Not All Answers Are Contextually Persuadable: Inference Dynamics in Large Language Models under Contextual Influence

Anonymous Authors[1]

Abstract

At the core of modern prompting techniques is contextual sensitivity, the ability of large language models to adapt their predictions based on inference-time context. Despite its central role, inference behavior under strong contextual influence remains poorly understood, particularly at the level of internal inference dynamics. To bridge this gap, we introduce a theoretical framework for analyzing contextual influence through inference dynamics, enabling quantitative characterization of inference behavior beyond output-level answer changes. Our analysis shows that inference dynamics do not exhibit unbounded drift under repeated contextual assertions. Instead, predictive representations converge to stable, query-dependent regimes that fundamentally constrain whether contextual signals can alter a model's prediction. This leads to a surprising finding: *Repeated contextual assertions do not act as accumulating evidence during inference and may therefore fail to alter a model's prediction even under unbounded repetition, while in other cases a prediction change becomes inevitable.* We empirically validate our theoretical predictions across diverse models and tasks, demonstrating strong alignment between theory and observed inference behavior. These contributions offer a principled pathway toward characterizing the limits of contextual influence during inference, and provide practical implications for designing and evaluating repetition-based prompting methods.

1. Introduction

Large language models (LLMs) routinely adjust their predictions in response to contextual information provided at inference time (Sclar et al., 2023; Zhuo et al., 2024; Dong et al., 2024). This behavior underlies a wide range of prompting, in-context learning, and answer-steering methods that deliberately exploit the model's sensitivity to input context (Liu et al., 2023; Vatsal & Dubey, 2024; Sahoo et al., 2024). At the same time, it poses risks in settings where predictions are expected to remain stable under spurious or adversarial cues, including factual question answering, evaluation benchmarks, and safety-critical deployments (Chang et al., 2024; Greshake et al., 2023; Russinovich et al., 2025). Across both settings lies a widely held yet largely unexamined intuition: repeated contextual assertions are implicitly treated as accumulating evidence during inference (Leviathan et al., 2025; Yan et al., 2023; Xu et al., 2024b). Accordingly, reiterating a candidate answer progressively biases the model toward that answer, eventually overwhelming the original query signal. This assumption is often taken for granted in empirical prompting practices and evaluation protocols alike, motivating the expectation that sufficient repetition will always succeed in altering a model's prediction (Hong et al., 2025; Laban et al., 2023; Fastowski & Kasneci, 2024; Geng et al., 2025).

In this work, we show that this intuition fails to capture the underlying inference behavior, and reveal heterogeneous response patterns in LLMs through a controlled single-round setting that isolates repetition as the sole source of contextual signal, as illustrated in Figure 1. Concretely, these

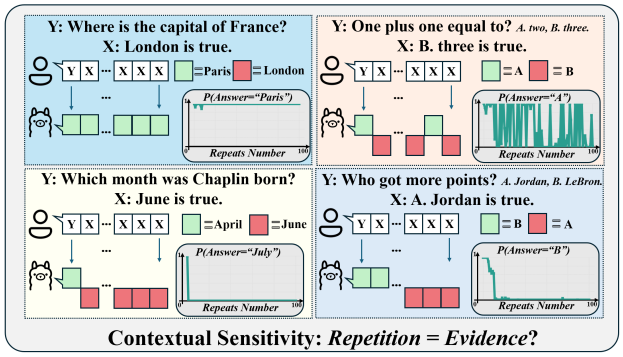

Figure 1. Heterogeneous effects of contextual repetition on model predictions. Illustrative examples showing qualitatively distinct behaviors of LLMs under repeated contextual assertions, including immediate answer flips, delayed changes, early saturation, and complete invariance to repetition.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

patterns range from predictions that flip almost immediately, to predictions that remain stable even under unbounded repetition, as well as cases that change only after substantial repetition or quickly saturate and become insensitive to further repetition. Such heterogeneous behaviors cannot be readily resolved by prior studies, which predominantly examine finite prompt manipulations and observable answer changes (Hong et al., 2025; Laban et al., 2023). This motivates our central question: *What is the asymptotic behavior of LLM inference under unbounded contextual repetition?*

To address this question, we study contextual influence through a formal inference framework that explicitly models repetition as a growing contextual signal. We focus on the same controlled single-round setting as in Figure 1, in which a fixed query is followed by repeated instances of an identical contextual assertion, without introducing any additional evidence, reasoning steps, or multi-turn interaction. By progressively increasing the repetition strength, this setting enables us to probe the asymptotic regime of inference under unbounded contextual repetition, while ruling out confounding effects from additional information. More importantly, it captures a fundamental primitive underlying many prompt-based steering and persuasion techniques (Leviathan et al., 2025; Yan et al., 2023; Xu et al., 2024b), including answer priming, instruction reinforcement, and repetition-based prompting, making it a principled testbed for analyzing the limits of contextual influence during inference.

Our analysis reveals that LLM inference does not exhibit unbounded drift under contextual repetition. Instead, the model's predictive representation along the key discriminative direction converges to a stable, query-dependent limit, reflecting intrinsic saturation effects in the model's internal predictive representations. This convergence behavior varies systematically across questions and models, leading to a surprising finding: *Repeated answer assertions may fail to alter a model's prediction even as repetition grows arbitrarily large, while in other cases a prediction flip becomes inevitable.* These results challenge the common intuition that repeatedly presenting a candidate answer in the input acts as accumulating evidence that progressively steers the model's prediction toward that answer (Leviathan et al., 2025; Yan et al., 2023; Xu et al., 2024b). By contrast, contextual repetition induces bounded and predictable inference dynamics rooted in the model's internal representations, rather than monotonic evidence accumulation. Our contributions can be summarized as follows:

- **Problem Formulation.** We cast contextual influence as a problem of internal inference dynamics and introduce a formal framework for analyzing how repeated contextual signals shape representation-level inference trajectories, moving beyond answer-level prediction changes to a mechanistic understanding of inference behavior.
- **Asymptotic Inference Convergence.** In a controlled single-round inference setting, we establish formal convergence guarantees showing that internal inference trajectories under repeated contextual assertions converge to stable, query-dependent limits rather than exhibiting unbounded drift as repetition grows. This result demonstrates that contextual repetition does *not* function as accumulating evidence during inference.
- **Representation–Prediction Alignment.** We empirically validate the theoretical predictions across models and tasks, demonstrating a tight correspondence between representation-level inference dynamics and observable prediction behavior. Our analysis reveals when prediction changes become inevitable and when they are provably unattainable under unbounded contextual repetition.

2. Related Work

We survey prior work on contextual influence in LLMs, covering both empirical characterizations of prediction sensitivity and efforts to analyze the internal inference mechanisms that underlie such behavior.

2.1. Contextual Sensitivity and Consistency in LLMs

Large language models are highly sensitive to inference-time context. Changes in prompt phrasing, instruction structure, example ordering, or the presence of candidate answers can substantially alter model predictions, even when task semantics are unchanged (Brown et al., 2020; Jiang et al., 2020; Liu et al., 2022; Min et al., 2022; Dong et al., 2024; Bertsch et al., 2025). While such sensitivity underlies effective prompting and in-context learning, it also raises concerns about prediction consistency and robustness under spurious or misleading contextual signals.

A closely related line of work studies contextual sensitivity when input context is implicitly treated as *evidence-like content*. In these settings, models are exposed to repeated assertions, rephrased claims, or redundant contextual signals within a single prompt, without introducing new supporting evidence. Empirical studies show that such repetition can strongly bias model outputs and induce answer instability across question answering, factual reasoning, and multimodal tasks (Yan et al., 2023; Pan et al., 2023; Zhang et al., 2026; Dalal & Misra, 2024). Prior work attributes these effects to surface-level reinforcement or aggregation over redundant content, and notes that repetition often exhibits non-linear influence (Yan et al., 2023; Zhang et al., 2026). Studies on sycophancy, conformity, and multi-turn persuasion show that models may align their responses with user-stated beliefs or sustained interactional pressure, leading to response instability driven by training incentives and dialogue dynamics rather than inference-time repetition (Ouyang et al., 2022; Hong et al., 2025; Sharma et al.,

2023b; Liu et al., 2025; Li et al., 2025; Weng et al., 2025). While powerful, most studies assess contextual sensitivity only through observable prediction changes, such as answer flips under specific prompt manipulations, without examining the underlying inference dynamics.

2.2. Understanding Transformer Inference under Repeated and Long Contexts

Representation-level analyses probe internal states to explain how predictions form across layers, and show that inference is structured with depth-dependent roles and occasional disconnects between encoded information and final outputs (Lad et al., 2024; Gao et al., 2024). Mechanistic studies further localize repetition-related generation failures to specific heads or neurons (Vaidya et al., 2023; Yao et al., 2025; Hiraoka & Inui, 2025). These works offer useful component-level insights, but they are typically conducted under fixed prompts or limited repetition regimes, and therefore do not characterize how the *inference trajectory* of internal representations evolves as a contextual signal is systematically strengthened.

Formal and asymptotic analyses complement representation-level probing by characterizing architectural limits of attention and positional encodings in large-context regimes (Noci et al., 2022; Hayase et al., 2025). Work on relative positional mechanisms (e.g., RoPE variants) reveals structured, sometimes oscillatory, behavior as context length grows (Su et al., 2024; Barbero et al., 2024; Zhong et al., 2025; Gelberg et al., 2026). While these results establish well-defined asymptotic structure in Transformer computation under specific modeling assumptions, they are typically task-agnostic and rarely connect such asymptotic properties to inference-time behavior on concrete prediction tasks.

Overall, existing representation-level and formal analyses illuminate important aspects of internal computation and architectural constraints, but they stop short of explaining how internal inference dynamics *evolve and converge* under progressively amplified contextual signals. Addressing this gap requires analyzing inference as a *trajectory* driven by increasing repetition within a single prompt, which is the focus of our work.

3. Theoretical Analysis of Inference Convergence under Repetition

In this section, we provide theoretical convergence analysis of a standard decoder-only transformer architecture in our problem setting.

Setting and Notations. We use (y, x, z) to denote the token sequences obtained by tokenizing different parts of the prompt: $y = (y_1, \dots, y_m)$ is the tokenized *query/prefix*, i.e., the non-repeated part containing the question and any fixed instructions, $x = (x_1, \dots, x_T)$ is the tokenized *loop template*, i.e., a fixed block repeated verbatim N times, and $z = (z_1, \dots, z_s)$ is the tokenized *suffix* after the repeated region (e.g., an answer cue). The resulting input is of the form $u^{(N)} = y\|x\|x\|\cdots\|x\|z$ where x is repeated N times so the length is $L_N = m + NT + s$, and we choose a target position t_N as a fixed offset into the suffix so it remains aligned to the same suffix token as N varies.

