Dynamic Attention-Guided Context Decoding for Mitigating Context Faithfulness Hallucinations in Large Language Models

Yanwen Huang[1,2,†], **Yong Zhang**[1,†], **Ning Cheng**[1,*],
Zhitao Li[1], **Shaojun Wang**[1], **Jing Xiao**[1],
[1] Ping An Technology (Shenzhen) Co., Ltd., China
[2] University of Electronic Science and Technology of China
{zhangyong203, chengning211}@pingan.com.cn

Abstract

Large language models (LLMs) often exhibit Context Faithfulness Hallucinations, where outputs deviate from retrieved information due to incomplete context integration. Our analysis reveals a strong correlation between token-level uncertainty and hallucinations. We hypothesize that attention mechanisms inherently encode context utilization signals, supported by probing analysis. Based on these insights, we propose **Dynamic Attention-Guided Context Decoding (DAGCD)**, a lightweight framework that leverages attention distributions and uncertainty signals in a single-pass decoding. Experiments on open-book QA datasets demonstrate DAGCD's effectiveness, yielding significant improvements in faithfulness and robustness while preserving computational efficiency.[1]

1 Introduction

Large Language Models (LLMs) (Brown et al., 2020; Achiam et al., 2023; Touvron et al., 2023) excel in generating fluent and contextually relevant responses. However, they often struggle with factual accuracy, especially when relying on external information. (Vu et al., 2023). Retrieval-Augmented Generation (RAG) (Lewis et al., 2020; Guu et al., 2020) mitigates this by grounding outputs in retrieved context, making it effective for tasks like question answering and reasoning (Gao et al., 2023; Fan et al., 2024). However, models often fail to faithfully utilize retrieved context, resulting in **Context Faithfulness Hallucinations**, where outputs deviate from the retrieved context (Huang et al., 2023a; Ji et al., 2023).

These hallucinations undermine the reliability of RAG systems, particularly in critical domains where factual accuracy is paramount(Chuang et al., 2024a). Existing methods, such as CAD (Shi et al., 2024b) and COIECD (Yuan et al., 2024), attempt to mitigate context faithfulness hallucinations by dynamically adjusting decoding distributions through token-level probability distribution comparisons or token-level uncertainty signals. While effective to some extent, these methods face several key limitations: limited interpretability, degraded performance when context-agnostic and context-aware outputs differ significantly, and computational complexity due to multiple decoding passes.

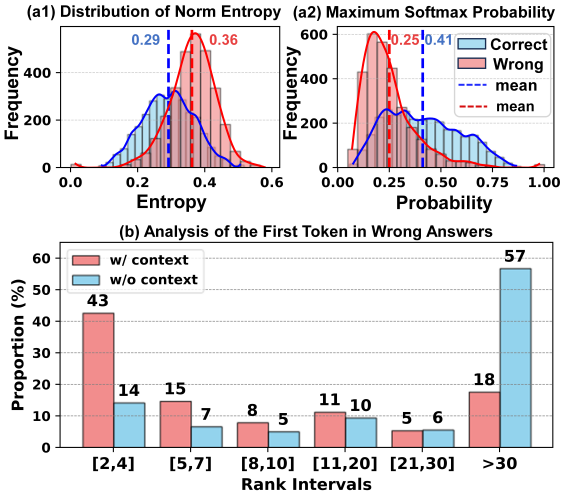

Figure 1: Analysis of the token-level probability distribution after context concatenation: (a1, a2) the model's uncertainty when generating correct versus wrong answers, measured by NE and MSP; (b) for wrong answers, the ranking of the golden answer token within the token-level probability distribution.

To better understand context faithfulness hallucinations and explore potential solutions, we take an internally-driven approach, analyzing intrinsic model signals that may explain why retrieval-augmented large language models struggle to utilize retrieved context faithfully (Liang et al., 2024). Motivated by prior research linking token-level uncertainty to factual hallucinations (Chuang et al., 2024b; Das et al., 2025), we examine whether

† Equal contribution.
* Corresponding author.
This work was done during Yanwen Huang's internship at Ping An Technology (Shenzhen) Co., Ltd., China.
[1]Our code is available at uestc-huangyw/DAGCD.

Findings of the Association for Computational Linguistics: ACL 2025, pages 5174–5193
July 27 - August 1, 2025

token-level probability distribution entropy correlates with context faithfulness hallucinations.

Our analysis reveals a strong correlation between higher uncertainty and context faithfulness hallucinations. Specifically, as shown in Figure 1 (a1), wrong answers exhibit higher entropy in the token-level probability distribution, indicating greater uncertainty in generation. Even for correct answers (Figure 1 (a2)), it often assigns low confidence to the highest-ranked token, suggesting incomplete integration of retrieved context. Notably, in incorrect responses, most gold answer tokens appear in the top 10 in the token-level probability distribution, but are not assigned the highest probability (Figure 1 (b)), implying that **the model identifies relevant context but fails to prioritize it effectively**.

These findings indicate that while models retrieve relevant context, they struggle to integrate and prioritize it during generation. Since most gold answer tokens appear within the top 10, an effective strategy to mitigate context faithfulness hallucinations is to dynamically identify and prioritize these tokens during generation. This requires detecting reliable signals that indicate how retrieved context influences the model's predictions. Attention mechanisms in Transformer models naturally emerge as a key source of such signals, since they facilitate information flow between tokens (Olsson et al., 2022; Meng et al., 2022; Geva et al., 2023). We hypothesize that attention distributions encode intrinsic indicators of context utilization.

To validate our hypothesis, we trained a probing classifier using Logistic Regression on attention distributions, achieving over 0.99 AUC in distinguishing contextually relevant tokens. Even with just 100 training samples, the classifier demonstrated strong generalization across in-domain and cross-domain test sets, indicating that attention distributions inherently encode context utilization signals. These results support attention-based context utilization as a fundamental mechanism in LLMs. By leveraging these intrinsic signals, attention distributions provide a lightweight and interpretable means to assess how models integrate retrieved context into their predictions.

Motivated by these findings, we propose Dynamic Attention-Guided Context Decoding (DAGCD), a novel method to mitigate context faithfulness hallucinations. Inspired by the copy-generator framework (See et al., 2017; Xu et al., 2020), DAGCD integrates attention weights to estimate the relevance of tokens in the retrieved context, dynamically adjusting output probabilities. Additionally, token-level uncertainty guides these adjustments by emphasizing underconfident yet contextually relevant tokens. By combining these strategies, DAGCD ensures output alignment with the retrieval context and maintains efficiency.

Our contributions are as follows:

1. **Comprehensive analysis of context faithfulness hallucinations:** We identify a strong correlation between token-level uncertainty and context faithfulness hallucinations, showing that incorrect responses exhibit higher entropy and retrieved context is often recognized but not effectively prioritized.
2. **Attention-driven interpretability framework:** We propose **Dynamic Attention-Guided Context Decoding (DAGCD)**, leveraging attention distributions to amplify contextually relevant tokens and ensure faithful utilization of retrieved context.
3. **Lightweight and efficient decoding:** DAGCD operates in a single decoding pass, integrating attention signals and uncertainty measures without additional overhead, improving efficiency.
4. **Extensive validation across datasets and models:** DAGCD outperforms greedy decoding across multiple QA datasets, improving EM by **17.67%** on pretrained models and **2.25%** on instruction-tuned models, demonstrating robustness and scalability.

2 Why Can't Generate Faithful Answers?

Token-level uncertainty is closely related to factual hallucinations, as models often exhibit higher entropy in the token-level probability distribution when generating factually incorrect outputs (Chuang et al., 2024b; Das et al., 2025). While uncertainty measures help detect hallucination-prone predictions, most studies focus on factual hallucinations, where responses are incorrect without retrieved context. In contrast, context faithfulness hallucinations arise when models rely on retrieved information but generate misaligned or contradictory outputs. Despite their significance in RAG, their relationship with uncertainty remains unclear. Inspired by prior research, we investigate whether unfaithful responses in RAG exhibit higher entropy and whether contextually relevant tokens are recognized but assigned insufficient confidence.

2.1 Uncertainty Leads to Unfaithful Answers

Experimental Setup To assess the relationship between uncertainty and response faithfulness, we use two common metrics: **Normalized Entropy (NE)**, which measures the overall uncertainty in the token-level probability distribution (Huang et al., 2023b), and **Maximum Softmax Probability (MSP)**, which quantifies the model's confidence in the highest probability within the token-level probability distribution (Hendrycks and Gimpel, 2017). Higher NE indicates greater uncertainty, while higher MSP corresponds to greater confidence in the predictions of the model. For detailed experimental descriptions, see Appendix A.

Results and Analysis Figure 1 (a1) and (a2) illustrates the Normalized Entropy and Maximum Softmax Probability of the token-level probability distribution when the model produces correct and wrong answers. The model exhibits higher uncertainty for wrong answers, with an average NE of 0.36 compared to 0.29 for correct cases, and a lower average MSP of 0.25 compared to 0.41. Notably, there is a substantial overlap between the correct (blue) and wrong (red) cases in the figure, indicating that even correct answers often exhibit high uncertainty. We also analyzed the correlation between prediction accuracy and uncertainty, and the results demonstrate a significant negative correlation, further confirming that **token-level uncertainty is strongly associated with unfaithful answers**. For detailed results see Appendix A.4.

2.2 LLM is Actually Utilizing Context

Our previous analysis links token-level uncertainty to unfaithful responses, showing that incorrect outputs often have higher entropy and lower confidence. However, this does not mean the model ignores retrieved context. A key question remains: **Do incorrect responses imply that LLMs have failed to leverage the retrieved context?**

To investigate this, we examine the ranking of gold answer tokens in the token-level probability distribution for incorrect responses. If these tokens frequently rank high but are not assigned the highest probability, it suggests the model identifies relevant context but fails to prioritize it effectively.

