Gradient-guided Attention Map Editing: Towards Efficient Contextual Hallucination Mitigation

Anonymous ACL submission

Abstract

In tasks like summarization and open-book question answering (QA), Large Language Models (LLMs) often encounter "contextual hallucination", where they produce irrelevant or incorrect responses despite having access to accurate source information. This typically occurs because these models tend to prioritize self-generated content over the input context, causing them to disregard pertinent details. To address this challenge, we introduce a novel method called "Guided Attention Map Editing" (`GAME`), which dynamically adjusts attention maps to improve contextual relevance. During inference, `GAME` employs a trained classifier to identify attention maps prone to inducing hallucinations and executes targeted interventions. These interventions, guided by gradient-informed "edit directions", strategically redistribute attention weights across various heads to effectively reduce hallucination. Comprehensive evaluations on challenging summarization and open-book QA tasks show that `GAME` consistently reduces hallucinations across a variety of open-source models. Specifically, `GAME` reduces hallucinations by **10%** in the XSum summarization task while achieving a **7X** speed-up in computational efficiency compared to the state-of-the-art baselines.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities across various natural language processing tasks. Despite these advances, their practical deployment is often compromised by their propensity to produce hallucinated outputs (Huang et al., 2023; Ji et al., 2023a; Tonmoy et al., 2024). Hallucinations arise from multiple sources, and significant research efforts have focused on detecting and mitigating them. For example, hallucinations due to outdated or incomplete knowledge in the training data can be addressed through techniques such as Retrieval-Augmented Generation (RAG) (Gao et al., 2024) or knowledge injection (Ovadia et al., 2023; Zhang et al., 2024a). Recent methods that intervene in the hidden representations(Li et al., 2024; Dathathri et al., 2019; Chuang et al., 2024b) of language models during inference have shown substantial improvements in reducing hallucinations related to inherent confirmation bias and spurious correlations (Zhang et al., 2024c).

This work mainly targets a particularly critical phenomenon known as "contextual hallucination" (Ainsworth et al., 2024; Chuang et al., 2024a), also referred to as "openbook hallucination" (Simhi et al., 2024). This occurs when LLMs generate misleading or unrelated content despite having access to pertinent information in the input context. Contextual hallucination presents significant challenges, particularly in high-stakes domains such as finance and healthcare, where accurate retrieval of relevant information is crucial. Failures in these contexts can lead to severe consequences, underscoring the urgent need for effective mitigation strategies.

Contextual hallucination represents an inherent deficiency within LLMs that cannot be resolved merely by injecting new knowledge or providing additional information, which makes it a difficult challenge. It often arises from a poor correlation between the input context and the generated outputs. Previous efforts, such as those in Shi et al. (2023) and van der Poel et al. (2022), have encouraged enhancing the mutual correlation between the input context and generation to reduce hallucinations. More recently, Chuang et al. (2024a) explores deep into the architecture of transformers, suggesting that contextual hallucinations stem from a suboptimal distribution of attention between context and self-generation. The studies also demonstrate that identifying problematic attention maps can be used to model and mitigate contextual hallucinations. However, while promising, these methods fail to actively intervene in the attention maps but rather select the most promising outputs from multiple random samples. These passive approaches, relying heavily on the inherent capabilities of LLMs, can be inefficient and may not consistently correct the root cause of hallucinations.

Inspired by these insights, we argue that directly modifying problematic attention maps shall be a more effective way to reduce contextual hallucination. This intervention encourages the model to focus more on pertinent content, enhancing relevance and coherence in the generated text. To assess this hypothesis, we initiated a behavioral study to evaluate the impact of attention editing on LLMs. We designed experiments to bias the model's attention towards the context during inference, aiming to guide the model to prioritize contextually relevant information.

Our preliminary findings reveal that directing attention towards contextual elements can prompt the model to produce more grounded and contextually relevant content, thus minimizing instances of ungrounded generation. However, the study also underscores the importance of precise attention editing. Arbitrary modifications to attention maps can disrupt the natural functioning of LLMs and may inadvertently introduce additional errors or hallucinations. This highlights the critical need for targeted and carefully calibrated interventions in attention mechanisms to avoid unintended consequences.

To address these challenges, we propose "Guided Attention Map Editing" (`GAME`), a method designed to perform precise interventions on attention maps to reduce contextual hallucination. `GAME` employs a classifier trained to identify problematic attention maps that are likely to induce hallucinations during inference. Once such maps are detected, `GAME` triggers an intervention to regenerate the corresponding outputs. This process involves using gradient information, referred to as "edit direction", to reallocate attention weights across different attention heads strategically. Importantly, the edit direction is tailored for each head, recognizing important findings in past literature on the diverse functionality of different attention heads (Zheng et al., 2024b). The proposed `GAME` can effectively reduce the hallucination rate by 10% of Llama2-7b on XSum (Narayan et al., 2018) dataset with negligible additional computation cost. The contributions of this work are threefold:

- We conduct a behavioral study to investigate the impact of editing attention maps in producing more contextually aware content and highlight the importance of editing direction.
- We propose `GAME`, a novel approach leveraging gradient information to guide the editing of attention map precisely, reducing contextual hallucinations efficiently.
- We demonstrate `GAME` on extensive experiments, including summarization and open-book QA tasks, showing superior performance compared to the state-of-the-art baseline methods.