Transformer Representations. We instantiate the computation on $u^{(N)}$ with a decoder-only pre-norm Transformer of depth L and hidden size d_{model}. Let $E_\ell^{(N)}(t) \in \mathbb{R}^{d_{\text{model}}}$ denote the residual stream at position t after layer ℓ when processing $u^{(N)}$, with $E_0^{(N)}(t)$ given by token embeddings (and any positional mechanism used by the model). Each layer applies a pre-norm multi-head self-attention block followed by a pre-norm feed-forward block:

$$\begin{aligned} E_{\ell+\frac{1}{2}}^{(N)}(t) &= E_\ell^{(N)}(t) + \mathrm{Attn}_\ell\big(\mathrm{LN}_\ell^{\mathrm{attn}}(E_\ell^{(N)})\big)(t), \\ E_{\ell+1}^{(N)}(t) &= E_{\ell+\frac{1}{2}}^{(N)}(t) + \mathrm{FFN}_\ell\big(\mathrm{LN}_\ell^{\mathrm{ffn}}(E_{\ell+\frac{1}{2}}^{(N)})\big)(t). \end{aligned} \tag{1}$$

where Attn_ℓ represents a attention block, FFN_ℓ represents a feed-forward block, and LN_ℓ represents a layer normalization map.

RoPE Attention Logits. Suppose Rotary Position Embedding (RoPE) is adopted in the attention block. Specifically, within the attention block Attn_ℓ, for each head h, the per-head vectors lie in $\mathbb{R}^{d_h}$, and the logit between a query at position t and a key at position $p < t$ takes the RoPE form

$$s_{\ell,h}(t,p) \;:=\; \frac{1}{\sqrt{d_h}} \left\langle q'_{\ell,h}(t),\; R\big(-(t-p)\big)\, k'_{\ell,h}(p) \right\rangle, \tag{2}$$

where it depends on the relative distance $t - p$ through a RoPE rotation $R(-(t-p))$ and $q'_{\ell,h}, k'_{\ell,h}$ are the pre-RoPE query/key vectors.

Our **object of interest** is the target representation $E_L^{(N)}(t_N)$ and its layer-wise limits as $N \to \infty$. In this section, we show that as $N \to \infty$, $E_L^{(N)}(t_N)$ converges to a well-defined limit. Informally,

Theorem 3.1 (Main theorem, informal)**.** *Under some mild assumptions,* $\lim_{N\to\infty} E_L^{(N)}(t_N) = E_L^{(\infty)}(t_N)$ *exists and is finite.*

Throughout the paper, we assume the following standard regularity conditions for pre-norm Transformers.

Assumption 3.2 (Regularity of pre-norm Transformer blocks)**.** Suppose that: i) the residual streams are uniformly bounded over all layers and repetition counts on the positions relevant to our analysis, i.e., $\sup_{N,\ell,t} \|E_\ell^{(N)}(t)\| \le C$ for some $C < \infty$; ii) the layer normalization maps $\mathrm{LN}_\ell^{\mathrm{attn}}$ and $\mathrm{LN}_\ell^{\mathrm{ffn}}$ are continuous and Lipschitz on bounded sets; and iii) the feed-forward blocks FFN_ℓ are continuous on bounded sets.

Under Assumption 3.2, the layer-wise update map formed by residual addition together with the normalization and FFN blocks is well-behaved, enabling a layer-wise induction argument in the convergence.

Section 3.1 characterizes the RoPE-induced dependence of a single-head attention logit on the relative distance d, showing that it is a bounded finite trigonometric polynomial (equivalently, a finite exponential expansion). Section 3.2 leverages this bounded oscillatory structure to establish Cesàro limits for the loop-region softmax normalizer and value-weighted numerator, implying a well-defined limiting loop-only attention output. Finally, we lift the head-wise limit to multi-head attention and combine it with continuity of the pre-norm LN/FFN blocks to propagate convergence across layers, yielding Theorem 3.1.

3.1. RoPE Logits Admit a Finite Exponential Expansion

Proposition 3.3 (Finite exponential form and boundedness of RoPE logits)**.** *Suppose that Assumption 3.2 holds. Fix a layer and an attention head of dimension d_h. For fixed $q', k' \in \mathbb{R}^{d_h}$, define*

$$s(d) \;:=\; \frac{1}{\sqrt{d_h}}\,(q')^\top R(-d)\,k', \qquad d \in \mathbb{Z}_{\geq 0}.$$

Then $s(d)$ can be written as a finite trigonometric polynomial in d, equivalently as a finite sum of complex exponentials:

$$s(d) \;=\; c_0 \;+\; \sum_{r\in\mathcal{K}} \gamma_r e^{i\mu_r d},$$

where $\mathcal{K}$ is finite and $\mu_r \in \mathbb{R}$. Moreover, $s(d) \in \mathbb{R}$ is ensured by conjugate-paired terms, and there exists $M_s > 0$ independent of d such that $|s(d)| \leq M_s$ for all $d \geq 0$.

Proposition 3.3 characterizes how a single-head attention logit varies with the relative distance d under RoPE. Specifically, it shows that with fixed layer, head, query, and template key, $s(d)$ is not an arbitrary sequence in d but a uniformly bounded oscillatory signal with only finitely many frequencies. This finite-frequency structure is the key input for our further limit analysis, enabling control of long-run averages of logit-dependent quantities such as Cesàro averages of $\exp(s(d))$ and its value-weighted variants. This in turn leads directly to a well-defined limiting loop-attention output in Section 3.2 (See Appendix B.3).

3.2. Cesàro Limits for Loop Attention Weights

Let t be the target position and let $p < t$ range over loop positions. Define the relative distance

$$d := t - p \in \mathbb{Z}_{\geq 1}.$$

For convenience, we re-index distances so that the loop contributions are represented by $d \in \mathbb{Z}_{\geq 0}$. In the repeated-loop setting, the value vectors are periodic with period T: there exist $v_0, \ldots, v_{T-1} \in \mathbb{R}^{d_h}$ such that

$$v(d+T) = v(d), \qquad \text{for all } d \geq 0.$$

We then define the Cesàro averages:

$$A_L := \frac{1}{L}\sum_{d=0}^{L-1} e^{s(d)}, \qquad B_L := \frac{1}{L}\sum_{d=0}^{L-1} e^{s(d)}\, v(d).$$

Proposition 3.4 (Cesàro limits for the loop normalizer and numerator)**.** *Suppose that Assumption 3.2 holds. Suppose that $v(d)$ is periodic with period T. Then the limits*

$$\mu_p := \lim_{L\to\infty} A_L \in (0,\infty), \qquad \mu_w := \lim_{L\to\infty} B_L \in \mathbb{R}^{d_h}$$

exist. Consequently, the loop-only attention output

$$\bar{v}_L \;:=\; \frac{\sum_{d=0}^{L-1} e^{s(d)} v(d)}{\sum_{d=0}^{L-1} e^{s(d)}} \;=\; \frac{B_L}{A_L}$$

converges to the finite limit

$$a^{(\infty)} \;:=\; \lim_{L\to\infty} \bar{v}_L \;=\; \frac{\mu_w}{\mu_p} \in \mathbb{R}^{d_h}.$$

Proposition 3.4 shows that, under the finite-frequency bounded logit structure of $s(d)$ in Proposition 3.3 and the periodicity of the loop values $v(d)$, the Ces‘aro limits of the softmax normalizer and the value-weighted numerator exist. As a result, the loop-only attention output converges to the well-defined limit $a^{(\infty)} = \mu_w/\mu_p$. This limiting characterization implies that the loop-region contribution to attention at the target stabilizes head-wise and therefore remains stable after multi-head aggregation, which is the key ingredient for establishing the layer-wise convergence theorem (See Appendix B.4).

3.3. From Loop-Only to Full Attention and Layer-Wise Convergence

This section lifts Proposition 3.4 from loop-only attention to the full Transformer at the target. Using Proposition 3.3 to bound unnormalized weights, we show non-loop tokens become negligible, lift head-wise limits to a single attention.

Adding Finitely Many Non-Loop Tokens. At the target t_N, attention ranges over NT loop tokens and only $O(1)$ non-loop tokens from the prefix/suffix. By Proposition 3.3, logits are uniformly bounded, hence each unnormalized weight $\exp(\text{logit})$ is bounded above and below by positive constants. Therefore, the total loop contribution to the softmax normalizer scales as $\Theta(NT)$ while the total non-loop contribution is $O(1)$, so the softmax mass on non-loop tokens vanishes as $N \to \infty$. Consequently, the head output converges to the same limit $a^{(\infty)}$ given by Proposition 3.4.

Theorem 3.5 (Layer-wise convergence for attention block)**.** *Suppose that Assumption 3.2 holds. Then for every $\ell \in \{0, 1, \ldots, L\}$, the limit*

$$E_\ell^{(\infty)}(t_N) := \lim_{N\to\infty} \mathrm{Attn}_\ell\big(\mathrm{LN}_\ell^{\mathrm{attn}}(E_\ell^{(N)})\big)(t_N)$$

exists and is finite, (See Appendix B.5).

3.4. Extension to Transformer with Multi-Head Attention, FFN, and Multiple Layers

In this section, we further extend our previous discussion to a full transformer architecture. This is achieved by applying Theorem 3.5 and Equation (1) to propagate convergence across layers.

Multi-Head Attention. Applying the head-wise argument independently to each head yields limiting head outputs $a_{\ell,1}^{(\infty)}, \ldots, a_{\ell,H}^{(\infty)}$. Concatenating and applying the output projection W_O gives a well-defined limit for the attention residual update at the target:

$$A_\ell^{(N)}(t_N) \to A_\ell^{(\infty)}(t_N) := W_O\big[a_{\ell,1}^{(\infty)}; \ldots; a_{\ell,H}^{(\infty)}\big].$$

Theorem 3.6 (Layer-wise convergence for transformer)**.** *Suppose that Assumption 3.2 holds. Then for every $\ell \in \{0, 1, \ldots, L\}$, the limit*

$$E_\ell^{(\infty)}(t_N) := \lim_{N\to\infty} E_\ell^{(N)}(t_N)$$

exists and is finite. In particular, $\lim_{N\to\infty} E_L^{(N)}(t_N) = E_L^{(\infty)}(t_N)$.

Theorem 3.1 formalizes that repeating a loop block arbitrarily many times does not cause the target representation to drift without bound. Instead, the representation converges to a stable, query-dependent limiting regime determined by RoPE-induced oscillatory logits and Cesàro averaging, rather than by accumulation of evidence through repetition (See Appendix B.6).

4. Quantifying Answer Shifts under Contextual Repetition

In Section 4, we formalize and estimate *answer shifts* under contextual repetition, connecting the convergence of inference dynamics to changes in model answer preferences.