Experimental Setup We analyze incorrect responses from §2.1 by examining the ranking distribution of gold answer tokens in the token-level probability distribution of the first generated token after context concatenation. The ranks are grouped into intervals, and we compute the proportion of gold answer tokens within each rank interval.

Results and Analysis As shown in Figure 1 (subplot b), **when the model generates incorrect responses, 66% of cases have the gold answer token ranked within the top 10 based on the token-level probability distribution, compared to only 26% when context is absent.** Moreover, as shown in Figure 6, the average probability gap between the gold answer token and the highest-probability token remains relatively small: 0.14 for ranks between 2 and 4, and 0.24 for ranks beyond 30.

These findings indicate that **in context faithfulness hallucination scenarios, the model recognizes relevant context tokens but fails to prioritize them effectively, limiting their impact on the generated output**. This highlights an incomplete integration of retrieved context, emphasizing the need for improved context incorporation strategies to enhance faithfulness and mitigate hallucinations.

3 Context Utilization Signal in Attention

Our analysis in Section 2 highlights that while models often identify relevant context tokens, they fail to assign them sufficient confidence, leading to context faithfulness hallucinations. To better understand this phenomenon, we seek reliable signals that indicate which retrieved context tokens are effectively utilized by the model during generation.

Due to their role in integrating and propagating information across tokens, attention mechanisms naturally emerge as a key candidate for capturing such signals (Olsson et al., 2022; Meng et al., 2022; Geva et al., 2023). We hypothesize that **attention distributions encode intrinsic indicators of context utilization**, providing a lightweight and interpretable means to assess how models incorporate retrieved context into their outputs.

3.1 Attention Ratio

A key challenge in analyzing attention weights is the noise from non-context tokens (e.g., delimiters) caused by attention sink effects (Bondarenko et al., 2021; Xiao et al., 2024). Additionally, attention magnitudes vary across heads and layers, complicating feature comparison (Jain and Wallace, 2019; Vig and Belinkov, 2019). To address these issues, we introduce the **attention ratio**, a normalized measure that captures how much attention a retrieved context token receives relative to the total attention assigned within the context. For a given token j

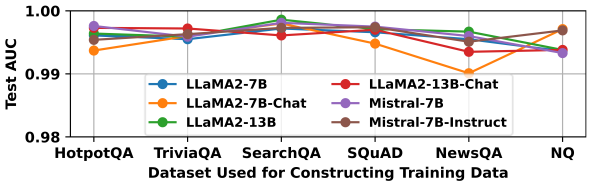

Figure 2: Cross Domain Validation. One dataset as the training set (X-axis) and the remaining datasets as the test sets, showing the AUC on the test sets (Y-axis).

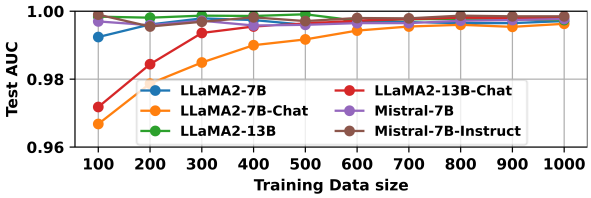

Figure 3: Training Data Size Validation. Training sets of varying sizes were constructed from a single dataset (HotpotQA), and evaluate on remain datasets.

in the retrieved context C, the attention ratio at the h-th head in the l-th layer is defined as:

$$r_{l,h}^{j} = \frac{a_{l,h}^{j}}{\sum_{j \in C} a_{l,h}^{j}} \tag{1}$$

where $a_{l,h}^{j}$ denotes the raw attention weight assigned to token j. This ratio quantifies the relative importance of j within the retrieved context for a specific attention head.

To construct token-level features, we aggregate attention ratios across all heads:

$$v_j = \left[r_{1,1}^{j}, \ldots, r_{\text{num\_layers,num\_heads}}^{j} \right] \tag{2}$$

This feature vector represents the distribution of attention across layers and heads, enabling a structured assessment of context token importance.

3.2 Experimental Setup

To validate our hypothesis and investigate whether this mechanism generalizes across different datasets, training sizes, and prompt templates, we conduct the following experiments.

Data Construction We constructed the dataset by randomly selecting samples from the MrQA training set dataset (contains six open-book QA datasets in different domains) (Fisch et al., 2019), focusing on cases where the model's output changed from incorrect to correct after context concatenation (Meng et al., 2022). These cases indicate that the model successfully leveraged the retrieved context to produce the correct answer. Context tokens were labeled as positive (utilized) if they corresponded to the gold answer, and negative (non-utilized) otherwise. Using these labels, we extracted attention ratio feature vectors $\mathbf{v}_j$ to train a Logistic Regression (LR) classifier.

3.3 Results

Cross Domain Validation We tested the classifier on six sub-datasets from different domains. Specifically, we selected one dataset as the training set and tested the performance on the remaining datasets (each sub-dataset construct 500 samples, contains 250 positive and 250 negative samples).

As shown in Figure 2, the classifier achieves an average AUC above 0.99 across all datasets and LLMs. The results indicate that the context utilization signal in attention is **data-independent**.

Training Data Size Validation Building on the cross domain experiment, we further tested the impact of training set size on classifier performance. Specifically, we trained the model using data constructed from one sub-dataset and evaluated its performance on the remaining sub-datasets.

Figure 3 shows the variation in model AUC with different training data sizes (where "Train size = 100" refers to a training set constructed with 50 positive and 50 negative samples). The results indicate that even with only 100 samples, the model's AUC exceeds 0.96, and the performance improvement becomes limited as the data size increases. This demonstrates that the LR classifier trained using the attention ratio exhibits strong **data-efficiency**.

Additional Results We then examined the impact of different prompt templates, and the results indicate that the classifier consistently maintains high performance regardless of the prompt template used. For Details, see Appendix B.

To better apply the classifier to practical tasks, we conducted a detailed analysis of the importance of different features and the relationships between them. The results show that **the classifier using the top-K features outperforms the one using the full feature set**. Furthermore, the attention heads exhibit Concentration and Complementarity features. For Details, see Appendix C.

Conclusion: A Fundamental Mechanism in Transformer-based LLMs The consistent generalization of attention-based context utilization across datasets, data sizes, and prompts reinforces its role as a fundamental mechanism in LLMs. Our findings show that attention heads encode robust context integration signals, providing a lightweight

and interpretable way to assess how models incorporate retrieved context.

4 Method

Inspired by the copy-generation mechanism (See et al., 2017; Xu et al., 2020), we propose **Dynamic Attention-Guided Context Decoding (DAGCD)** to mitigate context faithfulness hallucinations. DAGCD leverages utilization signals to dynamically guide the generation process, focusing on relevant contextual tokens. It integrates three steps: detecting utilized context tokens during inference (§4.1), constructing a utilization distribution (§4.2), and adjusting the token-level probability distribution to enhance contextual utilization (§4.3).

4.1 Context Utilization Detection

Context Utilization Detector To identify contextually relevant tokens during inference, we utilize a Logistic Regression (LR) classifier trained on attention-based utilization signals. Our analysis in Section 3.3 shows that selecting the most informative attention heads improves generalization. Thus, we construct the Context Utilization Detector based on the top-K most important attention heads.

Feature Data Collection To obtain feature vectors, we extract attention distributions at the current decoding step, as illustrated in Figure 4. Specifically, we take the last row of the attention map for each selected head $h_k \in H$ (H is the set of top-K attention heads). To ensure focus on relevant information, we filter out non-contextual tokens, such as query tokens and placeholder tokens in templates.

The top-K feature vector for each contextual token j is then constructed as:

$$\mathbf{v}_j^{(K)} = [r_{h_1}^j, r_{h_2}^j, \ldots, r_{h_K}^j] \tag{3}$$

where $r_{h_k}^j$ represents the normalized attention ratio of token j in attention head h_k.

Finally, the feature vector $\mathbf{v}_j^{(K)}$ is fed into the detector, which identifies the set of context tokens actively utilized at the current decoding step.

4.2 Utilization Distribution Construction

The context utilization detector identifies which tokens are utilized but does not quantify the degree of utilization for each token. To address this, we compute a utilization score s_j for each token j by aggregating attention ratios from selected attention heads, weighted by their normalized feature coefficients w_k. Tokens classified as unused by the detector are directly assigned a score of zero.

$$s_j = \sum_{k=1}^{K} (r_{h_k}^j \times w_k),\ w_k = \frac{c_k}{\sum_{k=1}^{K} c_k} \tag{4}$$

where $r_{h_k}^j$ is the normalized attention ratio of token j in attention head h_k, and w_k is the importance weight assigned to head h_k. The coefficient c_k is learned from the LR classifier, representing the contribution of each head to context utilization.

The utilization distribution U represents a probability distribution over context tokens, normalized based on their utilization scores:

$$\mathbf{U} = [u_1, u_2, \ldots, u_N], \quad u_i = \frac{s_i}{\sum_{j=1}^{N} s_j} \tag{5}$$

where u_i denotes the utilization probability of token i, N is the vocabulary size. Tokens either absent from the context or classified as non-utilized by the detector ($s_i = 0$) are assigned $u_i = 0$.

Top-Rank Constraint To enhance the reliability of generation adjustments, our approach applies a top-rank restriction, ensuring modifications focus on plausible tokens. Specifically, we define U_{top} as the subset of the utilization distribution U corresponding to tokens ranked within the top-R positions of the generation distribution.

This design builds on prior work (Li et al., 2023; Chuang et al., 2024b), addressing context faithfulness hallucination challenges while leveraging our observation that correct context tokens usually appear within the top-ranked positions in the token-level probability distribution, even when the model generates incorrect answers. Through this constraint, we reduce the risk of amplifying irrelevant or nonsensical tokens, preserving output integrity.