2 Preliminaries

2.1 Attention Mechanism in Transformers

Most LLMs today utilize the Transformer architecture (Vaswani, 2017), which predominantly features a decoder-only architecture and employs multi-head attention mechanism (Zheng et al., 2024a) to effectively handle the complex correlations among input tokens, as shown in Figure 1.

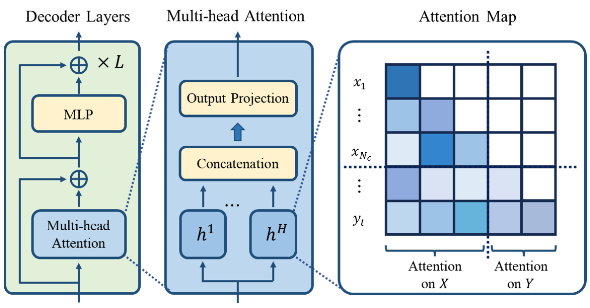

Figure 1: Illustration of a decoder-only Transformer featuring a multi-head attention mechanism. Each row in an attention map represents a weight vector that sums to one, reflecting the current token's relationship with preceding tokens. A deeper color indicates a higher attention weight.

We examine a Transformer model consisting of L decoder layers, each equipped with H attention heads, indexed by l and h for the layer and head, respectively. The model processes input with length N, which is a concatenation of the context $X = [x_1, x_2, \ldots, x_{N_c}]^{N_c}$ and the preceding generated sequence $Y = [y_1, y_2, \ldots, y_t]^{N_g}$, where N_c and N_g denote the lengths of the context and generated sequence, respectively.

As the model predicts the next token, each attention head independently computes an attention map from the input. These maps are lower triangular matrices where each row consists of weights that

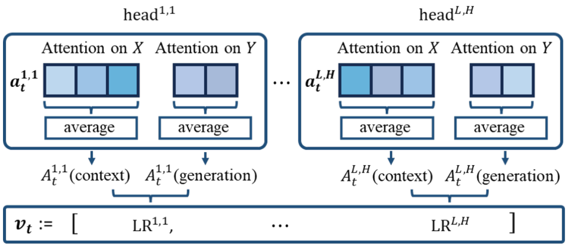

Figure 2: Derivation of the LR at the t_{th} decoding step.

sum to one, reflecting the relationship of the current token with preceding tokens. A higher weight within this matrix indicates a stronger correlation, suggesting that the model is more likely to generate tokens closely related to those with higher weights. Importantly, different attention heads are tailored to focus on various aspects of the input, allowing for a comprehensive, multi-faceted analysis of token relationships.

2.2 Attention Feature for Detecting Contextual Hallucination

Contextual hallucination occurs when LLMs generate outputs that do not exist or cannot be inferred from the provided context. Past literature (Chuang et al., 2024a) has demonstrated that the model's lack of attention to the context during the generation process can be the cause and utilizing the attention map as a feature can effectively detect such contextual hallucination. Specifically, an attention map based feature named "Lookback Ratio" (LR) is proposed to model contextual hallucination. As shown in Figure 2, at the t_{th} decoding step (to predict y_{t+1}), for the h_{th} head in the l_{th} layer, its LR is defined as:

$$\text{LR}_t^{l,h} = \frac{A_t^{l,h}(\text{context})}{A_t^{l,h}(\text{context}) + A_t^{l,h}(\text{generation})}, \quad (1)$$

where

$$A_t^{l,h}(\text{context}) = \frac{1}{N_c}\sum_{i=1}^{N_c} a_i^{l,h},$$
$$A_t^{l,h}(\text{generation}) = \frac{1}{N_g}\sum_{j=N_c+1}^{N} a_j^{l,h}, \quad (2)$$

$\boldsymbol{a^{l,h}}$ denotes the post-Softmax attention weights of the head. Across all heads, the lookback ratio vector is defined as:

$$\boldsymbol{v}_t = \left[\text{LR}_t^{1,1}, \text{LR}_t^{1,2} \cdots \text{LR}_t^{L,H}\right]. \quad (3)$$

The lookback ratio indicates the model's focus on context versus its own output during next-token prediction, with a higher ratio suggesting greater emphasis on context.

3 GAME: Gradient-guided Attention Map Editing

To mitigate contextual hallucination, we propose a strategic intervention in the attention mechanisms of these LLM models. By editing the attention maps, we aim to enhance the focus on contextual inputs, thereby anchoring the LLMs' outputs more effectively in the provided context and thereby reducing hallucinated contents.

We first conduct a behavioral study to examine the impact of introducing prior biases that augment attention weights towards contextual elements. While this approach yields promising results in summarization tasks, it simultaneously surfaces significant questions about the granularity and specificity of attention manipulation. Overly coarse interventions may inadvertently degrade the generative performance of LLMs.

To address the challenges, we propose GAME, a Gradient-guided Attention Map Editing method, which combines prior bias adjustments with gradient signals obtained from an attention feature-based hallucination classifier to facilitate precise and oriented modifications on attention maps, enabling more effective mitigation of contextual hallucinations.

3.1 Attention Map Editing with Prior Bias

We consider linear intervention to the attention map by adding a prior bias ($\boldsymbol{b}$) on the original attention map. In the implementation, we add the prior bias to the raw attention scores ($\boldsymbol{s}$) before the Softmax normalization step in the attention mechanism. This ensures a valid modified attention map after intervention:

$$\boldsymbol{a} = \text{Softmax}(\boldsymbol{s} + \eta \cdot \boldsymbol{b}), \quad (4)$$

where η is a hyperparameter that adjusts the intensity of the intervention.