4.1. Answer-Shift Metric in Latent Space

Let $W_{\mathrm{out}} \in \mathbb{R}^{|\mathcal{V}|\times d_{\mathrm{model}}}$ be the output head. Fix a token $e_b \in \mathcal{V}$ as the model's *reference answer* for the query, and a token $e_a \in \mathcal{V}$ as the *bias target* encouraged by the repeated loop. Define $\Delta_{e_a,e_b} := W_{\mathrm{out}}[e_a] - W_{\mathrm{out}}[e_b] \in \mathbb{R}^{d_{\mathrm{model}}}$ and, for any layer ℓ, $g_\ell^{(N)}(e_a, e_b) := \langle \Delta_{e_a,e_b}, E_\ell^{(N)}(t_N)\rangle$. We then use $g_\ell^{(N)}(e_a, e_b)$ as an *answer-shift metric*, whose sign indicates the direction of the shift.

4.2. Estimating Infinite-Repetition Limit

Next, we estimate the asymptotic answer shift $\widehat{g}_L^{(\infty)}(e_a, e_b)$ by estimating the limiting representation $E_L^{(\infty)}(t_N)$ at the prediction position. Based on the theorem 3.4, we approximate the $N \to \infty$ regime from a *single* forward pass at a large finite N. Intuitively, since the RoPE logit in (2) depends on a past position p only through the relative distance $d = t_N - p$ which represents the RoPE angle, we fix the last-loop templates and sample earlier loop copies to approximate the long-run average.

Fix a layer ℓ and head h and let $\{p_1^{\mathrm{last}}, \ldots, p_T^{\mathrm{last}}\}$ be the absolute token indices of the *last* loop block. For $m \in \{0, 1, \ldots, N-1\}$, where $m = 0$ is the last block, we define $p_{m,j} := p_j^{\mathrm{last}} - mT$ and the corresponding last-loop value template $v_{\ell,h,j}^{\mathrm{tpl}} := v_{\ell,h}(p_j^{\mathrm{last}}) \in \mathbb{R}^{d_h}$. We sample i.i.d. pairs (m_r, j_r) uniformly from $\{0, \ldots, N-1\} \times \{1, \ldots, T\}$ for $r = 1, \ldots, D$ and form the Monte Carlo estimates of the loop contribution: $\widehat{Z}_{\mathrm{loop}} := \frac{NT}{D}\sum_{r=1}^{D} \exp\big(s_{\ell,h}(t_N, p_{m_r,j_r})\big)$ and $\widehat{N}_{\mathrm{loop}} := \frac{NT}{D}\sum_{r=1}^{D} \exp\big(s_{\ell,h}(t_N, p_{m_r,j_r})\big)\, v_{\ell,h,j_r}^{\mathrm{tpl}}$.

In addition to the NT loop tokens, attention at t_N includes only finitely many non-loop indices $i < t_N$ from y and z. Let $\mathcal{I}_{\mathrm{nonloop}}(t_N)$ collect these indices, we add their exact contribution:$Z_{\mathrm{nonloop}} := \sum_{i\in\mathcal{I}_{\mathrm{nonloop}}(t_N)} \exp\big(s_{\ell,h}(t_N, i)\big)$ and $N_{\mathrm{nonloop}} := \sum_{i\in\mathcal{I}_{\mathrm{nonloop}}(t_N)} \exp\big(s_{\ell,h}(t_N, i)\big)\, v_{\ell,h}(i)$.

Combining the loop estimate with the exact non-loop correction yields the head-wise estimator $\widehat{a}_{\ell,h}^{(\infty)}(t_N) := \frac{N_{\mathrm{nonloop}} + \widehat{N}_{\mathrm{loop}}}{Z_{\mathrm{nonloop}} + \widehat{Z}_{\mathrm{loop}}} \in \mathbb{R}^{d_h}$, and after concatenating over heads and applying the output projection, we have

$$\widehat{A}_\ell^{(\infty)}(t_N) := W_{O,\ell}\big[\widehat{a}_{\ell,1}^{(\infty)}(t_N); \ldots; \widehat{a}_{\ell,H}^{(\infty)}(t_N)\big] \in \mathbb{R}^{d_{\mathrm{model}}}.$$

Using the same pre-norm update form as in (1), we then define $\widehat{F}_\ell^{(\infty)}(t_N) := \mathrm{FFN}_\ell\Big(\mathrm{LN}_\ell^{\mathrm{ffn}}\big(E_\ell^{(N)}(t_N) + \widehat{A}_\ell^{(\infty)}(t_N)\big)\Big)$, and aggregate via the residual telescope: $\widehat{E}_L^{(\infty)}(t_N) := E_0^{(N)}(t_N) + \sum_{\ell=0}^{L-1} \widehat{A}_\ell^{(\infty)}(t_N) + \sum_{\ell=0}^{L-1} \widehat{F}_\ell^{(\infty)}(t_N)$. Finally, we estimate the limiting answer shift by $\widehat{g}_L^{(\infty)}(e_a, e_b) := \langle \Delta_{e_a,e_b}, \widehat{E}_L^{(\infty)}(t_N)\rangle$. Using the residual telescope, we decompose it into attention/FFN contributions across layers:

$$\widehat{g}_L^{(\infty)}(e_a, e_b) = \widehat{g}_0^{(\infty)}(e_a, e_b) + \sum_{\ell=0}^{L-1} \widehat{g}_{\ell,A}^{(\infty)}(e_a, e_b) + \sum_{\ell=0}^{L-1} \widehat{g}_{\ell,F}^{(\infty)}(e_a, e_b). \tag{3}$$

5. Experiments

In this section, we empirically validate our theoretical analysis by examining representation-level convergence and answer-level behavior across various settings. Implementation details are presented in Appendix C.

Table 1. Layer-wise convergence of inference dynamics at $N = 1000$ on three benchmarks at various divergence levels $(0.1, 0.05, 0.01)$.

Model	Dataset																	
	Openbook QA						MINTAKA						Simple QA					
	0.1		0.05		0.01		0.1		0.05		0.01		0.1		0.05		0.01	
	Attn	FFN	Attn	FFN	Attn	FFN	Attn	FFN	Attn	FFN	Attn	FFN	Attn	FFN	Attn	FFN	Attn	FFN
Falcon3-7B-Base	98.1	97.3	95.3	94.3	81.5	81.3	98.7	99.0	96.6	97.4	78.6	80.7	100	99.9	99.9	99.4	84.7	86.3
Mistral-7B-v0.1	97.5	97.5	93.9	96.0	78.9	90.9	94.0	93.7	89.1	89.1	66.3	68.7	95.5	94.9	92.9	91.5	72.8	73.2
Apollo-1-4B	99.0	99.3	95.5	96.3	79.7	78.5	99.8	99.9	98.7	99.4	87.0	91.4	100	100	99.9	99.9	93.0	93.9
Qwen3-4B	99.5	99.4	97.0	97.5	83.8	84.4	99.5	99.7	98.6	99.0	86.8	90.4	100	100	99.9	99.9	92.7	94.1
Qwen2.5-1.5B	93.0	87.0	85.3	76.5	47.6	33.9	99.8	99.6	97.0	97.6	73.6	69.1	99.8	99.7	99.0	98.5	75.8	68.7
Falcon3-3B-Base	99.0	98.5	97.0	97.0	86.7	88.1	100	100	99.8	99.7	93.3	94.0	100	100	100	100	95.2	93.8

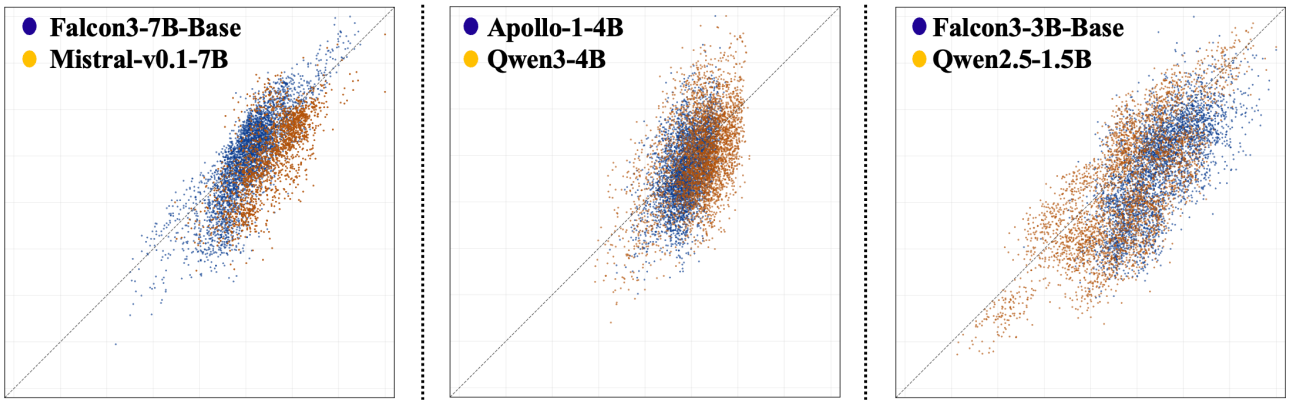

Figure 2. Comparison between representation-level predicted answer shifts and forward-computed output-level shifts across multiple models on the Openbook QA dataset at $N = 1000$. Each point corresponds to a query-answer pair, and the dashed line indicates perfect agreement. The strong diagonal alignment indicates high quantitative fidelity of representation-level predictions.

5.1. Experimental Setup

Benchmarks. We evaluate on seven benchmark subsets: OpenBookQA (Mihaylov et al., 2018) (closed, 1,769 QA), MINTAKA (Sen et al., 2022) (open, 370 QA) and SimpleQA (Wei et al., 2024) (open, 191 QA) to evaluate our study. Moreover, we include subsets from SYCON-Bench (Hong et al., 2025), Farm (Xu et al., 2024a), BeHonest (Chern et al.) and sycophancy-eval (Sharma et al., 2023a) to further validate our alignment with prior studies.

Models. We evaluate six LLMs, including Falcon3-7B-Base[1] (Team, 2024b), Mistral-7B-v0.1[2] (Jiang et al., 2023), Apollo-1-4B[3] (Research, 2025), Qwen3-4B[4] (Yang et al., 2025), Qwen2.5-1.5B[5] (Team, 2024a) and Falcon3-3B-Base[6] (Team, 2024b), covering three model families (Qwen, Mistral and TII). These models span large (>6B), medium (3B–6B) and small (<3B) parameter scales, enabling analysis across various families and capacities.