4.3 Generating More Faithful Answers

DAGCD adjusts token probabilities based on token-level uncertainty, measured using the normalized entropy $H_{\text{norm}}(P)$ of the token-level probability distribution. High entropy indicates greater uncertainty and a higher risk of generating contextually inconsistent responses. Since entropy correlates with uncertainty but lacks a fixed numerical relationship, we introduce a scaling factor α to compensate for model-specific entropy variations.

Adjustments are applied only when utilized tokens in U_{top} overlap with the top-ranked tokens in the token-level probability distribution. The adjusted generation distribution P' is computed as:

$$P' = P + \alpha H_{\text{norm}}(P) \cdot U_{\text{top}} \tag{6}$$

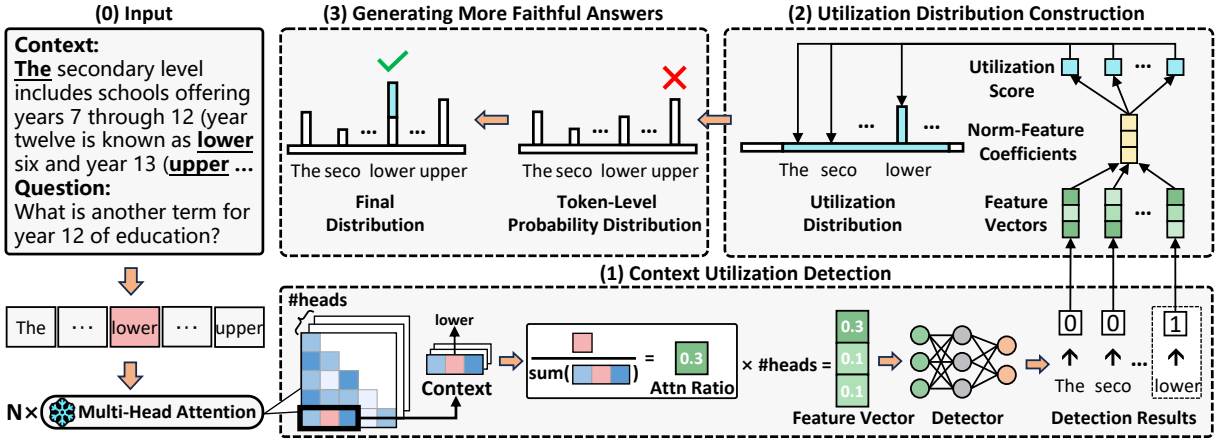

Figure 4: The illustration of the generation process of our proposed DAGCD method.

where P represents the original token-level probability distribution, and $\alpha H_{\text{norm}}(P)$ dynamically scales the adjustment based on model uncertainty.

5 Experiments

5.1 Experimental Setup

Datasets We evaluate context faithfulness on Open-Book Question-Answering (QA) datasets, where each question is paired with external context containing the correct answer. This setup ensures that only context-grounded answers are considered valid, allowing for the assessment of whether the model generates context-faithful hallucinations based on answer accuracy. Specifically, We conducted experiments on seven open-book QA datasets. For further details, refer to Appendix D.1.

Implementation Details **We used 100 samples constructed from a single dataset (HotpotQA) as the training set to train the Context Utilization Detector.** The scaling factor α is set to 2 for pre-trained models and 4 for instruction-tuned models to account for entropy variations. The logistic regression classifier and utilization distribution are computed using the top-10 attention heads, with adjustments restricted to the top-10 ranked tokens (U_{top}). Further detailed settings see Appendix D.2.

Metrics Consistent with prior work(Jin et al., 2024a; Yuan et al., 2024; Wang et al., 2024), we use EM and F1 score metrics to evaluate the performance of the models on open-book QA datasets.

The LLMs used in our experiments and the baselines are detailed in Appendix D.3 and D.4.

5.2 Model Performance Comparison

Table 1 shows DAGCD's effectiveness across diverse QA datasets and models. **We also tested DAGCD on summarization tasks, yielding improvements**. For details see Appendix D.5.

5.2.1 Dataset-Level Observations

DAGCD achieves consistent improvements across diverse QA tasks, including multi-hop reasoning, long-form retrieval, and document-level QA.

HotpotQA, TriviaQA, SearchQA: DAGCD excels in multi-paragraph reasoning and long-form retrieval tasks. It achieves the highest gains on HotpotQA with a **18.80%** EM and **14.81%** F1 improvement on Mistral-7B. DAGCD also outperforms baselines on TriviaQA and SearchQA, showing significant improvements across models.

SQuAD, NewsQA, NQ: DAGCD demonstrates robust performance in single-paragraph and document-level tasks. On NQ, it achieves a **71.46%** EM and **39.10%** F1 improvement on Mistral-7B over greedy decoding, while delivering consistent gains across SQuAD and NewsQA datasets.

NQ-Swap: In adversarial scenarios simulated by NQ-Swap, DAGCD shows notable improvements, including **74.52%** EM and **51.74%** F1 gains on Mistral-7B, highlighting its robustness.

5.2.2 Model-Level Observations

DAGCD demonstrates broad applicability across different model families, sizes, and tuning variants.

Model Families: DAGCD enhances performance across LLaMA and Mistral families. On Mistral-7B, DAGCD improves EM by **27.86%** on Mistral-7B and **14.33%** on LLaMA2-7B compared to greedy decoding.

Model Sizes: DAGCD improves performance across models of varying model sizes, achieving

Dataset	Decoding	HotpotQA		TriviaQA		SearchQA		SQuAD		NewsQA		NQ		NQ-swap		Average	
		EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
LLaMA2-7B	Greedy	44.74	54.71	55.28	68.01	54.21	59.24	39.90	52.61	32.93	45.46	38.85	50.59	36.03	36.62	43.13	52.46
	CAD	44.13	54.49	55.26	68.04	54.14	59.21	38.35	51.12	31.74	43.70	38.14	48.58	36.11	36.69	42.55	51.69
	COIECD	42.03	51.48	57.04	70.06	**57.03**	**63.14**	40.93	54.78	34.40	48.48	38.79	51.69	34.98	35.63	43.60	53.61
	OURs	**47.35**	**57.43**	**58.12**	**70.97**	54.35	59.70	**48.02**	**60.02**	**36.51**	**49.06**	**47.74**	**60.23**	**53.12**	**53.63**	**49.32**	**58.72**
LLaMA2-7B -Chat	Greedy	53.33	67.41	71.72	76.83	54.19	58.11	67.69	78.70	39.41	53.90	50.47	65.48	67.98	68.78	57.83	67.03
	CAD	52.86	67.16	71.70	76.83	54.16	58.11	65.89	77.91	38.46	53.26	48.89	65.00	68.04	68.85	57.14	66.73
	COIECD	53.14	67.03	**72.26**	**77.33**	**55.04**	**58.98**	68.32	79.56	40.00	54.55	**52.39**	66.84	69.48	70.13	**58.66**	67.77
	OURs	**55.31**	**68.61**	69.21	75.95	54.25	58.13	**68.49**	**79.76**	**40.53**	**54.81**	51.69	**66.92**	**69.50**	**70.30**	58.43	**67.78**
LLaMA2-13B	Greedy	52.36	63.40	58.25	69.95	63.22	68.33	51.64	64.57	30.84	40.11	42.26	54.08	49.02	49.59	49.66	58.58
	CAD	51.53	63.11	58.25	69.92	63.13	68.32	49.94	63.44	29.94	39.04	41.40	52.27	49.07	49.63	49.04	57.96
	COIECD	50.21	60.96	59.19	71.49	**65.97**	**71.00**	52.73	65.93	**35.66**	**50.58**	42.35	54.37	48.35	48.84	50.64	60.45
	OURs	**53.13**	**64.21**	**59.65**	**72.07**	61.41	67.08	**56.54**	**68.56**	33.26	42.20	**55.34**	**71.23**	**65.86**	**66.19**	**55.03**	**64.51**
LLaMA2-13B -Chat	Greedy	55.01	69.92	**74.58**	79.35	67.08	71.96	68.26	79.45	40.20	55.11	53.49	69.18	60.69	61.77	59.90	69.53
	CAD	54.44	69.66	**74.58**	**79.37**	67.01	71.95	66.77	78.70	39.44	54.44	52.89	68.63	60.83	61.92	59.42	69.24
	COIECD	56.15	70.43	73.87	78.96	67.28	71.93	68.49	80.39	40.75	56.16	53.69	69.81	62.47	63.21	60.39	70.13
	OURs	**57.76**	**71.69**	73.04	78.77	**68.19**	**72.73**	**69.66**	**80.76**	**40.78**	**56.24**	**55.36**	**71.31**	**64.03**	**65.20**	**61.26**	**70.96**
Mistral-7B	Greedy	53.41	64.36	**59.45**	68.39	63.79	67.77	44.19	56.11	31.51	38.94	33.74	51.18	39.80	46.04	46.56	56.11
	CAD	41.57	56.01	57.88	67.48	63.64	68.65	34.08	47.55	25.78	35.37	23.18	41.46	26.96	35.97	39.01	50.36
	COIECD	46.43	58.32	44.30	51.77	54.82	59.17	50.50	60.98	40.05	52.78	42.12	56.58	59.53	61.89	48.25	57.36
	OURs	**63.45**	**73.89**	56.76	**71.86**	**64.49**	**69.88**	**63.04**	**73.75**	**41.62**	**55.13**	**57.85**	**71.19**	**69.46**	**69.86**	**59.52**	**69.37**
Mistral-7B -Instruct	Greedy	58.70	72.18	69.64	75.61	44.42	49.63	67.28	79.37	39.79	54.72	52.29	66.93	66.90	67.83	57.00	66.61
	CAD	49.30	64.81	**70.23**	75.95	45.42	50.96	59.97	72.92	34.97	51.90	42.63	58.49	52.04	53.75	50.65	61.25
	COIECD	59.74	72.59	64.92	72.15	37.09	42.66	**68.45**	**81.03**	40.84	55.96	53.54	68.72	**72.81**	**73.81**	56.77	66.70
	OURs	**60.55**	**73.49**	69.70	**75.96**	**47.17**	**52.65**	68.30	80.62	**40.85**	**56.23**	**54.78**	**69.72**	71.38	72.12	**58.96**	**68.68**

Table 1: Performance comparison of different decoding methods. All baselines are reproduced under the same settings. **Bold** indicates the best performance, and underlined indicates the second-best performance.