Our design of the prior bias follows two principles: Firstly, the bias should enhance the model's focus on contextual information relative to self-generated content to help mitigate the effects of contextual hallucinations. As Softmax operation is order-preserving, this essentially requires the sum of the bias on the context part should be larger than the generation part, as in Equation (5).

$$\sum_{i=1}^{N_c} b_i > \sum_{j=N_c+1}^{N} b_j. \tag{5}$$

Secondly, it needs to counteract the natural decay of attention that occurs with increasing distance between tokens—a common challenge as discussed in (Liu et al., 2023; Chen et al., 2024a). Consequently, our bias implementation employs a reverse function and takes in the form: $b_i = \frac{1}{i}$, where b_i denotes bias of the i_{th} token when the LLM generates the current token. This function amplifies the model's attention to more distant tokens, effectively counteracting the typical attention decay observed in models like Transformers. A simple illustration of the naive attention editing is shown in Figure 3.

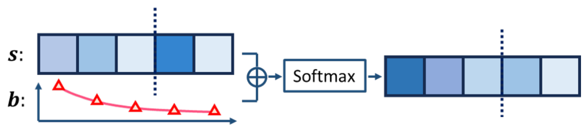

Figure 3: Utilization of the positional-based decay prior attention bias.

Empirical Analysis. We apply the naive attention editing uniformly across each attention head in the Llama2-7b model (Touvron et al., 2023). The experiments assess the impact on ROUGE (Lin, 2004) scores, testing the summarization capabilities on the XSum dataset. Details of this experimental setup are provided in the Appendix, with results summarized in Table 1.

Incorporating a prior attention bias can indeed encourage the model to generate more context-aware outputs, reflected in an increase of the score. However, as η increases, the large bias gradually disrupts the original behavior of attention heads and thus leads to dramatic model degradation. While the results demonstrate the effectiveness of the prior bias, they also highlight the need for precise attention editing. Specifically, two critical questions arise:

- **Q1**: *when should we apply attention editing?*
- **Q2**: *where should we perform attention editing?*

Addressing **Q1** relies on a robust method for detecting contextual hallucination. Given that hallucinations are rare and abnormal events, intervention is necessary only when hallucinated contents are detected. This targeted approach prevents arbitrary bias application, which could otherwise alter the model's desired behavior.

Regarding **Q2**, it is important to recognize that different attention heads exhibit diverse functional focuses (Michel et al., 2019)—some prioritize contextual coherence, while others emphasize content generation. Applying a bias without understanding these distinctions can significantly disrupt their intrinsic behaviors. Therefore, precise identification and selective editing of attention heads are crucial, in determining whether to enhance their focus on contextual coherence or content generation.

Intensity η	ROUGE 1	ROUGE 2	ROUGE L
0	0.2396	0.0669	0.1682
1	0.2422	0.0711	0.1697
2	0.2279	0.0634	0.1624
5	0.0832	0.0143	0.0624

Table 1: ROUGE score on XSum dataset for Llama2-7b by applying prior attention bias with different edit intensity η.

3.2 Gradient-guided Editing

`GAME` introduces two advanced techniques addressing the issues previously identified. Initially, `GAME` employs a hallucination classifier that utilizes attention features as input to compute a hallucination score. If the generation's score fails to meet a predefined threshold (λ), it is indicative of hallucination, thereby necessitating attention editing.

Moreover, the classifier not only detects hallucination but also provides gradient information to inform the application of prior biases across different attention heads. This gradient information, termed "edit direction" ($\mathbf{\Delta}$), is a signed binary vector that indicates whether an attention head should increase its focus on the context to elevate the hallucination score, thereby optimizing attention allocation in response to detected hallucinations.

In practice, `GAME` processes outputs in equally sized chunks. An illustrated depiction of the process for generating one chunk is shown in Figure 4. The details of training the classifier and deriving the edit direction are elaborated as follows. To ensure precise and effective modification of the attention maps, and to account for the varied roles of different attention heads, we utilize "edit direction ($\mathbf{\Delta}$)" derived from the gradient information from the classifier. This direction informs how the prior attention bias should be applied to each attention head, optimizing the attention allocation in