[1] https://huggingface.co/tiiuae/Falcon3-7B-Base

[2] https://huggingface.co/mistralai/Mistral-7B-v0.1

[3] https://huggingface.co/Loom-Labs/Apollo-1-4B

[4] https://huggingface.co/Qwen/Qwen3-4B

[5] https://huggingface.co/Qwen/Qwen2.5-1.5B

5.2. Representation-Level Convergence of Inference Dynamics

Layer-wise Stability at Large Repetition Length. We first examine the layer-wise stability of representation-level inference dynamics, a key empirical implication of our main Theorem 3.6. In Table 1, we report the fraction of layers whose answer-shift dynamics have effectively converged at $N = 1000$, measured separately for attention and feed-forward components across both closed-form (Openbook QA) and open-form datasets (MINTAKA and Simple QA). Convergence is determined using a tail-based criterion (detailed in Appendix C) that compares residual variation against the total shift magnitude, under multiple tolerance levels $(0.1, 0.05, 0.01)$. Across different evaluation regimes, we observe consistent layer-wise stability across various

[6] https://huggingface.co/tiiuae/Falcon3-3B-Base

Table 2. Evaluation of predicting answer preference changes under repetition across the TRANSFER, CORRECT, and MISLEAD regimes at different finite repetition lengths ($N \in \{500, 750, 1000\}$).

Model	Setting																	
	Transfer						Mislead						Correct					
	500		750		1000		500		750		1000		500		750		1000	
	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1
Falcon3-7B-Base	96.1	98.0	92.5	96.1	86.8	92.6	93.3	96.5	91.5	95.5	84.7	91.6	85.6	92.2	80.8	89.3	75.2	85.4
Mistral-7B-v0.1	94.4	97.1	92.8	96.3	88.7	94.0	88.5	93.9	85.0	91.8	79.8	88.6	82.5	90.4	81.5	89.7	74.6	85.0
Apollo-1-4B	81.2	89.6	83.4	91.1	90.7	95.1	71.7	82.7	71.4	82.5	80.1	88.6	70.0	82.3	79.7	88.7	78.2	87.8
Qwen3-4B	84.8	91.8	84.4	91.5	89.6	94.5	82.0	89.2	79.1	87.1	85.9	91.9	88.3	90.8	90.8	87.5	93.3	93.7
Qwen2.5-1.5B	88.8	91.7	87.8	91.6	89.2	92.4	76.3	63.4	72.1	59.8	71.2	59.7	72.2	22.2	63.7	20.1	98.8	99.4
Falcon3-3B-Base	67.9	80.1	73.2	84.5	71.8	83.5	83.6	91.0	83.2	90.7	78.5	87.8	91.0	95.3	86.0	92.4	78.4	87.8
Mean	**85.5**	**91.5**	**85.7**	**91.8**	**86.1**	**92.0**	**82.5**	**86.1**	**80.4**	**84.6**	**80.0**	**84.6**	**80.8**	**78.9**	**79.9**	**78.9**	**82.2**	**89.8**

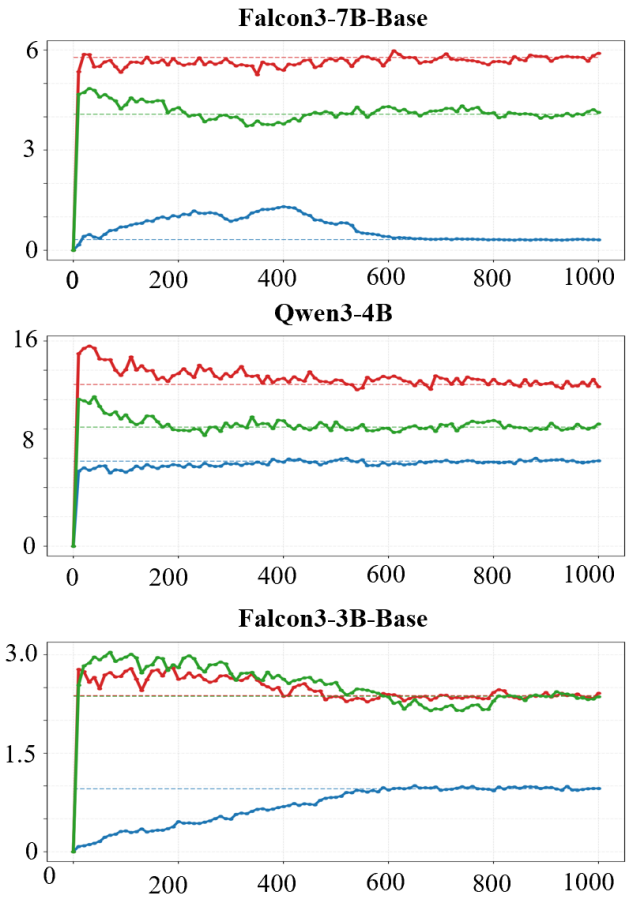

Figure 3. KL divergence trajectories of next-token predictive distributions under repetition. For each model, the KL divergence decreases and stabilizes as the repetition length increases, demonstrating convergence of answer-level predictive distributions under contextual repetition.

settings, with a large fraction of layers exhibiting negligible residual variation at the large but finite repetition length examined here. This pattern holds for both attention and feed-forward blocks and persists under increasingly strict tolerances, supporting our finding that representation-level inference dynamics under repetition approach a stable limiting regime rather than continuing to drift.

Quantitative Fidelity of Answer-Shift Predictions. Beyond stability, we further evaluate the quantitative fidelity of our representation-level predictions against the actual output-level answer shifts produced by standard forward inference. Figure 2 compares the representation-level estimator $\widehat{g}(e_a, e_b)$ (defined in Equation 3) with the forward-computed answer shift $g^{(N)}(e_a, e_b)$ at the same repetition length across multiple models on the Openbook QA dataset. Across model scales, the two quantities exhibit strong numerical agreement, with points closely concentrated along the diagonal. This alignment indicates that our representation-level analysis yields quantitatively accurate predictions of output-level answer shifts, even at finite repetition lengths.

5.3. Predicting Asymptotic Answer Changes

Answer-Level Convergence of Predictive Distributions. We begin by examining whether answer-level predictive distributions exhibit stable behavior under contextual repetition. Specifically, we track how the model's next-token predictive distribution at the target position evolves as the repetition length increases. For pairs of repetition lengths, we compute the KL divergence between the corresponding next-token distributions and analyze how this divergence changes as repetition grows. Figure 3 shows representative KL-divergence trajectories as a function of the repetition length. Across models and examples, the divergence decreases and eventually plateaus, indicating that the next-token predictive distribution stabilizes at the answer level. This stabilization mirrors the representation-level behavior observed in Section 5.2 and suggests that repetition induces a well-defined limiting answer distribution rather than continuing to perturb the model's output indefinitely.

Table 3. Evaluation of model-level stability on established sycophancy and persuasion benchmarks.

Model	Dataset							
	SYCON		Farm		BeHonest		SycoEval	
	Amp	Ours	Amp	Ours	Amp	Ours	Amp	Ours
Falcon3-7B-Base	0.57	3.26	1.05	3.33	1.06	1.79	0.74	1.95
Mistral-7B-v0.1	1.00	3.63	1.21	4.93	1.06	2.00	1.08	4.52
Apollo-1-4B	1.00	3.15	1.00	4.18	1.02	2.55	1.00	4.68
Qwen3-4B	1.06	6.28	1.02	7.31	1.07	4.09	1.24	8.42
Qwen2.5-1.5B	4.25	0.13	1.03	3.22	1.07	0.62	2.67	0.59
Falcon3-3B-Base	∞	0.49	1.03	3.64	1.07	0.60	2.57	0.42

Predicting Answer Preference Shifts via Representation-Level Estimates. We next evaluate whether representation-level predictions can be used to reliably predict answer-level preference changes. We consider three regimes, including TRANSFER, CORRECT, and MISLEAD, which characterize different repetition-induced transitions between original and alternative answers. Specifically, TRANSFER involves shifts between incorrect answers, CORRECT shifts from incorrect to correct answers, and MISLEAD shifts from correct to incorrect answers. For each regime, we evaluate fixed repetition lengths $N \in \{500, 750, 1000\}$. At each length, we compute the representation-level predicted answer shift $\widehat{g}^{(N)}(e_a, e_b)$ and use its sign to predict whether repetition induces a preference change, and then evaluate this prediction against the observed forward behavior.

Table 2 reports the resulting accuracy and F1 scores across models and regimes. Across all regimes, our predicted preference shifts are highly consistent with the actual answer changes observed under repetition. This consistency holds across multiple finite repetition lengths, indicating that the predictive signal is stable rather than sensitive to a particular choice of N. These results demonstrate that our approach enables reliable prediction of repetition-induced answer-level behavior from representation-space analysis, indicating its potential utility for monitoring and analyzing repetition effects in practice.

5.4. Model-Level Alignment with Prior Work

We situate our findings within the existing literature by evaluating our framework on four benchmarks commonly used in sycophancy and persuasion: SYCON, Farm, BeHonest, and SycoEval. Our goal is to assess whether the stability patterns revealed by our analysis align with established empirical observations. For each model and dataset, we report the model-level average of the predicted answer-shift magnitude $\widehat{g}(e_a, e_b)$ (Equation 3), where smaller values indicate greater stability under repetition. We also report the amplification metric (Amp), defined as MR@5/MR@1, following the evaluation protocol introduced in (Xu et al., 2024a). Table 3 shows that, except for a small number of gray-highlighted cases, our representation-based stability measure exhibits trends broadly consistent with prior evaluations. Importantly, our framework provides a mechanistic account of *why* these effects arise and *when* they saturate, are delayed, or fail to materialize entirely, complementing existing evaluations beyond answer-level outcome metrics.

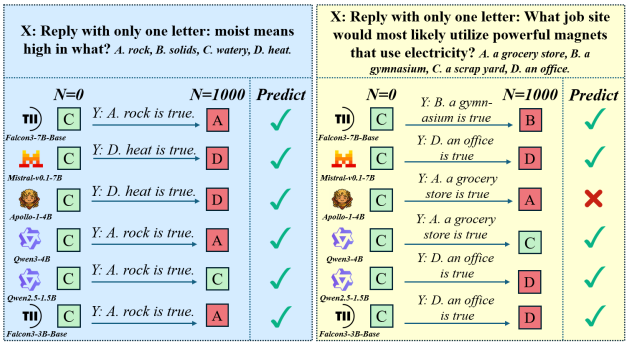

Figure 4. Case studies of repetition-induced answer changes and our predictions. For each query, we show the model's prediction at N=0 and N=1000, alongside our predicted outcome based on representation-space analysis. Checkmarks indicate correct predictions, illustrating that our method can reliably anticipate repetition effects across models.

5.5. Case Studies

Finally, we present a case study to illustrate the practical value of our framework in explaining repetition effects in large language models. Figure 4 shows two representative examples in which repeated contextual assertions induce answer changes under repetition. Across models, our method correctly anticipates whether repetition will change the model's answer, even when the final outcome differs substantially across architectures. These examples show that our framework not only explains repetition effects, but also predicts when repetition will succeed, fail, or saturate, offering actionable guidance for prompting and evaluation.