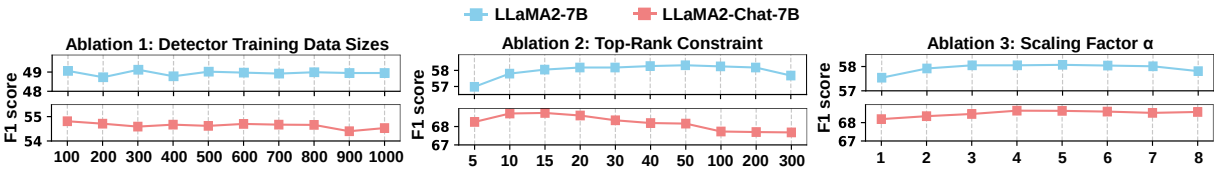

Figure 5: Ablation Study for DAGCD. **Left Part**: Ablation 1 Detector Training Data Sizes, **Center Part**: Ablation 2 Top-Rank Constraint, **Right Part**: Ablation 3 Scaling Factor α.

an average EM and F1 increase of **14.33%** and **11.93%** on LLaMA2-7B, and **10.82%** and **10.12%** on LLaMA2-13B, respectively.

Instruction-Tuned Models: Instruction-tuned models, after fine-tuning, show reduced uncertainty in generation probabilities, limiting our method's improvement margin. However, DAGCD still surpasses all baselines with the highest performance.

5.3 Ablation Study

We conducted three ablation experiments to evaluate the impact of variations in different modules on performance. Specifically, in each ablation experiment, the ablation module is adjusted while the other two modules are kept at their default settings.

Ablation 1: Detector Training Data Sizes We tested the impact of detectors trained on different data sizes on actual inference performance. The results, shown in Figure 5 (left), demonstrate that our method consistently maintains strong performance across various training data sizes, achieving notable results even with just 100 training samples.

Ablation 2: Top-Rank Constraint We evaluated various top-rank constraints on HotpotQA. Figure 5 (center), top-rank filtering reduces false positives, with F1 score initially improving as constraints loosen, then declining when overly relaxed.

Ablation 3: Scaling Factor α We evaluated the impact of different scaling factor α on model performance. The results, presented in Figure 5 (right), indicate that α determines the adjustment intensity applied to the original generation distribution. For pretrained models, optimal performance is achieved at $\alpha = 2$, whereas for Chat models, the best performance is observed at $\alpha = 4$.

Additional results and performance variations under different prompt templates see Appendix E.

6 Discussion and Analysis

6.1 Dynamic Decoding: Real-Time Efficiency Without Post-Generation Correction

Post-generation correction methods, such as CAD (Shi et al., 2024b) and COIECD (Yuan et al., 2024), improve contextual alignment but rely on multi-step processes, causing significant computational overhead. In contrast, DAGCD incorporates context adjustments during generation, ensuring

both efficiency and real-time optimization.

Lightweight Context Utilization Detector Using a logistic regression-based Context Utilization Detector, our method enables real-time adjustments with minimal computational cost. This detector is more efficient than resource-heavy methods like integrated gradients or attention head manipulation.

Single-Pass Decoding with Real-Time Faithfulness Optimization By integrating the Context Utilization Detector directly into the decoding process, our method removes redundant steps like output comparisons or external consistency checks. During generation, attention-based context utilization signals are leveraged in real time to proactively enhance faithfulness. This single-step strategy ensures that the output aligns with the input context without extra post-processing, while maintaining the theoretical time complexity of greedy decoding.

6.2 Interpretability Through Attention: Insights into Context Utilization

By systematically analyzing attention mechanisms, our approach uncovers how retrieved context influences the generation process and provides interpretable insights into the behavior of the model.

Feature-Based Attention Analysis Using natural cases, such as failure in closed-book settings but success in open-book settings, we isolate attention patterns that are indicative of context utilization. A logistic regression classifier trained on these patterns identifies the relevant attention heads with high accuracy, quantifying their contributions.

Transparent Decision-Making The feature coefficients of the logistic regression model directly map to the importance of specific attention heads. This transparency allows for intuitive interpretation, clarifying which heads are most responsible for leveraging context tokens during generation.

7 Related Work

7.1 Context Faithfulness Hallucination

Current solutions to context faithfulness hallucination primarily focus on detection and mitigation. For detection, Lei et al. (Lei et al., 2023) proposed a post-generation editing strategy using natural language inference to classify and revise hallucinated segments. Choi et al. (Choi et al., 2023) introduced Knowledge-Constrained Decoding, detecting hallucinations during generation and reweighting token distributions to guide output. Chuang et al. (Chuang et al., 2024a) proposed Lookback Lens Guided Decoding, selecting the most faithful output among candidates to improve consistency.

For mitigation, CAD (Shi et al., 2024b) compares outputs with and without concatenated context to enhance contextual adherence, while COIECD (Yuan et al., 2024) improves upon CAD by incorporating entropy-based constraints to balance context usage. Wang et al. (Wang et al., 2024) further introduced ADACAD, which dynamically adjusts token-level adherence using divergence between contextual and non-contextual outputs.

While effective, most existing methods face limitations such as high computational overhead and reliance on multiple decoding passes. In this work, we propose a real-time solution that integrates context utilization signals directly into the decoding process, achieving efficient and faithful generation, without the need for additional processing steps.

7.2 Attention and Interpretability

The attention mechanism provides valuable insights into how models prioritize different parts of an input sequence (Clark et al., 2019; Geva et al., 2023) and has become central to understanding Transformer-based LLMs (Vashishth et al., 2019; Hao et al., 2021; Zhao et al., 2024). In LLMs, attention heads often perform distinct roles, such as capturing syntactic dependencies or aligning semantic relationships (Olsson et al., 2022; Zheng et al., 2024; Jin et al., 2024b).

Recent studies have also explored the collaborative behavior of attention heads. For instance, the retrieval head framework (Wu et al., 2024) identifies heads that collectively retrieve relevant tokens, while cutting-off-heads (Jin et al., 2024b) highlights critical heads through systematic ablation. Gradient-based methods like IRCAN (Shi et al., 2024a) further investigate the contributions of attention scores and neurons to model outputs.

Unlike previous work, which focused on individual heads, our study examines the collaborative patterns of multiple attention heads. By analyzing how attention mechanisms collectively utilize contextual tokens, we provide a holistic view of their role in aligning outputs with user-provided context.

8 Conclusion

In this paper, we mitigate context faithfulness hallucinations in LLMs by proposing Dynamic Attention-Guided Context Decoding, a lightweight framework that integrates attention distributions and entropy-based uncertainty signals to amplify contextually relevant tokens during generation. Our analysis revealed a strong correlation between high

uncertainty and hallucinations, and probing experiments validated that attention mechanisms encode signals indicative of contextual utilization, and further demonstrated that this signal is a fundamental mechanism in Transformer-based LLMs. Experiments across multiple open-book QA datasets demonstrated that DAGCD achieves consistent improvements in context faithfulness, robustness, and scalability, providing an effective solution for context-sensitive generation tasks.

Limitations

Dependency on Classifier Accuracy and Robustness to Noisy Contexts DAGCD relies on an attention-ratio based classifier to assess the relevance of context tokens during generation. While the classifier demonstrates high accuracy across datasets and models, its performance may degrade in scenarios with extremely long contexts, complex dialogues, or noisy inputs. Misclassifications in these cases could lead to incorrect adjustments, potentially amplifying irrelevant tokens or diminishing the contribution of critical ones. Similarly, the method's robustness to adversarial or noisy contexts with misleading or irrelevant information remains an open challenge. Enhancing the classifier's resilience and incorporating mechanisms to filter or downweight adversarial noise could further strengthen DAGCD's applicability in real-world scenarios.

Scaling Factor Adjustment for Model Characteristics The scaling factor α introduced in DAGCD needs to be adjusted based on the characteristics of different models. Although our study shows a strong correlation between entropy-guided uncertainty measures and the model's uncertainty during generation, it does not establish a precise quantitative relationship. This limitation necessitates empirical calibration of the scaling factor for each model to ensure effective adjustments. Such calibration ensures that the method compensates for model-specific entropy variations, but it may introduce additional computational overhead during deployment.

Generalization Across Tasks and Domains Our evaluation primarily focuses on QA tasks, leaving the generalization of DAGCD to other tasks, such as summarization or dialogue generation, unexplored. The attention-ratio based classifier, optimized for QA datasets, may require additional fine-tuning or redesign to handle different output structures and task-specific challenges. Extending the method to diverse domains and tasks could further validate its robustness and scalability.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *ArXiv*.

Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. 2021. Understanding and overcoming the challenges of efficient transformer quantization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7947–7969, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Sehyun Choi, Tianqing Fang, Zhaowei Wang, and Yangqiu Song. 2023. KCTS: Knowledge-constrained tree search decoding with token-level hallucination detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14035–14053, Singapore. Association for Computational Linguistics.