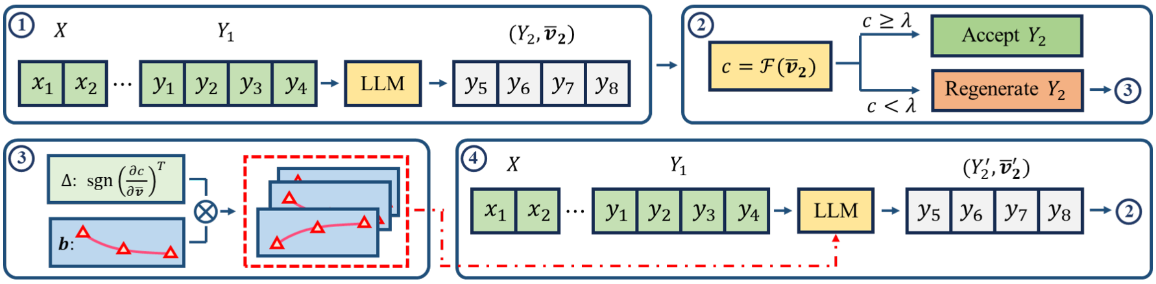

Figure 4: Illustrated example on the generation of one chunk of output in GAME. **Step** ①: the LLM predicts the next chunk (Y_2) and calculates the chunk attention feature ($\bar{\boldsymbol{v}}_2$) without any attention editing. **Step** ②: the classifier ($\mathcal{F}$) predicts the hallucination score (c) for the generated chunk with the corresponding feature. If the score exceeds a predefined threshold, the chunk will be accepted. Otherwise, attention editing will be applied to regenerate the chunk. **Step** ③: the attention edit signal for each head is computed with the prior bias and the edit direction $\boldsymbol{\Delta}$ derived from the gradient of the score. **Step** ④: a new chunk is generated with the calculated attention editing signal and re-evaluated with the classifier. If no qualified chunk is accepted with number of regeneration attempts, the chunk with the highest score during the generation process will be accepted.

response to the detected hallucinations. An illustrative overview is depicted in Figure 4.

3.2.1 Training the Classifier

We train a linear classifier based on the lookback ratio feature to model contextual hallucination. Notably, contextual hallucinated contents usually constitute only a portion of the entire generated text, with the remainder being accurate and relevant. Therefore, it is crucial to model hallucination with greater precision, to accurately capture the correlation between problematic attention features and the corresponding hallucination outputs. To train the classifier, the Llama2-7b model is prompted to generate summaries from a subset of 1,000 articles sampled from the Daily Mail CNN dataset (Nallapati et al., 2016). The generated outputs are segmented into fixed-size chunks. Each chunk is then annotated by GPT-4o, which assigns binary labels indicating the presence or absence of hallucination, as depicted in Figure 5. The attention feature of these chunks, along with their corresponding labels, are used as the training data for the classifier.

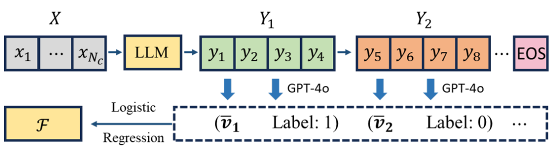

Figure 5: The training data construction and training process of lookback lens.

3.2.2 Deriving the Edit Direction Δ.

Given a detected hallucinated chunk (Y) with its corresponding feature $\bar{v}$ and the computed score c, the edit direction for regenerating this chunk is defined as:

$$\boldsymbol{\Delta} = \mathrm{sgn}(\left[\frac{\partial c}{\partial \bar{\boldsymbol{v}}}\right]^T), \tag{6}$$

where

$$\mathrm{sgn}(x) = \begin{cases} 1 & \text{if } x \geq \epsilon \\ 0 & \text{if } -\epsilon < x < \epsilon \\ -1 & \text{if } x \leq -\epsilon, \end{cases} \tag{7}$$

with ϵ as a pre-defined threshold parameter. During regeneration, the prior bias is multiplied by the edit direction and then added to the original attention, as shown in Figure 6.

$$\boldsymbol{a} = \mathrm{Softmax}(\boldsymbol{s} + \eta \cdot \Delta \cdot \boldsymbol{b}) \tag{8}$$

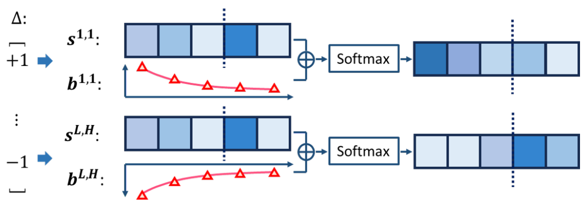

Figure 6: Illustrated example for the combination of prior attention bias and edit direction to perform attention editing.

3.3 Interpretation of Edit Direction Δ

The derivation of $\boldsymbol{\Delta}$ incorporates three key considerations to effectively guide attention map editing.

(1) Interpretation of the Gradient. The gradient term, $\left[\frac{\partial c}{\partial \bar{v}}\right]^T$, quantifies how modifications to the average lookback ratio of attention features influence the hallucination score. A higher lookback ratio, which indicates a greater focus on contextual information, is generally associated with reduced hallucination. The gradient thus reflects the necessary change in attention focus to mitigate hallucination effects. However, since this gradient is averaged across all tokens within a chunk, direct application in editing is impractical. Instead, we utilize the sign of the gradient (sgn) to determine the general direction for modifying the attention, thereby guiding the regeneration process in a binary manner—either increasing or decreasing focus based on the context.

(2) Effect of Sign Function. Utilizing the sign function simplifies the gradient information to a directional indicator that instructs whether to enhance or reduce the attention focus on specific elements of the context. This approach avoids the complexities and potential overfitting that might arise from using the precise gradient values, providing a robust mechanism for attention modification.

(3) The Role of Gradient Threshold. To further refine our approach, we introduce a gradient threshold, ϵ, which serves to filter out attention heads with relatively minor gradients. This selection criterion ensures that only those heads with substantial discrepancies in attention allocation—indicating a strong need for adjustment—are edited. Attention heads with gradients below this threshold are considered adequately aligned and are not subjected to modification. This selective editing helps maintain the model's overall stability and prevents unnecessary adjustments that could disrupt the model's performance.