6. Conclusion

We introduce a theoretical framework for quantitatively analyzing contextual influence through internal inference dynamics, moving beyond answer-level phenomena to representation-level behavior. Our study establishes convergence of representation-level inference under unbounded contextual repetition, providing a mechanistic explanation for when repetition succeeds, fails, or saturates. Building on this insight, our framework enables accurate prediction of answer-level behavior at finite repetition lengths, offering practical guidance for prompt design, evaluation robustness, and safety analysis. Through extensive experiments, we reconcile diverse empirical observations of repetition effects and show that our framework complements existing evaluation practices with a principled, mechanistic perspective.

Impact Statement

This paper presents theoretical and empirical work aimed at advancing the understanding of inference dynamics in large language models. By providing a mechanistic analysis of how contextual repetition influences model behavior, our work contributes to improved interpretability, evaluation robustness, and reliability of existing models. We do not introduce new model architectures or deployment mechanisms, and we do not foresee direct negative societal consequences arising from this work. Potential downstream impacts are consistent with those commonly associated with advances in machine learning research, including applications to safer prompting, more reliable evaluation protocols, and better understanding of model behavior under manipulation.

References

Barbero, F., Vitvitskyi, A., Perivolaropoulos, C., Pascanu, R., and Veličković, P. Round and round we go! what makes rotary positional encodings useful? *arXiv preprint arXiv:2410.06205*, 2024.

Bertsch, A., Ivgi, M., Xiao, E., Alon, U., Berant, J., Gormley, M. R., and Neubig, G. In-context learning with long-context models: An in-depth exploration. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 12119–12149, 2025.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., et al. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45, 2024.

Chern, S., Hu, Z., Yang, Y., Chern, E., Guo, Y., Jin, J., Wang, B., and Liu, P. Behonest: Benchmarking honesty in large language models. *arXiv preprint arXiv:2406.13261*. URL https://arxiv.org/abs/2406.13261.

Dalal, S. and Misra, V. Beyond the black box: A statistical model for llm reasoning and inference. *arXiv preprint arXiv:2402.03175*, 2024.

Dong, Q., Li, L., Dai, D., Zheng, C., Ma, J., Li, R., Xia, H., Xu, J., Wu, Z., Chang, B., et al. A survey on in-context learning. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 1107–1128, 2024.

Fastowski, A. and Kasneci, G. Understanding knowledge drift in llms through misinformation. In *International Workshop on Discovering Drift Phenomena in Evolving Landscapes*, pp. 74–85. Springer, 2024.

Gao, M., Lu, T., Yu, K., Byerly, A., and Khashabi, D. Insights into llm long-context failures: when transformers know but don't tell. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7611–7625, 2024.

Gelberg, Y., Eguchi, K., Akiba, T., and Cetin, E. Extending the Context of Pretrained LLMs by Dropping their Positional Embeddings. Technical report, Sakana AI, January 2026. Technical Report.

Geng, J., Chen, H., Liu, R., Ribeiro, M. H., Willer, R., Neubig, G., and Griffiths, T. L. Accumulating context changes the beliefs of language models. *arXiv preprint arXiv:2511.01805*, 2025.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., and Fritz, M. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM workshop on artificial intelligence and security*, pp. 79–90, 2023.

Hayase, T., Collins, B., and Karakida, R. Gaussian equivalence for self-attention: Asymptotic spectral analysis of attention matrix. *arXiv preprint arXiv:2510.06685*, 2025.

Hiraoka, T. and Inui, K. Repetition neurons: How do language models produce repetitions? In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 483–495, 2025.

Hong, J., Byun, G., Kim, S., and Shu, K. Measuring sycophancy of language models in multi-turn dialogues. *arXiv preprint*, 2025. arXiv:2505.23840.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. Mistral 7b, 2023.

Jiang, Z., Xu, F. F., Araki, J., and Neubig, G. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020.

Laban, P., Murakhovs' ka, L., Xiong, C., and Wu, C.-S. Are you sure? challenging LLMs leads to performance drops in the flipflop experiment. *arXiv preprint arXiv:2311.08596*, 2023.

Lad, V., Lee, J. H., Gurnee, W., and Tegmark, M. The remarkable robustness of LLMs: Stages of inference? *arXiv preprint arXiv:2406.19384*, 2024.

Leviathan, Y., Kalman, M., and Matias, Y. Prompt repetition improves non-reasoning llms. *arXiv preprint arXiv:2512.14982*, 2025.

Li, Y., Miao, Y., Ding, X., Krishnan, R., and Padman, R. Firm or fickle? evaluating large language model consistency in sequential interactions. *arXiv preprint arXiv:2503.22353*, 2025.

Liu, J., Shen, D., Zhang, Y., Dolan, W. B., Carin, L., and Chen, W. What makes good in-context examples for gpt-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd workshop on knowledge extraction and integration for deep learning architectures*, pp. 100–114, 2022.

Liu, J., Jain, A., Takuri, S., Vege, S., Akalin, A., Zhu, K., O'Brien, S., and Sharma, V. Truth decay: Quantifying multi-turn sycophancy in language models. *arXiv preprint*, 2025. arXiv:2503.11656.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys*, 55(9):1–35, 2023.

Mihaylov, T., Clark, P., Khot, T., and Sabharwal, A. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*, 2018.

Min, S., Lyu, X., Holtzman, A., Artetxe, M., Lewis, M., Hajishirzi, H., and Zettlemoyer, L. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

Noci, L., Anagnostidis, S., Biggio, L., Orvieto, A., Singh, S. P., and Lucchi, A. Signal propagation in transformers: Theoretical perspectives and the role of rank collapse. *Advances in Neural Information Processing Systems*, 35: 27198–27211, 2022.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Pan, Y., Pan, L., Chen, W., Nakov, P., Kan, M.-Y., and Wang, W. On the risk of misinformation pollution with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 1389–1403, 2023.

Research, N. Apollo-1-4b. https://huggingface.co/NoemaResearch/Apollo-1-4B, 2025.

Russinovich, M., Salem, A., and Eldan, R. Great, now write an article about that: The crescendo multi-turn LLM jailbreak attack. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 2421–2440, 2025.

Sahoo, P., Singh, A. K., Saha, S., Jain, V., Mondal, S., and Chadha, A. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*, 2024.

Sclar, M., Choi, Y., Tsvetkov, Y., and Suhr, A. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *arXiv preprint arXiv:2310.11324*, 2023.

Sen, P., Aji, A. F., and Saffari, A. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 1604–1619, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.138.

Sharma, M., Tong, M., Korbak, T., Duvenaud, D., Askell, A., Bowman, S. R., Cheng, N., Durmus, E., Hatfield-Dodds, Z., Johnston, S. R., Kravec, S., Maxwell, T., McCandlish, S., Ndousse, K., Rausch, O., Schiefer, N., Yan, D., Zhang, M., and Perez, E. Towards understanding sycophancy in language models, 2023a.

Sharma, M., Tong, M., Korbak, T., Duvenaud, D., Askell, A., Bowman, S. R., Cheng, N., Durmus, E., Hatfield-Dodds, Z., Johnston, S. R., et al. Towards understanding sycophancy in language models. *arXiv preprint*, 2023b. arXiv:2310.13548.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Team, Q. Qwen2.5: A party of foundation models, September 2024a. URL https://qwenlm.github.io/blog/qwen2.5/.

Team, T. Falcon 3 family of open foundation models, December 2024b.

Vaidya, A., Turek, J., and Huth, A. Humans and language models diverge when predicting repeating text. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pp. 58–69, 2023.

Vatsal, S. and Dubey, H. A survey of prompt engineering methods in large language models for different NLP tasks. *arXiv preprint arXiv:2407.12994*, 2024.

Wei, J., Karina, N., Chung, H. W., Jiao, Y. J., Papay, S., Glaese, A., Schulman, J., and Fedus, W. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Weng, Z., Jin, X., Jia, J., and Zhang, X. Foot-in-the-door: A multi-turn jailbreak for llms. *arXiv preprint arXiv:2502.19820*, 2025.

Xu, R., Lin, B., Yang, S., Zhang, T., Shi, W., Zhang, T., Fang, Z., Xu, W., and Qiu, H. The earth is flat because...: Investigating LLMs' belief towards misinformation via persuasive conversation. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16259–16303, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.858. URL https://aclanthology.org/2024.acl-long.858/.

Xu, R., Lin, B., Yang, S., Zhang, T., Shi, W., Zhang, T., Fang, Z., Xu, W., and Qiu, H. The earth is flat because...: Investigating llms' belief towards misinformation via persuasive conversation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16259–16303, 2024b.

Yan, J., Xu, J., Song, C., Wu, C., Li, Y., and Zhang, Y. Understanding in-context learning from repetitions. *arXiv preprint arXiv:2310.00297*, 2023.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yao, J., Yang, S., Xu, J., Hu, L., Li, M., and Wang, D. Understanding the repeat curse in large language models from a feature perspective. *arXiv preprint arXiv:2504.14218*, 2025.

Zhang, C., Ding, W., Liu, J., Wu, M., Wu, Q., and Mooney, R. Do images speak louder than words? investigating the effect of textual misinformation in VLMs. *arXiv preprint*, 2026. arXiv:2601.19202.

Zhong, M., Zhang, C., Lei, Y., Liu, X., Gao, Y., Hu, Y., Chen, K., and Zhang, M. Understanding the RoPE extensions of long-context LLMs: An attention perspective. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 8955–8962, 2025.

Zhuo, J., Zhang, S., Fang, X., Duan, H., Lin, D., and Chen, K. Prosa: Assessing and understanding the prompt sensitivity of llms. *arXiv preprint arXiv:2410.12405*, 2024.

A. RoPE rotation

Let the head dimension be d_h. RoPE acts on the first d_{rope} coordinates of a vector $x \in \mathbb{R}^{d_h}$ via paired 2D rotations, and leaves the remaining coordinates unchanged.

More concretely, let d_{rope} be even, and for $i = 0, \ldots, d_{\text{rope}}/2 - 1$, define frequencies $\omega_i > 0$. For a position index $p \in \mathbb{Z}$, the RoPE operator $R(p)$ acts as

$$R(p)x = \Big(\begin{bmatrix} x_{2i}\cos(\omega_i p) - x_{2i+1}\sin(\omega_i p) \\ x_{2i}\sin(\omega_i p) + x_{2i+1}\cos(\omega_i p) \end{bmatrix}_{i=0}^{d_{\text{rope}}/2-1}, \; x_{d_{\text{rope}}}, \ldots, x_{d_h-1} \Big).$$

Thus $R(p)$ is an orthogonal map on $\mathbb{R}^{d_h}$ and preserves the Euclidean norm: $\|R(p)x\|_2 = \|x\|_2$.

B. Theoretical Results and Proof

In this section, we provide proof for results in Section 3. As in the main paper, we first consider a single layer and a single attention head to analyze the contribution of repeated loop blocks to the attention output at the target position. Then we handle multi-head attention and FFN.