Yung-Sung Chuang, Linlu Qiu, Cheng-Yu Hsieh, Ranjay Krishna, Yoon Kim, and James R. Glass. 2024a. Lookback lens: Detecting and mitigating contextual hallucinations in large language models using only attention maps. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1419–1436, Miami, Florida, USA. Association for Computational Linguistics.

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

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.

Souvik Das, Lifeng Jin, Linfeng Song, Haitao Mi, Baolin Peng, and Dong Yu. 2025. Entropy guided extrapolative decoding to improve factuality in large

language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 6589–6600.

Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*.

Wenqi Fan, Yujuan Ding, Liang bo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Knowledge Discovery and Data Mining*.

Shangbin Feng, Vidhisha Balachandran, Yuyang Bai, and Yulia Tsvetkov. 2023. FactKB: Generalizable factuality evaluation using language models enhanced with factual knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 933–952, Singapore. Association for Computational Linguistics.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13, Hong Kong, China. Association for Computational Linguistics.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *ArXiv*, abs/2312.10997.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, Singapore. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Yaru Hao, Li Dong, Furu Wei, and Ke Xu. 2021. Self-attention attribution: Interpreting information interactions inside transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12963–12971.

Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023a. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*.

Yuheng Huang, Jiayang Song, Zhijie Wang, Huaming Chen, and Lei Ma. 2023b. Look before you leap: An exploratory study of uncertainty measurement for large language models. *ArXiv*, abs/2307.10236.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not explanation. 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 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

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

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

Zhuoran Jin, Pengfei Cao, Hongbang Yuan, Yubo Chen, Jiexin Xu, Huaijun Li, Xiaojian Jiang, Kang Liu, and Jun Zhao. 2024b. Cutting off the head ends the conflict: A mechanism for interpreting and mitigating knowledge conflicts in language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 1193–1215, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

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.

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*, 7:452–466.

Deren Lei, Yaxi Li, Mengya Hu, Mingyu Wang, and Xi Yun. 2023. Chain of natural language inference

for reducing large language model hallucinations. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

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.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.

Xun Liang, Shichao Song, Zifan Zheng, Hanyu Wang, Qingchen Yu, Xunkai Li, Rong-Hua Li, Yi Wang, Zhonghao Wang, Feiyu Xiong, et al. 2024. Internal consistency and self-feedback in large language models: A survey. *arXiv preprint arXiv:2407.14507*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. 2022. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*.

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.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Dan Shi, Renren Jin, Tianhao Shen, Weilong Dong, Xinwei Wu, and Deyi Xiong. 2024a. 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*.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. 2024b. Trusting your evidence: Hallucinate less with context-aware decoding. In *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. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Shikhar Vashishth, Shyam Upadhyay, Gaurav Singh Tomar, and Manaal Faruqui. 2019. Attention interpretability across nlp tasks. *arXiv preprint arXiv:1909.11218*.

Jesse Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 63–76, Florence, Italy. Association for Computational Linguistics.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. 2023. Freshllms: Refreshing large language models with search engine augmentation. In *Annual Meeting of the Association for Computational Linguistics*.

Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2024. Adacad: Adaptively decoding to balance conflicts between contextual and parametric knowledge. *arXiv preprint arXiv:2409.07394*.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. 2024. Retrieval head mechanistically explains long-context factuality. *arXiv preprint arXiv:2404.15574*.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2024. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*.

Song Xu, Haoran Li, Peng Yuan, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2020. Self-attention guided copy mechanism for abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1355–1362, Online. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Xiaowei Yuan, Zhao Yang, Yequan Wang, Shengping Liu, Jun Zhao, and Kang Liu. 2024. Discerning and resolving knowledge conflicts through adaptive decoding with contextual information-entropy constraint. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3903–3922, Bangkok, Thailand. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38.

Zifan Zheng, Yezhaohui Wang, Yuxin Huang, Shichao Song, Bo Tang, Feiyu Xiong, and Zhiyu Li. 2024. Attention heads of large language models: A survey. *arXiv preprint arXiv:2409.03752*.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14544–14556, Singapore. Association for Computational Linguistics.

A Details of Experiment "2.1 Uncertainty Leads to Unfaithful Answers"

A.1 Dataset Used for Analysis

We randomly sampled 6,000 instances from the MrQA training set (Fisch et al., 2019), which consolidates six open-domain question answering datasets. Specifically, we selected 500 correctly answered and 500 incorrectly answered instances from each sub-dataset. An answer was considered correct if it achieved an Exact Match (EM) with the reference answer.

A.2 Model Used for Analysis, Prompts, and Decoding Method

We conduct our analysis using LLaMA2-7B (Touvron et al., 2023) as the target model. The answers are generated using the following prompt:

`"Given the following information: {context} Answer the following question based on the given information with one or a few words: {question} Answer:"`

To ensure deterministic outputs, we employ greedy decoding.

A.3 Computation Process

For each sample, we calculated two metrics to quantify uncertainty and confidence. First, the **Normalized Entropy (NE)** measures the dispersion of probabilities across the vocabulary, providing an overall view of the model's uncertainty. And NE is defined as:

$$H_{\text{norm}}(P) = -\frac{\sum_{i=1}^{N} P_i \log P_i}{\log N}, \qquad (7)$$

where P represents the token-level probability distribution, and N denotes the vocabulary size.

Second, the **Maximum Softmax Probability (MSP)** represents the likelihood of the most probable token, offering a complementary perspective on the model's prediction confidence. These metrics focus on the initial generated token distribution to analyze how uncertainty affects response faithfulness.

A.4 Spearman Correlation Analysis

To further investigate the relationship between uncertainty and model errors (i.e., the inability to faithfully respond to the input context), we conducted a Spearman correlation analysis. Specifically, we used the normalized entropy of the token level probability distribution to measure the model's token-level uncertainty, and evaluated answer accuracy using the F1 score. We then analyzed the correlation between answer accuracy and uncertainty during answer generation, with results presented in Table 2. Our findings reveal a significant negative correlation, which becomes more pronounced after concatenating the context—i.e., higher uncertainty corresponds to lower answer accuracy (in open-book QA tasks, lower accuracy indicates that the model's response deviates from the provided context). This analysis further suggests that **token-level uncertainty is strongly associated with unfaithful answers.**

Model	w/o context	w/ context
LLaMA2-7B	-0.43	-0.53
LLaMA2-7B-Chat	-0.27	-0.33
LLaMA2-13B	-0.30	-0.51
LLaMA2-13B-Chat	-0.26	-0.32
Mistral-7B	-0.23	-0.22
Mistral-7B-Instruct	-0.30	-0.33

Table 2: Spearman correlation analysis. We examine the relationship between F1 scores (pred-ans vs. gold-ans) and norm-entropy (generation distribution) under both w/o and w/ context settings ($p \ll 0.05$ for all models).

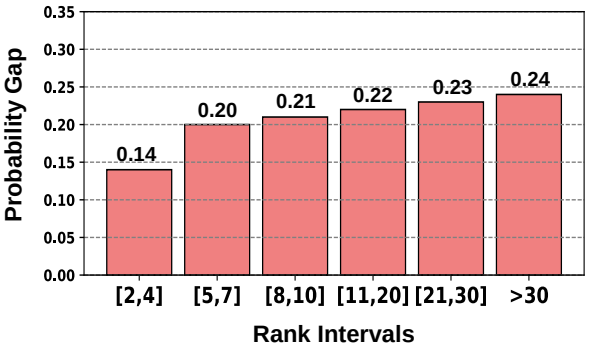

Figure 6: Probability Gap. For incorrect responses, the probability gap between the gold answer token and the highest-probability token.

A.5 Probability Gap Between The Golden Answer Token and The Ranked Top-1 Token

As shown in Figure 6, we analyze the wrong answer samples from A.1 by calculating the average probability gap between the gold answer token and the highest-probability token.

B Details of "3 Context Utilization Signal in Attention"

B.1 LR Classifier Training and Evaluation

We trained a Logistic Regression (LR) classifier using 5-fold cross-validation. L2 regularization was

Prompt	LLaMA2-7B		Mistral-7B	
	ACC	AUC	ACC	AUC
Prompt2	0.9797	0.9932	0.9762	0.9902
Prompt3	0.9768	0.9926	0.9763	0.9889
Prompt4	0.9794	0.9946	0.9771	0.9927

Table 3: Performance testing under different prompts. Training data: attention ratio feature vector under Prompt 1. Test data: attention ratio feature vector under Prompts 2, 3, and 4.

applied to prevent overfitting. The classifier was evaluated across Transformer-based LLMs, including:

- LLaMA2: 7B, 13B, 7B-Chat, 13B-Chat
- Mistral: 7B, 7B-Instruct

B.2 Cross Prompt Templates Testing

To evaluate the robustness of the classifier under different prompts, we reconstructed the attention ratio feature vectors using Prompts 2, 3, and 4. These prompts differ in structure and phrasing but are consistent in task objectives. The classifier, trained using Prompt 1, was then tested on these alternative prompts. (Templates shown in Figure 15)

The results, shown in Table 3, demonstrate that the classifier maintains an ACC exceeding 97% and AUC above 0.99 across all prompts. This indicates that the attention ratio signal is prompt-agnostic and generalizes well across different input structures.

C Analyzing Context Utilization Signal Contribution Across Heads

To further explore the role of attention heads in encoding contextual utilization signals, we conducted a detailed analysis to examine the strength and distribution of signals across heads. This section presents insights into the concentration of strong signals in certain heads, the independent utility of individual heads, and the complementarity of weaker heads.

C.1 Analysis of the Importance of Different Features

In analyzing the LR classifier trained with attention values from all attention heads as features, we found that most feature coefficients had small absolute values. This indicates that only a few attention heads are crucial for utilization detection.