4 Experiments

We evaluate the proposed method, `GAME`, on summarization task and open-book QA task, with two different open source models: Llama2-7B (Touvron et al., 2023) and Phi3-mini (Abdin et al., 2024). The detailed configuration of datasets and evaluation metrics for contextual hallucination are summarized below.

4.1 Experimental Setup

Datasets. We adopt Daily CNN mail (Nallapati et al., 2016) and Extreme Sum (XSum) (Narayan et al., 2018) for the summarization task. Following the setup in (Chuang et al., 2024a), we randomly select 1000 data points from the whole datasets for evaluation.

For the openbook QA task, we adopt the constructed Natural Question (Kwiatkowski et al., 2019) dataset in (Chuang et al., 2024a; Liu et al., 2023) and follow the sample principle to construct a subset of 1000 data points from the Trivial QA (web) (Joshi et al., 2017) dataset. Details for preparing the datasets can be found in Appendix A.

Evaluation Metrics. For the summarization task, we use GPT-4o as a judge to detect whether there is any hallucination in the summarization. The non-hallucination rate (accuracy) is defined as the number of non-hallucinated summarizations over the total number of data points, which is higher the better. For the openbook QA task, we calculate the best span exact match (EM) rate.

Model Configurations All baseline models utilize greedy search for decoding. For guided attention editing, we train the classifier using Logistic regression as detailed in Appendix C. This trained classifier is shared by all models across experiments.

4.2 Main Results

We demonstrate `GAME` improves both accuracy and EM rate compared to the baseline LLMs. As summarized in Table 2. Across 4 datasets and 2 models, applying the proposed `GAME` effectively reduces contextual hallucination. Specifically, on XSum, Llama2-`GAME` improves the accuracy by 10%, reducing the number of hallucinated generations from 510 samples to 410 samples in a total of 1000 samples. On other datasets, Llama2-`GAME` demonstrates consistent improvement over 2% on other datasets. This demonstrates the overall effectiveness of the proposed method.

The Necessity of Edit Direction. We additionally analyze the importance of utilizing the editing direction when doing attention editing. We compare a Llama2-7b model utilizing edit direction (denoted by Llama2-`GAME`-w direction), with a Llama2-7b that uniformly applies the prior attention bias (denoted by Llama2-`GAME`-w/o direction), on XSum and NQ-Open. The corresponding results are shown in Table 3.

Consistent with our preliminary findings, the uniform application of prior attention bias does enhance the model's focus on contextual elements, yielding improved performance relative to the baseline. However, the incorporation of edit direction further enhances overall performance. This sup-

Models	CNN/Daily Mail	XSum	NQ-Open	Trivial QA
Llama2-base (Touvron et al., 2023)	0.214	0.490	0.712	0.838
Llama2-GAME (ours)	**0.232**	**0.590**	**0.732**	**0.868**
Phi3-base (Abdin et al., 2024)	0.203	0.504	0.690	0.803
Phi3-GAME (ours)	0.225	0.523	0.720	0.823

Table 2: Results on summarization and openbook QA for baseline Llama2-7b, Phi3-mini and their corresponding counterpart with the proposed GAME.

ports literature indicating that different attention heads contribute variably to the generation process and underscores the critical need for employing edit direction in the modification of attention maps to optimize model efficacy.

Models	XSum	NQ-Open
Llama2-base (Touvron et al., 2023)	0.490	0.712
Llama2-GAME with direction	**0.590**	**0.732**
Llama2-GAME without direction	0.539	0.717

Table 3: Results on XSum and NQ-Open for Llama2-GAME with direction and Llama2-GAME without direction.

4.3 Analysis

Computational Efficiency. Authors in (Chuang et al., 2024a) mitigate hallucination by applying random sampling to generate candidate chunks and accept chunks with the highest score produced by the classifier. This method can suffer from two deficiencies. First, it relies on the model's original ability, assuming the models can eventually generate less hallucinated outputs via repeated sampling. Second, the repetitive sampling process for each chunk reduces the efficiency of the method.

We compare the proposed method and the decoding method in (Chuang et al., 2024a) on XSum and NQ-Open, following the setup in their paper. We summarize the results and the averaged cost, denoted by Run-time (in seconds) per sample on XSum as in Table 4. The results show that our proposed method outperforms lookback lens guided decoding on XSum and is **7X** more efficient. For the NQ-Open dataset, lookback lens guided decoding benefits from high temperature in generating diverse outputs for final selection, but at the cost of low efficiency.

Models	XSum	NQ-Open	Run-time/sample
Llama2-GAME	**0.590**	0.732	2.69s
Lookback Lens	0.583	**0.742**	18.93s

Table 4: Comparison between Llama2-GAME and lookback lens guided decoding on XSum and NQ-Open.

Comparison of different prior bias. The design and choice of the prior attention bias also affects the intervention performance. We consider and compare another uniform bias where $b_i = 1$ for $i \leq N_c$ and $b_i = 0$ for $i > N_c$. In our experiments, we compared the performance of the Llama2 model with two different variants: Llama2-GAME-decay and Llama2-GAME-uniform, on the XSum dataset. Both Llama2-decay and Llama2-uniform demonstrated improvements over the baseline Llama2 model. Notably, Llama2-GAME-decay outperformed Llama2-GAME-uniform by an additional $\mathbf{5.7\%}$, suggesting that a dynamically scaled bias, which accounts for the positional relevance within the context, is more effective in enhancing model performance.