B.1. Relative-position formulation of logits

Consider a single attention head. At the target position t_N in layer ℓ, after pre-normalization, we have a query

$$q_{t_N} = R(t_N) W_q E_\ell^{(N)}(t_N) \in \mathbb{R}^{d_h}.$$

For each token in the loop template $(x_1, \ldots, x_T)$ at absolute position p,

$$k_p = R(p) W_k E_\ell^{(N)}(p).$$

The relative distance (for causal attention) is

$$d = t_N - p > 0.$$

A key property of RoPE is that it can be rewritten in terms of *relative* position. Let $q' = W_q E_\ell^{(N)}(t_N)$ and $k'_p = W_k E_\ell^{(N)}(p)$ such that the logit between the target query and the key at position p satisfies

$$\frac{1}{\sqrt{d_h}} q_{t_N}^\top k_p = \frac{1}{\sqrt{d_h}} (q')^\top R(-d)\, k'_p.$$

We henceforth write the unnormalized logit as a function of d:

$$s(d) = \frac{1}{\sqrt{d_h}} (q')^\top R(-d)\, k'_p.$$

B.2. Trigonometric polynomial structure of $s(d)$

Write

$$(q'_{2i}, q'_{2i+1}) = (a_i, b_i), \quad (k'_{p,2i}, k'_{p,2i+1}) = (c_i, d_i),$$

for $i = 0, \ldots, d_{\text{rope}}/2 - 1$, and let the remaining components be collected into a vector r. Then

$$R(-d)k'_p = \big(u_i \cos(\omega_i d) + v_i \sin(\omega_i d),\, -v_i \cos(\omega_i d) + u_i \sin(\omega_i d)\big)_{i=0}^{d_{\text{rope}}/2-1},\; r.$$

Therefore the contribution to $s(d)$ from the $(2i, 2i+1)$ coordinates is

$$a_i\big(u_i \cos(\omega_i d) + v_i \sin(\omega_i d)\big) + b_i\big(-v_i \cos(\omega_i d) + u_i \sin(\omega_i d)\big) = \alpha_i \cos(\omega_i d) + \beta_i \sin(\omega_i d),$$

for some coefficients α_i, β_i depending on a_i, b_i, u_i, v_i but *independent of* d.

Adding also the non-rotated coordinates (an affine constant term), we obtain

$$s(d) = c_0 + \sum_{k=1}^{M} \alpha_k \cos(\lambda_k d) + \beta_k \sin(\lambda_k d),$$

for some finite M, coefficients $c_0, \{\alpha_k, \beta_k\}$, and frequencies $\lambda_k = \omega_{i(k)}$. That is, For fixed query q' and key template k'_p, the logit $s(d)$ is a trigonometric polynomial in the discrete variable d.

B.3. Proof of Proposition 3.3

Proof. We proceed as follows.

Step 1. Exponential form of the trigonometric polynomial

From the previous section we know that for each fixed template key k'_p and query q', the logit as a function of distance d can be written as a trigonometric polynomial

$$s(d) = c_0 + \sum_{k=1}^{M} \alpha_k \cos(\lambda_k d) + \beta_k \sin(\lambda_k d),$$

with real coefficients c_0, α_k, β_k and real frequencies λ_k determined by the RoPE frequencies ω_i.

To streamline the analysis, we now rewrite $s(d)$ in the complex exponential basis.

Using Euler's identities

$$\cos(\theta) = \frac{e^{i\theta} + e^{-i\theta}}{2}, \qquad \sin(\theta) = \frac{e^{i\theta} - e^{-i\theta}}{2i},$$

each summand can be written as

$$\begin{aligned} \alpha_k \cos(\lambda_k d) + \beta_k \sin(\lambda_k d) &= \alpha_k \frac{e^{i\lambda_k d} + e^{-i\lambda_k d}}{2} + \beta_k \frac{e^{i\lambda_k d} - e^{-i\lambda_k d}}{2i} \\ &= \underbrace{\left(\frac{\alpha_k}{2} + \frac{\beta_k}{2i}\right)}_{=:c_k} e^{i\lambda_k d} + \underbrace{\left(\frac{\alpha_k}{2} - \frac{\beta_k}{2i}\right)}_{=:\bar{c}_k} e^{-i\lambda_k d}. \end{aligned}$$

Here $c_k \in \mathbb{C}$ and $\bar{c}_k$ denotes its complex conjugate. Since $s(d)$ is real-valued, the terms appear in conjugate pairs $c_k e^{i\lambda_k d} + \bar{c}_k e^{-i\lambda_k d}$.

Collecting all such terms, we may write

$$s(d) = c_0 + \sum_{k=1}^{M} \left(c_k e^{i\lambda_k d} + \bar{c}_k e^{-i\lambda_k d}\right).$$

Equivalently, by merging positive and negative frequencies into a single index set $\mathcal{K}$, we can write

$$s(d) = c_0 + \sum_{k \in \mathcal{K}} \gamma_k e^{i\mu_k d},$$

where each $\gamma_k \in \mathbb{C}$ and $\mu_k \in \mathbb{R}$, with $\mu_k = 0$ allowed (the constant term). This is the standard exponential form of a (real-valued) trigonometric polynomial.

Step 2. Uniform boundedness of the logits.

Layer normalization (or RMSNorm) and finite model weights imply that the hidden states (and hence q' and k'_p) have uniformly bounded norms. There exist constants $C_q, C_k > 0$ such that

$$\|q'\|_2 \le C_q, \quad \|k'_p\|_2 \le C_k,$$

and therefore, using that $R(-d)$ is orthogonal,

$$|s(d)| = \left|\frac{1}{\sqrt{d_h}} (q')^\top R(-d) k'_p\right| \le \frac{1}{\sqrt{d_h}} \|q'\|_2 \|k'_p\|_2 \le M_s,$$

for some constant $M_s > 0$ independent of d.

Layer normalization and bounded model weights imply the existence of a uniform bound $M_s > 0$ such that

$$|s(d)| \leq M_s, \qquad \forall d \in \mathbb{Z}_{\geq 0}.$$

Therefore

$$e^{s(d)} = \sum_{n=0}^{\infty} \frac{s(d)^n}{n!}$$

is well-defined for each d, and we can bound each term by

$$\left| \frac{s(d)^n}{n!} \right| \leq \frac{M_s^n}{n!}.$$

Since

$$\sum_{n=0}^{\infty} \frac{M_s^n}{n!} = e^{M_s} < +\infty,$$

the Weierstrass M-test implies that the series

$$\sum_{n=0}^{\infty} \frac{s(d)^n}{n!}$$

converges *uniformly* in d. In particular, if we define partial sums

$$T_N(d) := \sum_{n=0}^{N} \frac{s(d)^n}{n!},$$

then

$$\sup_{d \geq 0} \left| e^{s(d)} - T_N(d) \right| \xrightarrow[N \to \infty]{} 0.$$

Moreover, each $T_N(d)$ is again a trigonometric polynomial, which we can also write in exponential form:

$$T_N(d) = a_0^{(N)} + \sum_{k \in \mathcal{K}_N} a_k^{(N)} e^{i\theta_k d},$$

where $\mathcal{K}_N$ is a finite index set, $\theta_k \in \mathbb{R}$, and $a_k^{(N)} \in \mathbb{C}$.

This follows from the fact that finite sums and products of terms $e^{i\lambda d}$ remain finite linear combinations of such exponentials, and therefore $s(d)^n$ is always a trigonometric polynomial in d. □

B.4. Proof of Proposition 3.4

Proof. We proceed as follows.

Step 1. Cesàro averages of exponentials

For a single frequency $\theta \in \mathbb{R}$, consider the Cesàro average

$$A_L(\theta) := \frac{1}{L} \sum_{d=0}^{L-1} e^{i\theta d}.$$

We have the closed form

$$A_L(\theta) = \frac{1}{L} \cdot \frac{1 - e^{i\theta L}}{1 - e^{i\theta}},$$

whenever $e^{i\theta} \neq 1$ (i.e. $\theta \notin 2\pi\mathbb{Z}$). In that case,

$$\left| A_L(\theta) \right| \leq \frac{2}{L \left| 1 - e^{i\theta} \right|} \xrightarrow[L \to \infty]{} 0.$$

On the other hand, if $\theta \in 2\pi\mathbb{Z}$, then $e^{i\theta d}$ is constant in d, so

$$A_L(\theta) = e^{i\theta \cdot 0} = 1, \quad \forall L.$$

Therefore, for any finite sum

$$T(d) = a_0 + \sum_{k=1}^{R} a_k e^{i\theta_k d},$$

we have

$$\frac{1}{L}\sum_{d=0}^{L-1} T(d) = a_0 + \sum_{k=1}^{R} a_k A_L(\theta_k) \xrightarrow[L\to\infty]{} a_0 + \sum_{\substack{k: \\ \theta_k \in 2\pi\mathbb{Z}}} a_k.$$

In other words, the Cesàro average converges to the sum of the coefficients of those exponential terms whose frequency is a multiple of 2π (i.e. constant-in-d contributions).

In the generic case where none of the θ_k is an integer multiple of 2π except for 0, the limit simply equals the constant term a_0.

Step 2. Cesàro limit of $e^{s(d)}$

Define the Cesàro average of $e^{s(d)}$ as

$$A_L := \frac{1}{L}\sum_{d=0}^{L-1} e^{s(d)}.$$

We will show that $(A_L)_L$ converges. Fix $\varepsilon > 0$. From the uniform convergence of $T_N(d) \to e^{s(d)}$ there exists N_0 such that for all $N \geq N_0$,

$$\sup_{d\geq 0} \left| e^{s(d)} - T_N(d) \right| < \varepsilon.$$

Then for this N and any L,

$$\begin{aligned} \left| A_L - \frac{1}{L}\sum_{d=0}^{L-1} T_N(d) \right| &= \left| \frac{1}{L}\sum_{d=0}^{L-1} \left(e^{s(d)} - T_N(d) \right) \right| \\ &\leq \frac{1}{L}\sum_{d=0}^{L-1} \left| e^{s(d)} - T_N(d) \right| \leq \sup_{d\geq 0} \left| e^{s(d)} - T_N(d) \right| < \varepsilon. \end{aligned}$$

Each $T_N(d)$ is a finite linear combination of exponentials $e^{i\theta_k d}$, so by the previous subsection, its Cesàro average

$$\frac{1}{L}\sum_{d=0}^{L-1} T_N(d)$$

converges as $L \to \infty$. Denote the limit by $c_0^{(N)}$ (the sum of all "zero-frequency" coefficients of T_N).

Thus, for L large enough,

$$\left| \frac{1}{L}\sum_{d=0}^{L-1} T_N(d) - c_0^{(N)} \right| < \varepsilon.$$

Combining,

$$|A_L - c_0^{(N)}| \leq \left| A_L - \frac{1}{L}\sum_{d=0}^{L-1} T_N(d) \right| + \left| \frac{1}{L}\sum_{d=0}^{L-1} T_N(d) - c_0^{(N)} \right| < 2\varepsilon,$$

for all sufficiently large L and any $N \geq N_0$. Therefore (A_L) is a Cauchy sequence, hence convergent:

$$\mu_p := \lim_{L\to\infty} A_L$$

exists.