To explore their impact, we selected the top-K and bottom-K features based on the absolute values of their coefficients and trained LR models using these subsets. Figure 7: subplots (a1) and (a2) shows how classification accuracy (ACC) changes with K. Using top-K features, the model achieves over 0.95 accuracy for all LLMs with K=10, matching the performance of using all features. In contrast, models with bottom-K features perform poorly, failing to reach 0.95 even with K=100. The AUC curves in Figure 7: subplots (b1) and (b2) for different K further confirm this. Models with top-K features maintain high accuracy and robustness, while those with bottom-K features show significantly worse performance. These results emphasize that a small number of key attention heads are enough for effective detection, while irrelevant features add little value and may introduce noise.

We also compared the performance of LR classifiers trained with all features versus only the top-10 features on out-of-domain data. As shown in Figure 8, the LR trained with only the top-10 features achieved better ACC and AUC on out-of-domain data.

Based on the ACC and AUC results, we find that **the LR classifier trained with the Top-10 features achieves good accuracy and robustness while using the minimum number of features**. Therefore, all subsequent inference employs LR classifiers trained with the Top-10 features.

C.2 Signal Strength and Concentration in Heads

To identify influential heads, we visualized the coefficients of the trained Logistic Regression (LR) classifier, which were derived from the attention ratio features of all heads. Approximately 5% of the heads exhibited significantly high coefficients, suggesting that these heads dominate the classification task (Figure 9). Repeating this analysis across 100 random seeds revealed consistent selection of these top heads, indicating their robustness as key signal carriers.

To evaluate the standalone utility of these heads, we trained LR classifiers using the attention ratio from a single head as the feature. The results in Figure 10 show that About 5% to 10% of the heads achieved classification accuracies (ACC) above 0.8, highlighting their ability to independently encode contextual utilization signals. However, the majority of heads performed poorly in isolation, with

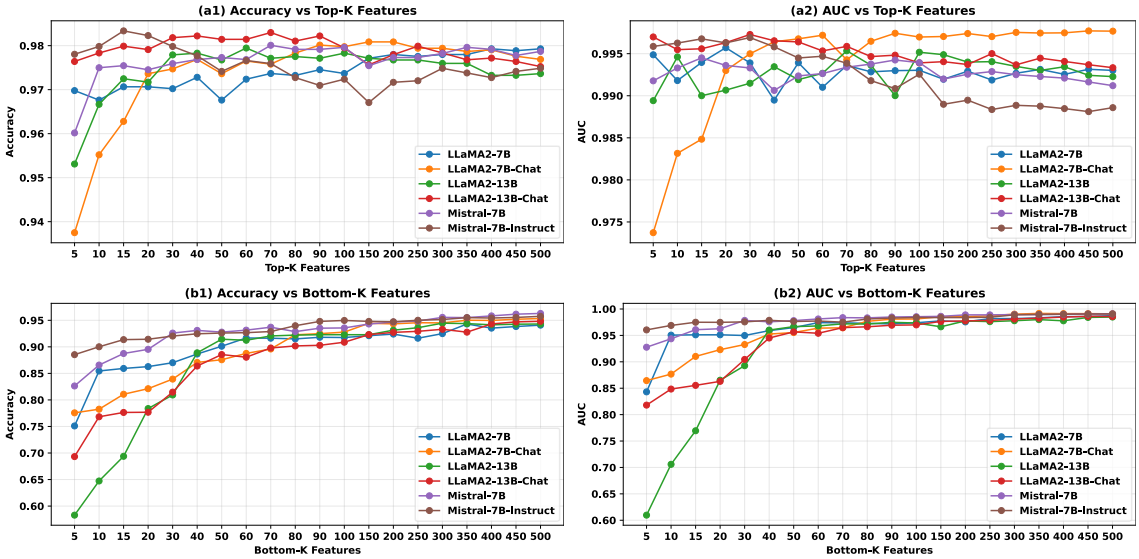

Figure 7: Performance of the LR classifier with Top-K and Bottom-K features. Based on the absolute values of feature coefficients, the Top-K and Bottom-K features were selected to train an LR classifier with sparse features. The figure shows the ACC and AUC performance of the classifier on the test set for different values of K.

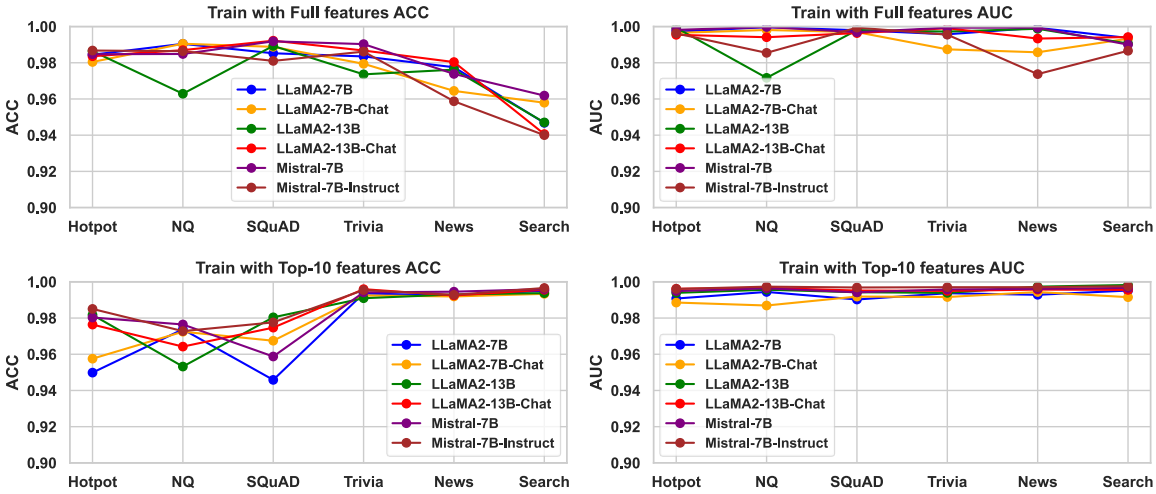

Figure 8: Out-of-domain performance of the LR classifier trained with full features and Top-10 features. Performance variations on out-of-domain data for LR classifiers trained using all features versus the top-10 features.

ACCs below 0.8. This disparity emphasizes the varying degrees of utility across heads, with a small subset contributing disproportionately strong signals.

Additionally, we also observed that on the Mistral model, the vast majority of heads perform well when acting individually. This indicates **the presence of more high-performing heads in the Mistral model, which may explain why our method achieves greater improvements on Mistral compared to other models**.

C.3 Complementary Contributions of Weaker Heads

Although most attention heads have limited standalone utility, we observe that combining weaker heads into subsets significantly improves classification performance. Figure 11 illustrates that we selected the bottom-K features, based on the classification accuracy of individual heads, to train the classifier and analyze the performance gain from combining weaker heads. The results show that when the number of bottom-K heads reaches 500, the classification accuracy stabilizes at approximately 90%. This finding highlights the complementarity of weaker heads, as their aggregated signals collectively achieve robust token classification.

Summary of Findings Our analysis reveals three key characteristics of attention heads in encoding

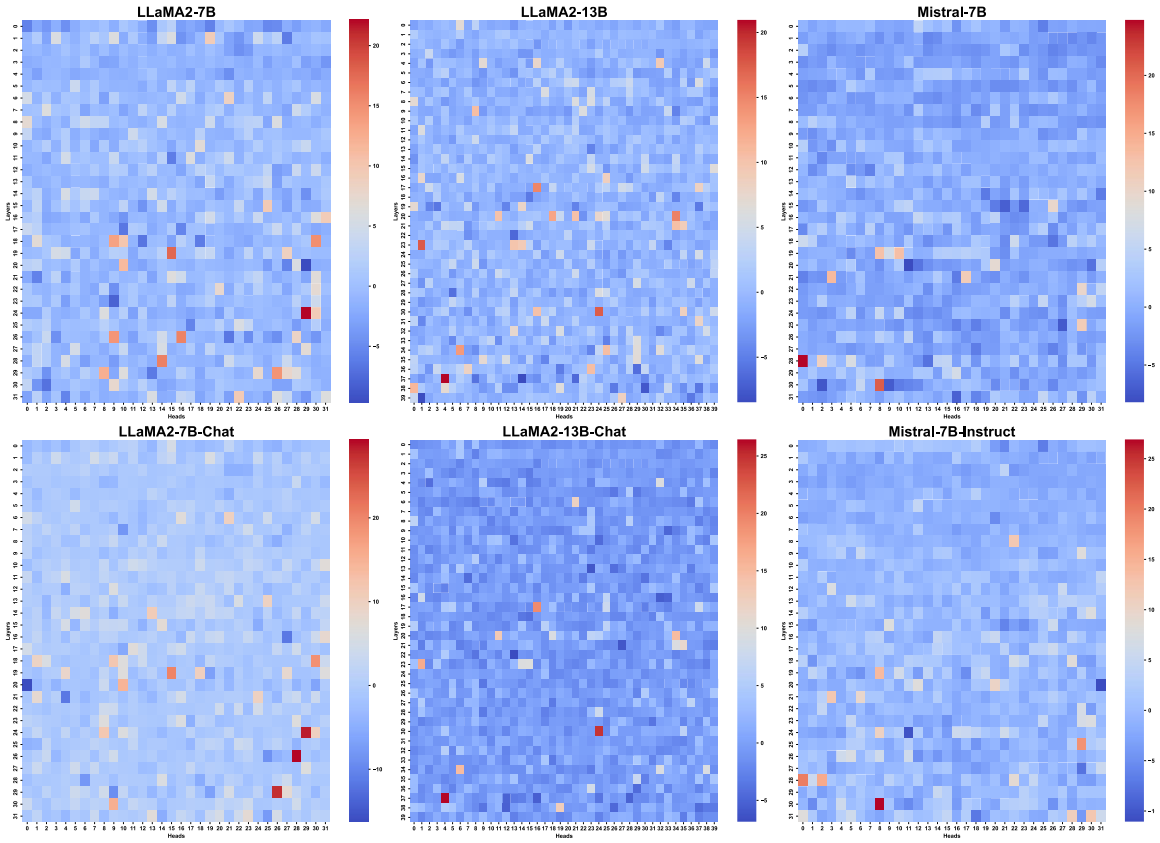

Figure 9: Heatmap of feature coefficients for LR. The heatmap of feature coefficients for LR classifiers trained using attention ratios from different LLMs as features.