Models	XSum
Llama2-GAME-decay bias	**0.590**
Llama2-GAME-uniform bias	0.543

Table 5: Results on XSum for Llama2-GAME-decay bias and Llama2-GAME-uniform bias.

Impact of Edit Intensity. The hyper-parameter η plays a critical role in modulating the extent of intervention during the attention map regeneration process. A value of η that is too low may not sufficiently influence the attention map, thereby failing to alter the model's final outputs significantly. Conversely, an excessively high value of η can disrupt the intrinsic behavior of the target LLM, leading to compromised generation quality. We systematically evaluated the impact of various settings of η, ranging from 0.1 to 2.0, using the NQ-Open dataset. The results, illustrated in Figure 7 (Left), are consistent with our intuition: moderate increases in η enhance the model's focus on relevant context and improve the performance. However, excessively high values result in a dramatic degradation of model behavior.

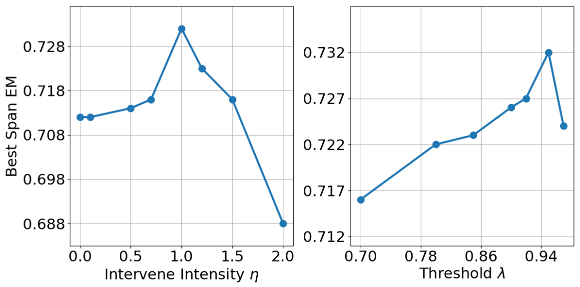

Figure 7: Illustrated results on the impact of different η (Left) and different λ (Right) on the performance on NQ-Open when applying GAME.

Effect of Editing Threshold. The threshold λ serves as a criterion to decide whether a generated chunk needs regeneration and subsequent attention editing. A higher λ imposes stricter criteria for accepting generated chunks and an excessively high threshold can potentially lead to rejection of even well-formed outputs. In such case, GAME can inadvertently lead to the unnecessary regeneration of originally accurate chunks, thereby disrupting their quality. We conducted an evaluation of different λ values, ranging from 0.7 to 1.0, on the NQ-Open dataset using the Llama2-7B model. The results, depicted in Figure 7 (Right), indicate that increasing λ initially leads to more chunks being flagged for regeneration, which correlates with a performance improvement. However, further increases in λ result in a performance decline, as even well-formed chunks are subjected to unnecessary regeneration, leading to outputs that deviate from the model's originally accurate generations.

5 Related Work

Attention Mechanism. Attention heads have been found to be closely related to the model behavior and focus on different tasks, from knowledge rescaling to latent reasoning, as summarized in a recent survey (Zheng et al., 2024b). Liao and Vargas (2024) introduces dropout before the feed-forward network, aiming to recalibrate attention matrices, focusing more on semantically important tokens to reduce the influence of outlier high-score tokens. Yu et al. (2024) dive into "attention sinks" and reveals that certain tokens disproportionately attract attention without adding semantic value. It proposes to recalibrate these attention distributions to enhance LLM reasoning ability. Chuang et al. (2024a) proposes to detect and mitigate contextual hallucination with a feature, named by "Lookback Ratio" derived from the attention map. Differing from these studies, our proposed method, GAME, focuses on actively editing attention maps via gradient-oriented information to control context-aware generation in LLMs.

Hallucination Mitigation with Representation Editing. Extensive research efforts have been devoted to mitigating various types of hallucinations in LLMs (Huang et al., 2023; Tonmoy et al., 2024; Chen et al., 2024b; Ji et al., 2023b; Luo et al., 2024; Wang et al., 2024). A significant line of works propose to intervene on hidden representation to mitigate hallucination. For instance, Plug and Play (Dathathri et al., 2019) leverages the gradient of an attribute model to adjust the hidden representations, guiding LLMs toward generating outputs with specific desired attributes. ITI (Li et al., 2024) employs neural probing (Alain and Bengio, 2018) to classify attention head outputs, suggesting adjustments to activations during inference to enhance factual correctness. Zhang et al. (2024b) integrates an auto-encoder to split the hidden representation into components related to factual and semantic content, thus enabling more precise control. Chuang et al. (2024b) posits that factual information is progressively revealed across decoder layers and introduces a method of contrastive decoding between layers to highlight this dynamic. These methodologies primarily address the mitigation of closed-book hallucination, as defined in (Simhi et al., 2024). However, GAME diverges from these approaches in two key aspects. Firstly, GAME specifically targets contextual hallucination, rather than the closed-book scenario. Secondly, it is inspired by the observed correlations between attention mechanisms and contextual hallucination, directly intervening at the level of the attention map to influence output generation.

6 Conclusion

We introduce GAME, a novel method designed to perform precise attention map editing to counteract contextual hallucination. This is achieved by incorporating meticulously designed prior bias and gradient information derived from hallucination classifiers. GAME has been rigorously tested across two open-source LLMs on four distinct datasets, effectively demonstrating its capability to mitigate contextual hallucination. Future research will aim to further enhance the efficiency and explore its applicability to a broader range of tasks.