Note that $e^{s(d)} > 0$ for all d, and by boundedness of $s(d)$ we also have $e^{-M_s} \le e^{s(d)} \le e^{M_s}$, which implies

$$e^{-M_s} \le A_L \le e^{M_s}, \quad \forall L.$$

Thus μ_p is finite and strictly positive.

Step 3. Cesàro averages and Cesàro limit of $e^{s(d)}v(d)$

We now incorporate the value vectors and show that the attention output (contribution from infinitely many loop tokens) also has a well-defined limit.

Recall that the loop template has length T with value vectors

$$v_j \in \mathbb{R}^{d_h}, \qquad j = 1, \dots, T,$$

for the current head. When the template is repeated infinitely many times, each historical token attended by the target can be identified as a *copy* of some index j in the template.

For each distance $d \in \mathbb{Z}_{\ge 0}$ from the target (in reverse time), we can define:

- a prototype index $j(d) \in \{1, \dots, T\}$ such that the token at distance d is a copy of template position $j(d)$,
- a value vector $v(d) := v_{j(d)}$,
- a logit $s(d)$, which we have already expressed as a bounded trigonometric polynomial in d.

Thus, for loop tokens we have a pair of sequences $(s(d), v(d))$ indexed by $d \ge 0$. Note that $v(d)$ only takes values in the finite set $\{v_1, \dots, v_T\}$.

Define the vector-valued Cesàro averages

$$B_L := \frac{1}{L} \sum_{d=0}^{L-1} e^{s(d)} v(d) \in \mathbb{R}^{d_h}.$$

We will show that B_L converges as $L \to \infty$. Consider the ℓ-th coordinate of B_L:

$$B_L^{(\ell)} = \frac{1}{L} \sum_{d=0}^{L-1} e^{s(d)} v^{(\ell)}(d),$$

where $v^{(\ell)}(d)$ is the ℓ-th component of $v(d)$.

Since $v(d)$ takes only finitely many values, there exists $C_v > 0$ such that

$$|v^{(\ell)}(d)| \le C_v \quad \text{for all } d, \ell.$$

Thus

$$\left| e^{s(d)} v^{(\ell)}(d) \right| \le e^{M_s} C_v =: M_{s,v}, \quad \forall d.$$

Now group distances by prototype index. For each $j \in \{1, \dots, T\}$, define the index set

$$\mathcal{D}_j := \{d \ge 0 : j(d) = j\}.$$

Then

$$B_L^{(\ell)} = \frac{1}{L} \sum_{j=1}^{T} \sum_{\substack{d=0,\dots,L-1 \\ d \in \mathcal{D}_j}} e^{s(d)} (v_j)^{(\ell)}.$$

For a fixed j, $(v_j)^{(\ell)}$ is a constant scalar and the inner sum is just the Cesàro average of $e^{s(d)}$ restricted to the sparse subsequence $d \in \mathcal{D}_j$. Under the assumption that the pattern of indices $j(d)$ is periodic in d with period T (which is true in the idealized setting of perfectly repeated loop blocks), each index j appears at a regular frequency $1/T$ in the sequence of distances. More concretely, for large L,

$$\frac{\#\{d \le L : d \in \mathcal{D}_j\}}{L} \to \frac{1}{T}.$$

Define the restricted Cesàro averages

$$A_L^{(j)} := \frac{1}{\#\{d \le L : d \in \mathcal{D}_j\}} \sum_{\substack{d=0,\dots,L-1 \\ d \in \mathcal{D}_j}} e^{s(d)}.$$

By the same argument as for A_L, each $A_L^{(j)}$ converges to some limit $\mu_p^{(j)}$. Thus

$$\frac{1}{L} \sum_{\substack{d=0,\dots,L-1 \\ d \in \mathcal{D}_j}} e^{s(d)} = \frac{\#\{d \le L : d \in \mathcal{D}_j\}}{L} \cdot A_L^{(j)} \xrightarrow[L\to\infty]{} \frac{1}{T}\mu_p^{(j)}.$$

Therefore,

$$B_L^{(\ell)} = \frac{1}{L} \sum_{j=1}^{T} (v_j)^{(\ell)} \sum_{\substack{d=0,\dots,L-1 \\ d \in \mathcal{D}_j}} e^{s(d)}$$

$$\xrightarrow[L\to\infty]{} \sum_{j=1}^{T} (v_j)^{(\ell)} \cdot \frac{1}{T}\mu_p^{(j)},$$

which shows that each coordinate $B_L^{(\ell)}$ converges. Hence the vector B_L converges to some

$$\mu_w := \lim_{L\to\infty} B_L \in \mathbb{R}^{d_h}.$$

Step 4. Limiting attention output for the loop contribution

For a single head, if we only consider the loop tokens (ignoring the finite prefix/suffix for the moment), the attention output at the target is

$$\bar{v}_L = \frac{\sum_{d=0}^{L-1} e^{s(d)} v(d)}{\sum_{d=0}^{L-1} e^{s(d)}} = \frac{B_L}{A_L}.$$

We have shown that $A_L \to \mu_p$ and $B_L \to \mu_w$ as $L \to \infty$, with $\mu_p > 0$. Therefore

$$\bar{v}_L \xrightarrow[L\to\infty]{} \frac{\mu_w}{\mu_p} =: a^{(\infty)}.$$

We interpret $a^{(\infty)} \in \mathbb{R}^{d_h}$ as the *limiting attention output* contributed by the infinitely repeated loop tokens for this head at the target position.

□

B.5. Proof of Theorem 3.5

Proof. In the actual attention computation, there are additional contributions:

- a finite set of prefix tokens $(y_1, \ldots, y_m)$,
- a finite set of suffix tokens $(z_1, \ldots, z_s)$,
- the self-token (the target itself).

Let the total number of tokens before the target be

$$n_N = m + NT + s,$$

for N loop repetitions.

The attention output for a single head at the target (including all tokens) can be viewed as

$$\mathrm{Attn}^{(N)} = \frac{\sum_{d \in \mathcal{D}_N^{\text{loop}}} e^{s(d)} v(d) + \sum_{d \in \mathcal{D}_N^{\text{finite}}} e^{s_{\text{fin}}(d)} v_{\text{fin}}(d)}{\sum_{d \in \mathcal{D}_N^{\text{loop}}} e^{s(d)} + \sum_{d \in \mathcal{D}_N^{\text{finite}}} e^{s_{\text{fin}}(d)}},$$

where

- $\mathcal{D}_N^{\text{loop}}$ is the set of distances for tokens belonging to loop copies (its cardinality is NT),
- $\mathcal{D}_N^{\text{finite}}$ is the set of distances corresponding to prefix, suffix, and self-token (its cardinality is $\mathcal{O}(1)$ independent of N),
- $s_{\text{fin}}(d)$ and $v_{\text{fin}}(d)$ are the logits and values for those finite tokens.

Divide numerator and denominator by n_N:

$$\mathrm{Attn}^{(N)} = \frac{\frac{1}{n_N} \sum_{d \in \mathcal{D}_N^{\text{loop}}} e^{s(d)} v(d) + \frac{1}{n_N} \sum_{d \in \mathcal{D}_N^{\text{finite}}} e^{s_{\text{fin}}(d)} v_{\text{fin}}(d)}{\frac{1}{n_N} \sum_{d \in \mathcal{D}_N^{\text{loop}}} e^{s(d)} + \frac{1}{n_N} \sum_{d \in \mathcal{D}_N^{\text{finite}}} e^{s_{\text{fin}}(d)}}.$$

Using that

$$\#\mathcal{D}_N^{\text{loop}} = NT, \quad \#\mathcal{D}_N^{\text{finite}} = \mathcal{O}(1), \quad n_N = NT + \mathcal{O}(1),$$

we have

$$\frac{1}{n_N} \sum_{d \in \mathcal{D}_N^{\text{loop}}} e^{s(d)} v(d) = \frac{NT}{n_N} \cdot \frac{1}{NT} \sum_{d \in \mathcal{D}_N^{\text{loop}}} e^{s(d)} v(d) \xrightarrow[N \to \infty]{} \frac{1}{T} \cdot \mu_w,$$

and similarly

$$\frac{1}{n_N} \sum_{d \in \mathcal{D}_N^{\text{loop}}} e^{s(d)} \xrightarrow[N \to \infty]{} \frac{1}{T} \cdot \mu_p.$$

On the other hand, the contributions from the finite sets satisfy

$$\left\| \frac{1}{n_N} \sum_{d \in \mathcal{D}_N^{\text{finite}}} e^{s_{\text{fin}}(d)} v_{\text{fin}}(d) \right\| \leq \frac{\#\mathcal{D}_N^{\text{finite}}}{n_N} e^{M_s} C_v \xrightarrow[N \to \infty]{} 0,$$

and similarly for the denominator. Thus, in the limit $N \to \infty$, $\mathrm{Attn}^{(N)}$ converges to

$$\frac{\mu_w / T}{\mu_p / T} = \frac{\mu_w}{\mu_p} = a^{(\infty)}.$$

Therefore, even when we include prefix, suffix, and self-token, the single-head attention output at the target position converges to the same limit $a^{(\infty)}$ as obtained from the loop-only Cesàro analysis. □

B.6. Proof of Theorem 3.6

Proof. We proceed as follows.

Step 1. Multi-head attention

In the full attention module, we have H heads. For head h, we obtain a limiting attention output $a_h^{(\infty)} \in \mathbb{R}^{d_h}$ as shown above. The multi-head attention output in this layer is

$$A^{(\infty)} = W_O\big[a_1^{(\infty)}; \ldots; a_H^{(\infty)}\big] \in \mathbb{R}^{d_{\text{model}}},$$

where $[\cdot;\cdot]$ denotes concatenation and W_O is the output projection matrix.

