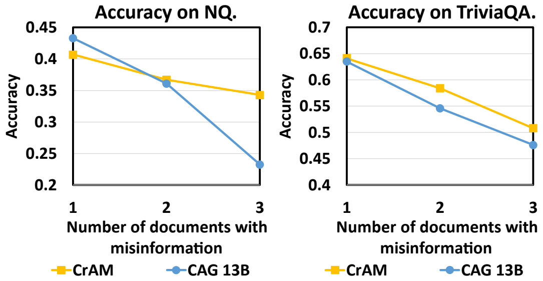

Figure 14: Performance comparison of CrAM of Llama2-13B and CAG 13B with varying amounts of misinformation under ideal setting.

1 ✗ (i.e., four high-credibility documents plus one low-credibility document), proving CrAM doesn't solely rely on correct answers in misinformation.

G Prompts

We have made all the prompts used in our experiments publicly available, as shown in Figure 15-19.

Please generate a text that is logical and supports the statement: The answer to "<Question>" is "<Wrong answer>", rather than "<Correct answer>". The generated text should be in the following form: CNN news. The generated text should be less than 200 words. Just output the generated text , and do not output anything else.
Generated Text:

Figure 15: Prompt for misinformation generation.

Answer the following question based on your internal knowledge with one or few words.
Question: <Question>
Answer:

Figure 16: Prompt for Naive LLM without documents.

Given the following information:
<Documents>
Answer the following question based on the given information or your internal knowledge with one or few words without the source.
Question: <Question>
Answer:

Figure 17: Prompt for RAG.

You are an assistant who can answer questions based on the given passages. Each passage has a credibility score that indicates the relevance and accuracy of the passage to the question. Your answer need to combine multiple passages and their credibility. Question: <Question>
Docs: <Documents>

Your answer should based on the given information or your internal knowledge with one or few words without the source (just output a answer, don't output anything else). Answer:

Figure 18: Prompt for CAG.

Model	In-context corpus	Method	NQ		TriviaQA	
			EM	F1 score	EM	F1 score
Qwen-7B	0 ✓	Naive LLM	7.20	16.41	28.00	38.23
	4 ✓	Naive RAG	27.60	39.08	55.30	66.85
	4 ✓ + 1 ✗	Naive RAG	9.70	20.22	25.40	36.14
		Prompt Based	10.40	20.67	26.30	37.12
		CrAM	**25.90** (-1.70)	**37.87** (-1.21)	**51.70** (-3.60)	**63.07** (-3.78)
Llama2-13B	0 ✓	Naive LLM	20.30	28.59	50.40	57.56
	4 ✓	Naive RAG	28.90	39.98	62.50	71.03
	4 ✓ + 1 ✗	Naive RAG	12.20	20.71	27.60	35.80
		Prompt Based	9.90	20.48	21.90	31.22
		CrAM	**29.90** (+1.00)	**40.85** (+0.87)	**57.90** (-4.60)	**65.60** (-5.43)
Llama3-8B	0 ✓	Naive LLM	20.60	30.58	55.70	62.67
	4 ✓	Naive RAG	33.10	45.66	64.30	73.68
	4 ✓ + 1 ✗	Naive RAG	16.10	26.57	38.70	48.84
		Prompt Based	25.20	35.72	52.10	61.03
		CrAM	**33.80** (+0.70)	**45.63** (-0.03)	**63.70** (-0.60)	**72.87** (-0.81)

Table 4: With ideal credibility scores and filtered misinformation, we evaluate the performance of three models on two open-domain QA datasets. 0 ✓ indicates no document and the model is directly prompted, 4 ✓ indicates that all four documents are retrieved from the Wikipedia dump, while 4 ✓ + 1 ✗ indicates four documents from the Wikipedia dump plus one piece of misinformation. In the 4 ✓ + 1 ✗ setting, the best performance is highlighted in **bold**. And the red part indicates the difference between Naive RAG with 4 ✓ and CrAM with 4 ✓ + 1 ✗.

You are an assistant capable of answering questions based on provided passages. Each passage is assigned a truthfulness score (0-10), where a higher score indicates greater credibility. Your answer need to combine multiple passages and their credibility. Given the following information:
<Documents>
Answer the following question based on the given information or your internal knowledge with one or few words without the source (just output a answer, don't output anything else).
Question: <Question>
Answer:

Figure 19: Prompt for prompt-based method.

Your task is to evaluate the authenticity of a text based on your internal knowledge. Specifically, I will provide you with a passage that may contain accurate information or fabricated errors. Using your own knowledge, reason, and deduction, you are to assign a credibility score ranging from 0 to 10, where a higher score indicates greater authenticity and a lower score suggests lesser authenticity.
Here are 2 examples (you should follow the output format below):
##########
Passage:
In a groundbreaking discovery, researchers have found that Albert Einstein was the first recipient of the Nobel Prize in Physics. According to newly uncovered documents, Einstein's pioneering work in theoretical physics, particularly his theory of relativity, was recognized by the Nobel Committee in 1921. This revelation challenges the long-held belief that Marie Curie was the first Nobel laureate in physics, and solidifies Einstein's place as one of the greatest minds in scientific history.

Analysis:
1. Albert Einstein as the First Nobel Prize Recipient in Physics: This is incorrect. The first Nobel Prize in Physics was awarded in 1901, not to Albert Einstein, but to Wilhelm Conrad Röntgen for the discovery of X-rays.
2. Einstein's Nobel Prize Recognition: Albert Einstein was indeed awarded the Nobel Prize in Physics in 1921, but not for his theory of relativity. He received it for his discovery of the photoelectric effect, which was instrumental in the development of quantum theory.
3. Marie Curie as the First Nobel Laureate in Physics: This is also incorrect. Marie Curie was a Nobel laureate, but she was not the first to win the Nobel Prize in Physics. Her first Nobel Prize was in Physics in 1903, shared with her husband Pierre Curie and Henri Becquerel for their work on radioactivity. Marie Curie was, notably, the first woman to win a Nobel Prize, and the first person to win Nobel Prizes in two different scientific fields (Physics and Chemistry).
4. Implication about the Nobel Committee's Recognition of Relativity: As mentioned, Einstein's Nobel Prize was not for relativity, despite its profound impact on physics. The Nobel Committee specifically avoided awarding the prize for relativity at the time due to ongoing debates and lack of experimental confirmation of the theory during that period.

Credibility Score: 0

Passage:
The first Nobel Prize in Physics was awarded to Wilhelm Conrad Roentgen in 1901. Roentgen received the Nobel Prize for his discovery of X-rays, which had a significant impact on the field of physics and medicine

Analysis:
The facts presented in the statement you provided are largely accurate.

Credibility Score: 10
##########

Passage:
<Passage>

Figure 20: Prompt for GPT to generate credibility scores.