7 Limitations

While the proposed guided attention editing method demonstrates significant improvements in reducing contextual hallucination, this work still presents limitations that pave the way for promising future research. First, the prior attention bias, though effective, is currently heuristic-based and could potentially be optimized through learning from a small dataset to more accurately intervene in the model's behavior. Second, our method adheres to a "detect then mitigate" paradigm, where hallucinations are identified and rectified post-generation in discrete chunks. However, the propensity of the model to generate hallucinated content might be identifiable prior to actual content generation. Early detection and prediction of potential hallucinations represent a compelling direction for future research, which could lead to more proactive strategies in managing and mitigating errors in LLMs.

References

Marah Abdin et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Eloise Ainsworth, Justin Wycliffe, and Florence Winslow. 2024. Reducing contextual hallucinations in large language models through attention map optimization. *Authorea Preprints*.

Guillaume Alain and Yoshua Bengio. 2018. Understanding intermediate layers using linear classifier probes. *Preprint*, arXiv:1610.01644.

Longze Chen, Ziqiang Liu, Wanwei He, Yunshui Li, Run Luo, and Min Yang. 2024a. Long context is not long at all: A prospector of long-dependency data for large language models. *Preprint*, arXiv:2405.17915.

Shiqi Chen, Miao Xiong, Junteng Liu, Zhengxuan Wu, Teng Xiao, Siyang Gao, and Junxian He. 2024b. In-context sharpness as alerts: An inner representation perspective for hallucination mitigation. *arXiv preprint arXiv:2403.01548*.

Yung-Sung Chuang, Linlu Qiu, Cheng-Yu Hsieh, Ranjay Krishna, Yoon Kim, and James Glass. 2024a. Lookback lens: Detecting and mitigating contextual hallucinations in large language models using only attention maps. *Preprint*, arXiv:2407.07071.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2024b. Dola: Decoding by contrasting layers improves factuality in large language models. *Preprint*, arXiv:2309.03883.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: A simple approach to controlled text generation. *CoRR*, abs/1912.02164.

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.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

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

Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023b. Towards mitigating llm hallucination via self reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1827–1843.

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.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024. Inference-time intervention: Eliciting truthful answers from a language model. *Preprint*, arXiv:2306.03341.

Bingli Liao and Danilo Vasconcellos Vargas. 2024. Extending token computation for llm reasoning. *Preprint*, arXiv:2403.14932.

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.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *Preprint*, arXiv:2307.03172.

Junliang Luo, Tianyu Li, Di Wu, Michael Jenkin, Steve Liu, and Gregory Dudek. 2024. Hallucination detection and hallucination mitigation: An investigation. *arXiv preprint arXiv:2401.08358*.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? *Preprint*, arXiv:1905.10650.

Ramesh Nallapati, Bowen Zhou, Cicero Nogueira dos santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *Preprint*, arXiv:1602.06023.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *Preprint*, arXiv:1808.08745.

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

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Scott Wen tau Yih. 2023. Trusting your evidence: Hallucinate less with context-aware decoding. *Preprint*, arXiv:2305.14739.

Adi Simhi, Jonathan Herzig, Idan Szpektor, and Yonatan Belinkov. 2024. Constructing benchmarks and interventions for combating hallucinations in llms. *Preprint*, arXiv:2404.09971.

SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*.

Hugo Touvron et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Liam van der Poel, Ryan Cotterell, and Clara Meister. 2022. Mutual information alleviates hallucinations in abstractive summarization. *Preprint*, arXiv:2210.13210.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Song Wang, Xun Wang, Jie Mei, Yujia Xie, Sean Muarray, Zhang Li, Lingfeng Wu, Si-Qing Chen, and Wayne Xiong. 2024. Developing a reliable, general-purpose hallucination detection and mitigation service: Insights and lessons learned. *arXiv preprint arXiv:2407.15441*.

Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. 2024. Unveiling and harnessing hidden attention sinks: Enhancing large language models without training through attention calibration. *Preprint*, arXiv:2406.15765.

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

Shaolei Zhang, Tian Yu, and Yang Feng. 2024b. Truthx: Alleviating hallucinations by editing large language models in truthful space. *Preprint*, arXiv:2402.17811.

Yuji Zhang, Sha Li, Jiateng Liu, Pengfei Yu, Yi R. Fung, Jing Li, Manling Li, and Heng Ji. 2024c. Knowledge overshadowing causes amalgamated hallucination in large language models. *Preprint*, arXiv:2407.08039.

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

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

A Dataset Details

A.1 CNN/Daily Mail

CNN/Daily Mail (Nallapati et al., 2016) is originally designed for machine-reading and text understanding, the CNN/Daily Mail dataset consists of news articles and their respective highlights, enabling models to practice summarization tasks. It is used by us in training the classifier and evaluation dataset in summarization task. For the training of the classifier, we follow (Chuang et al., 2024a) to randomly sample 1000 data from the testing set. For summarization task evaluation, we sampled another 1000 samples from the testing set.

The dataset uses the Apache-2.0 license and can be found at: https://huggingface.co/datasets/abisee/cnn_dailymail.

A.2 XSum

The Extreme Summarization (XSum) (Narayan et al., 2018) dataset is tailored for generating single-sentence news summaries, presenting a challenge in capturing the main point of an article with minimal context. In this paper, we randomly sampled 1000 data from the testing dataset for evaluation.

The dataset uses the MIT license and can be found at: https://github.com/EdinburghNLP/XSum.