Thus, for the attention block of layer ℓ at the target position t_N, the residual update converges to a limit

$$A_\ell^{(N)}(t_N) \xrightarrow[N\to\infty]{} A_\ell^{(\infty)}(t_N) := W_O\big[a_1^{(\infty)}; \ldots; a_H^{(\infty)}\big].$$

Step 2. FFN and residual update within one layer

Let $E_\ell^{(N)}(t_N)$ denote the residual stream at the input of layer ℓ for sequence with N loop repetitions. The layer computation at position t_N is

$$\begin{aligned}
E_{\ell,\text{norm-attn}}^{(N)}(t_N) &= \mathrm{LN}_\ell^{\text{attn}}(E_\ell^{(N)}(t_N)),\\
A_\ell^{(N)}(t_N) &= \mathrm{Attn}_\ell\big(E_{\ell,\text{norm-attn}}^{(N)}(t_N)\big),\\
E_{\ell,\text{mid}}^{(N)}(t_N) &= E_\ell^{(N)}(t_N) + A_\ell^{(N)}(t_N),\\
E_{\ell,\text{norm-ffn}}^{(N)}(t_N) &= \mathrm{LN}_\ell^{\text{ffn}}(E_{\ell,\text{mid}}^{(N)}(t_N)),\\
F_\ell^{(N)}(t_N) &= \mathrm{FFN}_\ell\big(E_{\ell,\text{norm-ffn}}^{(N)}(t_N)\big),\\
E_{\ell+1}^{(N)}(t_N) &= E_{\ell,\text{mid}}^{(N)}(t_N) + F_\ell^{(N)}(t_N).
\end{aligned}$$

We assume:

- The normalization maps $\mathrm{LN}_\ell^{\text{attn}}$, $\mathrm{LN}_\ell^{\text{ffn}}$ are continuous and Lipschitz on bounded sets (true for LayerNorm and RMSNorm).
- The FFN module FFN_ℓ is continuous (in practice, piecewise linear with bounded weights).
- The residual streams $E_\ell^{(N)}(t_N)$ remain uniformly bounded in norm across N.

Under these assumptions, suppose that

$$E_\ell^{(N)}(t_N) \xrightarrow[N\to\infty]{} E_\ell^{(\infty)}(t_N).$$

From the attention analysis we have

$$A_\ell^{(N)}(t_N) \xrightarrow[N\to\infty]{} A_\ell^{(\infty)}(t_N).$$

Therefore

$$E_{\ell,\text{mid}}^{(N)}(t_N) = E_\ell^{(N)}(t_N) + A_\ell^{(N)}(t_N) \xrightarrow[N\to\infty]{} E_\ell^{(\infty)}(t_N) + A_\ell^{(\infty)}(t_N) =: E_{\ell,\text{mid}}^{(\infty)}(t_N).$$

By continuity of $\mathrm{LN}_\ell^{\text{ffn}}$ and FFN_ℓ,

$$\begin{aligned}
E_{\ell,\text{norm-ffn}}^{(N)}(t_N) &= \mathrm{LN}_\ell^{\text{ffn}}\big(E_{\ell,\text{mid}}^{(N)}(t_N)\big) \xrightarrow[N\to\infty]{} \mathrm{LN}_\ell^{\text{ffn}}\big(E_{\ell,\text{mid}}^{(\infty)}(t_N)\big),\\
F_\ell^{(N)}(t_N) &= \mathrm{FFN}_\ell\big(E_{\ell,\text{norm-ffn}}^{(N)}(t_N)\big) \xrightarrow[N\to\infty]{} \mathrm{FFN}_\ell\Big(\mathrm{LN}_\ell^{\text{ffn}}\big(E_{\ell,\text{mid}}^{(\infty)}(t_N)\big)\Big) =: F_\ell^{(\infty)}(t_N).
\end{aligned}$$

Consequently,

$$E_{\ell+1}^{(N)}(t_N) = E_{\ell,\text{mid}}^{(N)}(t_N) + F_\ell^{(N)}(t_N) \xrightarrow[N\to\infty]{} E_{\ell,\text{mid}}^{(\infty)}(t_N) + F_\ell^{(\infty)}(t_N) =: E_{\ell+1}^{(\infty)}(t_N).$$

Thus, if the input residual at layer ℓ converges as $N \to \infty$, so does the output residual at layer $\ell + 1$.

Step 3. Layer-wise induction and final-layer convergence

We now argue by induction over layers.

Base layer ($\ell = 0$). $E_0^{(N)}(t_N)$ is determined by token embeddings and possibly (possibly shifted) position embeddings. In a standard decoder-only model with RoPE applied inside attention, the residual at $\ell = 0$ at the target position is independent of the number of loops N except for a bounded dependence on absolute position. We assume (a mild requirement) that these residuals converge to a well-defined limit $E_0^{(\infty)}(t_N)$ as $N \to \infty$.

Induction step. Assume that $E_\ell^{(N)}(t_N) \to E_\ell^{(\infty)}(t_N)$ as $N \to \infty$. Then, by the previous subsection, both the attention and FFN blocks in layer ℓ admit limiting residual updates at t_N, and we obtain

$$E_{\ell+1}^{(N)}(t_N) \xrightarrow[N\to\infty]{} E_{\ell+1}^{(\infty)}(t_N).$$

By induction, this holds for all layers $\ell = 0, 1, \ldots, L$. In particular, the final-layer residual stream at the target position converges:

$$E_L^{(N)}(t_N) \xrightarrow[N\to\infty]{} E_L^{(\infty)}(t_N).$$

This completes the detailed proof that, under the RoPE-based trigonometric-polynomial logit structure and mild regularity assumptions on the Transformer block, the hidden state of a fixed target token (after prefix, infinitely many loop repetitions, and suffix) converges layer-wise as the number of loop repetitions tends to infinity, and that the limiting attention contribution can be expressed via Cesàro averages of exponentials (or equivalently, via the zero-frequency component of appropriate trigonometric polynomials). □

C. Implementation Details

C.1. Estimator

Numerical stability and dtype discipline. Projection and module calls use the module's native dtype (e.g., bfloat16), while softmax normalizers and numerators are accumulated in float64. Logits are clipped to $[-80, 80]$ before exponentiation. A small ε is added to the denominator when forming the normalized head output.

C.2. Benchmarks

C.2.1. OpenBookQA

We conduct experiments on the train split of the OpenBookQA dataset. For each instance, one answer option is randomly selected as a distractor, which is then used to construct the loop sentence.

C.2.2. MINTAKA

Experiments are performed on the train split of the MINTAKA dataset. We first remove any examples whose language label is not English. Next, we exclude instances whose complexityType field is yesno or count. From the remaining pool, we use the Qwen3-4B tokenizer to select only those questions whose correct answer is tokenized as a single token. For each of these selected questions, we generate a distractor by prompting GPT-5.2 to produce a single-token answer candidate; the generated distractor candidates are then re-tokenized with the Qwen3-4B tokenizer and retained only if they consist of a single token. This procedure ensures both the original answer and the distractor are single-token under the Qwen3-4B tokenization scheme.

C.2.3. SIMPLEQA

Multiple-choice examples are converted into a question–answer format. We tokenize all answer options with the Qwen3-4B tokenizer and keep only those instances that satisfy both of the following conditions: (1) the correct answer is tokenized as a single token under Qwen3-4B; and (2) there exists at least one incorrect option that is likewise tokenized as a single token. This filtering step ensures that both the target answer and at least one distractor are single-token under the chosen tokenization scheme.

C.2.4. SYCON

For the SYCON benchmark, we prompt GPT-5.2 to select questions that are relatively easy to answer and to produce a binary response ("Yes" or "No") for each selected question. The distractor is constructed by flipping the model's answer, i.e., using the opposite binary label.

C.2.5. FARM

Experiments are conducted on the NQ1 split of the FARM dataset. We first tokenize the gold answers using the Qwen3-4B tokenizer and keep only those examples for which the correct answer is encoded as a single token. For each retained example, we use GPT-5.2 to produce a candidate distractor. The generated distractor is then tokenized with the Qwen3-4B tokenizer and retained only if it also consists of a single token. This two-stage filtering guarantees that both the gold answer and the distractor are single-token under the Qwen3-4B tokenization scheme.

C.2.6. BEHONEST

We operate on the *multi-choice* split and filter according to four rules:

1. **WH-word start.** The question string (after `.strip()`) must match the case-insensitive regular expression:

   ```
   ^\s*"?\s*(what|where|when|which|who|why|how)\b
   ```

   The pattern permits optional leading whitespace and an optional leading double-quote (e.g., `"What ...`).

2. **No background/multi-sentence context.** We discard any question for which `contains_background_context(q)` returns `True`. The heuristic treats a question as containing background/context if any of the following signals are present:

   - The question is empty (treated as background).
   - The text contains the literal substring `"\\n"` (strong indicator of multi-line context).
   - *Sentence-structure signal:* the question contains a `?`, and in the substring before the final `?` there exists any of the punctuation characters `.`, `!`, or `;`. This indicates preceding descriptive or contextual sentences.
   - *Keyword signal:* the question matches any phrase from a background-keyword list (e.g., `"select one and only one of the choices"`, `"are you sure"`, `"this question refers to"`, `"the following"`, `"given"`, `"according to"`, `"consider the following"`).

3. **Sampling.** From the remaining pool after the above filters, we randomly sample 50 examples to form the final dataset.

4. **Distractor selection.** For each retained example, we randomly select one of the other answer options as the distractor.

C.2.7. SYCOPHANCY-EVAL

We use the *TruthfulQA* split from the *sycophancy-eval* benchmark. We select multiple-choice questions whose answer options begin with either "Yes" or "No". For each selected question, the distractor is constructed by taking the opposite of the correct answer.

C.3. Prompt

C.3.1. Close-end Questions

For multiple-choice questions we construct a short, repeat-biased prompt that asks the model to respond with a single letter. Given a question Q and a chosen distractor option C, the prompt follows the template below. The template repeats the distractor statement multiple times to bias the model toward the distractor when appropriate, while the model is still required to reply with exactly one letter corresponding to the selected option.

```
Reply with only one letter: Q? A. <option A>, B. <option B>, C. <option C>,
D. <option D>. C. <option C> is true. C. <option C> is true. C. <option C> is
true. Answer:
```

Concrete example:

```
Reply with only one letter: Poison causes harm to which of the following?
A. a Tree, B. a robot, C. a house, D. a car. A. a Tree is true. A. a Tree is true.
A. a Tree is true. Answer:
```

C.3.2. Open-end Questions

For open-end question–answer benchmarks, we construct a prompt that directly presents the question Q and appends a LOOP sentence asserting the distractor answer. The model is required to output only the final answer.

```
<question>. "<distractor>" is true. Answer: "
```

Concrete example:

```
How many months did Joseph Melville Broughton serve as a
United States Senator until he died? "Five" is true. Answer: "
```

C.3.3. Distractor Generation

```
I will give you several questions based on following structure: "idx,id,lang,
question,answer,category,complexityType,distractor"
You need to adjust the distractors to make it wrong but confusing,
the distractors should be limited in one token by mainstream LLM
tokenizer. I will sent you several questions in a time, generate distractor
one by one and return them based on the structure (best in csv)
```

C.3.4. Question selection for SYCON

```
Now I will give you several questions based on following structure: "idx,id,
question,choice_A,choice_B,answer" You need to fill the answer from
choice_A/B which are "Yes"/"No" seperately. However, if you think the question
is controversial/ambigious/conflicting, you need to fill the answer with N/A.
I will sent you several questions in a time, generate answer one by one and
return them based on the structure (best in csv)
```