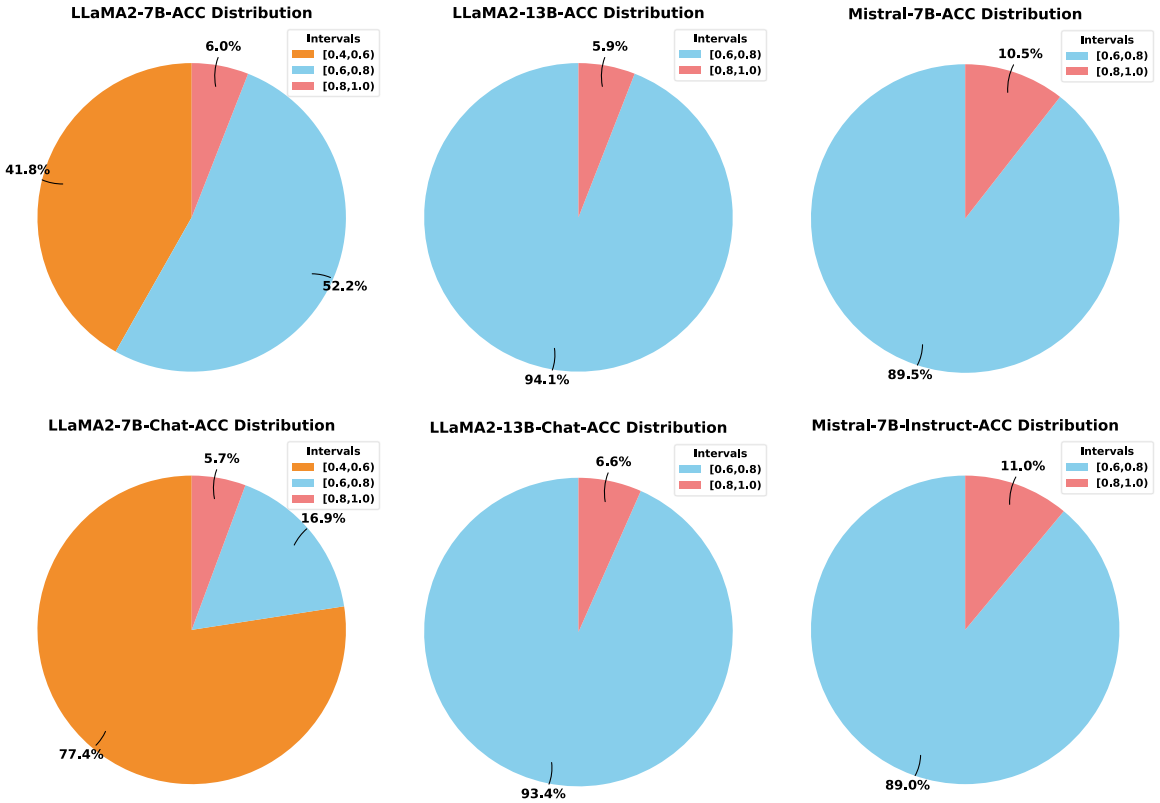

Figure 10: Performance distribution of LR with single features. An LR classifier is trained using the attention ratio from a single head as features. The figure shows the ACC distribution of LR classifiers trained with attention ratios from different heads on the test set.

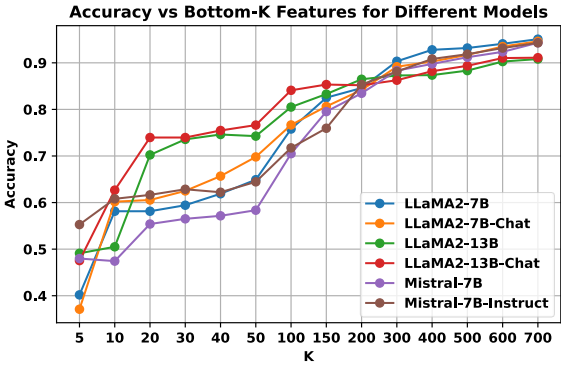

Figure 11: Performance of the LR classifier trained with Bottom-K features. Based on the ACC of LR classifiers trained using the attention ratio from a single head as features, the Bottom-K heads with the lowest ACC are selected. The LR classifier is then retrained using the attention ratios of these Bottom-K heads as features, and its performance on the test set is presented.

contextual utilization signals:

1. **Concentration:** A small subset of heads consistently contributes strong independent signals, dominating the classification task.

2. **Complementarity:** Weaker heads collectively provide complementary signals, enabling robust classification when aggregated.

These findings highlight the nuanced roles of attention heads in contextual token utilization and provide a foundation for further exploration of their properties and applications.

D Experimental Details

D.1 Dataset Details

We conducted experiments on seven open-book question-answering (QA) datasets, representing a variety of QA tasks. These include multi-hop reasoning datasets (HotpotQA (Yang et al., 2018)), long-form retrieval-based QA datasets (TriviaQA (Joshi et al., 2017) and SearchQA (Dunn et al., 2017)), single-paragraph extraction tasks (SQuAD (Rajpurkar et al., 2016) and NewsQA (Trischler et al., 2017)), and document-level QA datasets (NQ (Kwiatkowski et al., 2019)). All datasets are formatted in the unified schema provided by the MrQA repository (Fisch et al., 2019). Additionally, we used the artificially constructed NQ-swap dataset (Longpre et al., 2021), designed to simulate conflicting or ambiguous scenarios by replacing entities.

D.1.1 Dataset Categories and Statistics

The datasets used in this study include seven open-book question-answering (QA) datasets, grouped into three categories based on their QA task characteristics: multi-hop reasoning, long-form retrieval-based QA, and single-paragraph extraction tasks. Additionally, an adversarial dataset is included for evaluating the robustness of the proposed method. Detailed descriptions and dataset statistics are provided below.

Multi-Hop Reasoning (HotpotQA). HotpotQA (Yang et al., 2018) is a benchmark dataset for multi-hop reasoning, requiring models to synthesize information across multiple paragraphs to generate an answer. This dataset emphasizes complex reasoning over distributed evidence, making it a critical benchmark for evaluating context utilization.

Long-Form Retrieval-Based QA (TriviaQA, SearchQA). TriviaQA (Joshi et al., 2017) and SearchQA (Dunn et al., 2017) require reasoning over longer contexts, with answers scattered across retrieved documents. These datasets test the model's ability to focus on relevant content in lengthy contexts and generate precise answers.

Single-Paragraph Extraction (SQuAD, NewsQA). SQuAD (Rajpurkar et al., 2016), and NewsQA (Trischler et al., 2017) are standard extractive QA datasets where the answer is typically located within a single paragraph. These datasets are widely used for evaluating the span-extraction capabilities of QA systems.

Document-Level QA (NQ). NQ (Kwiatkowski et al., 2019) is a document-level open-domain question answering dataset driven by real user queries. It requires systems to extract long answers from entire Wikipedia documents and generate specific short answers, evaluating document-level information retrieval and natural language understanding capabilities.

Simulated Conflict Scenarios (NQ-Swap). NQ-Swap (Longpre et al., 2021) is an artificially constructed dataset that introduces adversarial entity swaps into NQ to create ambiguous or conflicting contexts. It evaluates the model's ability to resolve conflicts and faithfully utilize context.

D.1.2 Dataset Sources and Formats

All datasets are standardized in the unified schema provided by the MrQA repository (Huggingface

Dataset	Number of Samples
HotpotQA (Multi-Hop)	5904
TriviaQA (Long-Form Retrieval)	7785
SearchQA (Long-Form Retrieval)	16980
SQuAD (Single-Paragraph)	10507
NewsQA (Single-Paragraph)	4212
NQ (Document-Level)	12836
NQ-Swap (Simulated Conflicts)	4746

Table 4: Dataset statistics. A summary of the dataset sizes used for evaluation across different datasets.

ID: mrqa-workshop/mrqa), except for NQ-Swap, which is sourced from a separate repository (Huggingface ID: pminervini/NQ-Swap). The datasets used for training the logistic regression model (§ 2) and attention analysis (§ 3) are drawn from the training sets of the MrQA repository. Model performance evaluation is conducted using the validation sets from the same repository. All datasets have been preprocessed to ensure compatibility with our experimental framework.

D.1.3 Dataset Statistics

Table 4 presents the size of the datasets used in this study to evaluate model performance.

D.2 Implementation Details

At each decoding step, DAGCD determines whether utilized tokens are detected by the Context Utilization Detector. If detected, their probabilities are amplified; otherwise, or if a termination condition is met (e.g., the top-1 token is "\n"), probabilities remain unchanged. All experiments utilized a unified prompt template (Prompt 1, as shown in Figure F) to ensure consistency across methods. The prompt format is detailed in Appendix F. For decoding, greedy decoding was employed to produce deterministic outputs and facilitate direct comparisons across methods. All models run on NVIDIA A100 GPUs.