A.3 Natural Questions

Natural Questions (NQ) (Kwiatkowski et al., 2019) is developed by Google and contains real user questions sourced from Google search, paired with Wikipedia article answers. We follow (Chuang et al., 2024a) to construct its openbook variant (NQ-Open) by randomly sampling 1000 data points from a processed dataset (Liu et al., 2023), which can be found athttps://github.com/nelson-liu/lost-in-the-middle. Specifically, the input contexts are created by concatenating three different source documents, where the first and third documents are irrelevant and the second document contains the relevant information.

The original NQ dataset uses the Apache-2.0 license and can be found at: https://github.com/google-research-datasets/natural-questions.

A.4 Trivial QA

Trivial QA (Joshi et al., 2017) is a collection of trivia question-answer pairs with supporting documents from Wikipedia (trivial QA-wiki) or from web search results (trivial QA-web). We randomly sampled 1000 data points from the trivial QA-web. For each data point, we select the source document with the highest score as the context. During the construction, we remove instances whose context length is larger than the base models' context window.

The original dataset uses the Apache-2.0 license and can be found at: https://github.com/mandarjoshi90/triviaqa.

B Models and License

We utilize two open-source models in this paper, both are adopted from their Huggingface Transformer implementation. Llama2-7b (model ID: meta-llama/Llama-2-7b-hf) https://huggingface.co/meta-llama/Llama-2-7b is a 7B parameter, instructional finetuned LLM by Meta, under Llama 2 Community License Agreement. Phi3-mini (Model ID: Microsoft/Phi-3-mini-4k-instruct) https://huggingface.co/microsoft/Phi-3-mini-4k-instruct is a 2B parameter, instructional finetuned LLM by Microsoft.

C Detailed Experimental Setup

C.1 LLM Setting

We adopt greedy search in baseline models and their variants that are applied with GAME. The number of maximum new tokens is set to 256. When using GAME, the LLM generates output in chunks and the chunk size is set to 8.

C.2 Classifier Setting

We utilize 1000 samples from the CNN/Daily Mail dataset to train the classifier. The Llama2-7b model is prompt (see the prompt in Appendix C.3) to generate summarization on these examples. The outputs are divided into chunks with a size of 8 tokens each. We utilize GPT-4o as an annotator as described in Appendix D to create labels for each chunk. The labels as well as the Lookback Ratio of each chunk are used as the training data for a linear logistic regression classifier with scikit-learn[1].

Due to the rarity of hallucination phenomena, the training data is highly imbalanced. To address this, we additionally drew 200 samples to determine the default threshold. Setting this threshold at 0.9

[1] https://scikit-learn.org/1.5/modules/generated/sklearn.linear_model.LogisticRegression.html

```
CNN/Daily Mail: Generate a summary based on the information in the document.
XSum: Generate a summary comprising of 1 sentence for the given article.
NQ-Open: Answer the question based on the information in the document. Explain your reasoning in
the document step-by-step before providing the final answer.
Trivial QA: Answer the question based on the information in the document. Explain your reasoning in
the document step-by-step before providing the final answer.
```

Table 6: System prompts for different datasets used by Llama2-7b and Phi3-mini.

```
You will be provided with a document and a proposed summary. Your task is to determine if the
proposed summary can be directly inferred from the document. If the summary contains any information
not found in the document, it is considered false. Even if the summary is different from a ground
truth summary, it might still be true, as long as it doesn't contain false information.
For each proposed summary, explain why it is true or false based on the information from the
document. Focus only on the original document's content, disregarding any external context.
After your explanation, give your final conclusion as Conclusion: True if the proposed summary is
completely accurate based on the document, or Conclusion: False if it contains any incorrect or
unsupported information. If your conclusion is 'False', identify the exact phrases or name entities
from the summary that is incorrect by stating Problematic Spans: [the inaccurate text spans from
the summary, in Python list of strings format].

#Document#: {document}

#Ground Truth Summary#: {ground_truth_summary}

#Proposed Summary#: {response}

Write your explanation first, and then give your final conclusion as Conclusion: True if
the proposed summary is completely accurate based on the document, or Conclusion: False if it
contains any incorrect or unsupported information. Add Problematic Spans: [the exact inaccurate
text spans from the summary, in a list of strings] if your conclusion is 'False'.
```

Table 7: Prompts for GPT-4o to annotate the span-level hallucinations for given generation from LLMs on summarization tasks (CNN/Daily Mail and XSum).

resulted in the highest test AUROC. The default ϵ in the $\text{sgn}(\cdot)$ function is set to be 1e-4.

C.3 Prompts for Different Datasets

We use the same system prompts for both Llama2-7b and Phi3-mini as shown in Table 6.

D GPT-4o Annotation Prompts

We utilize GPT-4o (model version: gpt-4o-2024-05-13) to annotate hallucinated spans in the generation of LLMs to prepare training data for the classifier. It is also utilized to judge whether contextual hallucination happens in our evaluation on summarization datasets. For both tasks, we adopt the same prompt from (Chuang et al., 2024a), detailed as in Table 7:

E Code Implementation and Computation Resources

The code of the paper is developed based on Huggingface Transformer (4.42.0) `https://github.com/huggingface/transformers` and part of the code is adopted from Lookback Lens `https://github.com/voidism/Lookback-Lens`. The application of GAME requires no training or finetuning of LLMs. All experiments can be run on a single Nvidia Ampere 100 (80GB) GPU. The average inference time per sample in the XSum dataset is around 3 seconds. For other datasets, the time may vary based on the input context length.