D.3 Model Details

The LLMs used in this work, along with its HuggingFace ID, is as follows:

- LLaMA2-7B: meta-llama/Llama-2-7b-hf
- LLaMA2-7B-Chat: meta-llama/Llama-2-7b-chat-hf
- LLaMA2-13B: meta-llama/Llama-2-13b-hf
- LLaMA2-13B-Chat: meta-llama/Llama-2-13b-chat-hf
- Mistral-7B: mistralai/Mistral-7B-v0.1
- Mistral-7B-Instruct: mistralai/Mistral-7B-Instruct-v0.1

D.4 Baseline Configurations

We compare the proposed method DAGCD with three decoding strategies: Greedy Decoding, CAD (Shi et al., 2024b), and COIECD (Yuan et al., 2024). CAD and COIECD are specifically designed to mitigate context faithfulness hallucination. We implemented the baseline methods with their recommended hyperparameter settings for fair comparisons:

- **CAD** (Shi et al., 2024b): The contrastive adjustment factor α was set to 1.
- **COIECD** (Yuan et al., 2024): The entropy regularization parameter λ was set to 0.25, and the contrastive adjustment factor α was set to 1.

D.5 Results on Summarization Tasks

To validate the performance of our approach on long-form answer generation tasks, we conducted experimental evaluations on the CNN_DM (See et al., 2017) summarization dataset (we randomly sampled 500 instances from the dataset for evaluation). Similar to prior work (Shi et al., 2024b), we adopted ROUGE-L (Lin, 2004), factKB (Feng et al., 2023), and BERTScore (Zhang et al., 2020) as comprehensive evaluation metrics to assess both the accuracy and factual consistency of the generated content. The experimental results, as shown in Table 5, demonstrate that our method achieves significant improvements on both the pretrained and chat versions of LLaMA2.

E Detailed Results of "5.3 Ablation Study"

E.1 Additional Results of "Ablation 1: Detector Training Data Sizes"

Table 12 shows the performance variations of all LLMs used in this study when trained with detectors on different amounts of data. As observed, our method maintains consistent performance across various data sizes for all LLMs.

E.2 Additional Results of "Ablation 3: Scaling Factor α"

We additionally evaluated the performance variations of our method on Mistral-7B and Mistral-7B-Instruct under different scaling factors α. Figure

Model	Decoding	ROUGE-L	factKB	BERT-P	BERT-R	BERT-F1
LLaMA2-7B	Greedy	0.2081	**0.9932**	0.9000	0.7997	0.8465
	CAD	**0.2361**	0.9786	0.9054	0.8016	0.8514
	COIECD	0.2089	0.9845	0.9152	0.8014	0.8543
	OURs	0.2134	0.9856	**0.9210**	**0.8026**	**0.8576**
LLaMA2-7B-Chat	Greedy	0.2368	0.9846	0.9056	0.8035	0.8515
	CAD	0.2082	0.9417	0.9001	0.7997	0.8466
	COIECD	0.2371	0.9807	0.9055	0.8034	0.8513
	OURs	**0.2426**	**0.9866**	**0.9104**	**0.8036**	**0.8536**

Table 5: Comparison of evaluation results on CNN/DailyMail. The table compares the evaluation results between greedy decoding and our proposed method on the CNN/DailyMail dataset. **Bold** denotes the best performance, while underlined indicates the second-best performance.

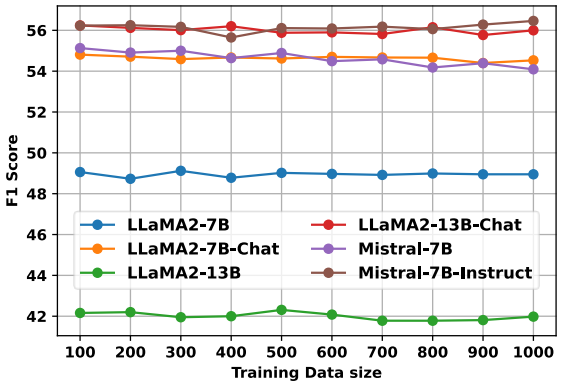

Figure 12: Detector Training Data Size Validation. The variation in inference performance across different models when using detectors trained on varying amounts of data.

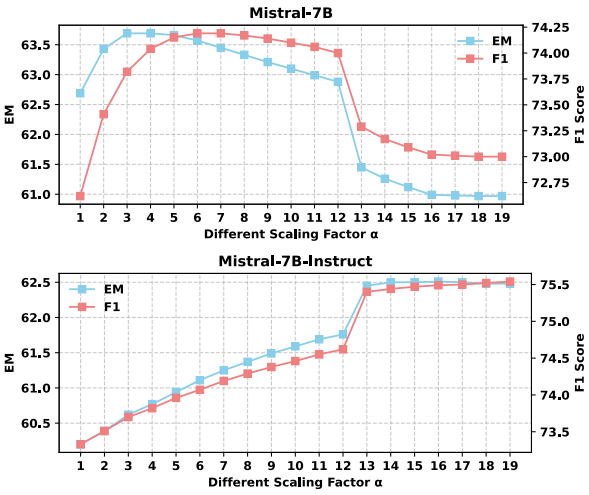

Figure 13: The performance on HotpotQA for DAGCD under different scaling factors.

13 illustrates the performance changes on the HotpotQA dataset. For Mistral-7B, the optimal performance is achieved at $\alpha = 5$. In contrast, for Mistral-7B-Instruct, the performance only stabilizes after $\alpha = 13$. This indicates that different models may require different optimal scaling factors for the best performance.

E.3 Impact of Different Prompts

To assess robustness to prompt variations, we tested multiple prompts from prior studies (Zhou et al., 2023; Yuan et al., 2024; Wang et al., 2024) (templates in Figure 15). Figure 14 illustrates the variations in F1 scores for LLaMA2-7B and Mistral-7B on the HotpotQA and NewsQA datasets under different prompt templates. The results show that DAGCD consistently outperforms baselines across all tested prompts, demonstrating its adaptability to diverse input formats and reliability across QA tasks.

F Prompt Templates

Figure 15 shows the prompt templates used in this paper.

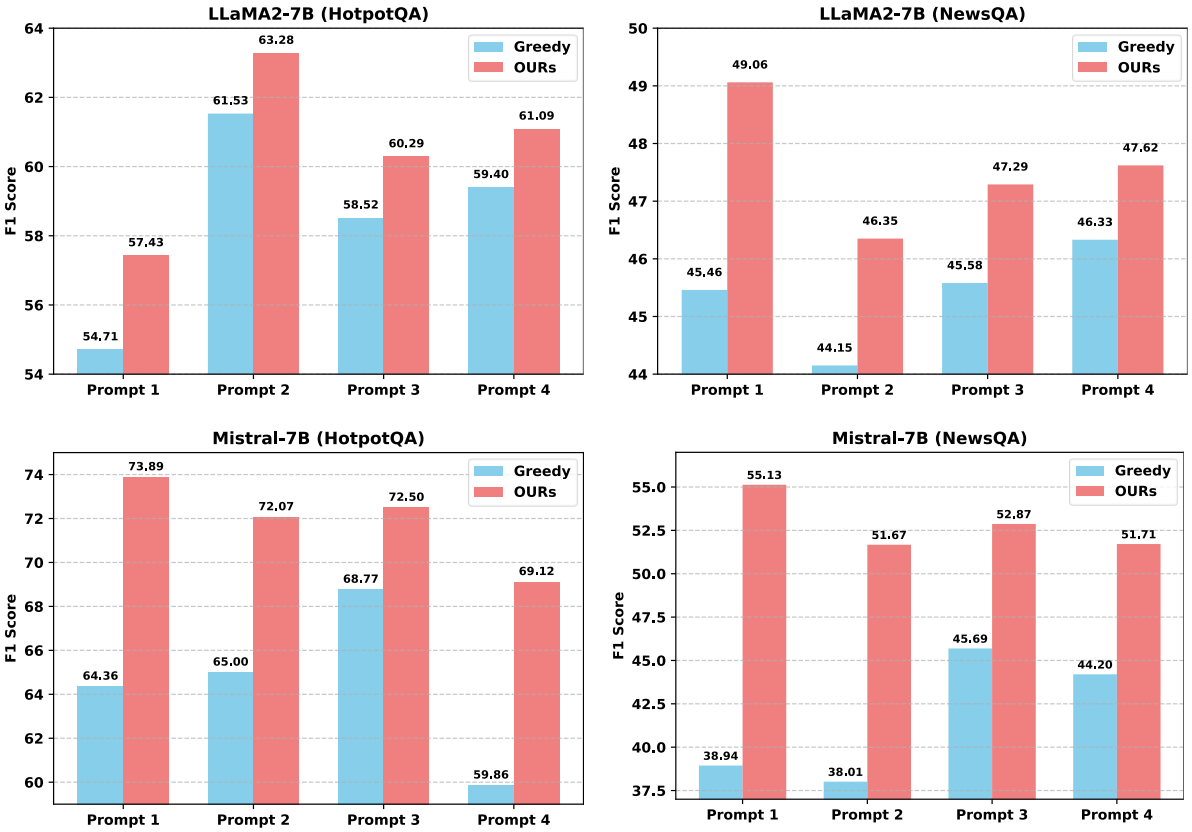

Figure 14: Performance variations across different prompt templates. The figure shows F1 score variations on the HotpotQA and NewsQA datasets for Greedy Decoding and OURs (DAGCD) under different prompt templates.

Prompt 1

With Context:
Given the following information: {context}
Answer the following question based on the given information with one or few words: {question}
Answer:

Without Context: (for CAD and COIECD)
Answer the following question based on your internal knowledge with one or few words:{question}
Answer:

Prompt 2

Given the following context, answer the question below:
Context: {context}
Question: {question}
Answer:

Prompt 3

Read the given information and answer the corresponding question. {context}
Question: {question}
Answer:

Prompt 4

{context}
Using only the references listed above, answer the following question:
Question: {question}
Answer:

Figure 15: Prompt templates used in this paper.