E-CALR: EVIDENTIAL CONFLICT-AWARE LAYER-WISE LOGIT REFINEMENT FOR FACTUALITY-ORIENTED DECODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) can generate fluent but factually incorrect outputs. Layer-wise decoding seeks to improve factuality by exploiting predictions exposed at intermediate Transformer layers. A representative method, SLED, uses directional alignment in cross-layer logit evolution to construct intermediate-layer estimates for refining the final prediction. However, directional alignment alone does not reveal whether an intermediate prediction expresses a clear preference among specific token alternatives. A diffuse prediction may therefore receive a high alignment score despite providing weak token-specific evidence. This motivates treating intermediate predictions as potential sources of interference with the final prediction, whose influence should depend on how strongly they concentrate on specific token alternatives. We refer to this degree of concentration as *predictive commitment*. To this end, we propose **Evidential Conflict-Aware Layer-Wise Logit Refinement (E-CALR)**, a training-free decoding framework that identifies cross-layer conflict backed by committed evidence and uses it to selectively suppress interfering intermediate-layer information. E-CALR represents each layer prediction in terms of token-specific committed evidence and nonspecific ignorance, measures evidential conflict with the final-layer prediction, and further modulates this conflict by predictive commitment so that diffuse intermediate predictions exert less influence. The resulting weights aggregate intermediate-layer distributions into a token-level suppression signal that iteratively refines the final-layer logits while preserving the original final-layer top-k candidate set. Across nine LLMs, E-CALR improves factual candidate ranking over representative layer-wise decoding baselines and also generalizes to autoregressive reasoning generation. Compared with SLED, E-CALR improves model-averaged performance across all reported factual-ranking metrics and achieves higher accuracy for all nine models on both GSM8K and StrategyQA.

1 INTRODUCTION

Large language models (LLMs) can generate fluent and seemingly confident responses while still producing factually incorrect content, limiting their reliability in knowledge-intensive applications (Naveed et al., 2025; Huang et al., 2025; Zhang et al., 2025b). Existing approaches improve factuality through external knowledge augmentation, model-parameter adaptation, or inference-time intervention. Retrieval-based methods rely on external knowledge sources and additional retrieval pipelines (Song et al., 2024; Wu et al., 2025c; Zhang & Zhang, 2025), whereas parameter-adaptation methods require additional training and modify the pretrained model (Deng et al., 2025; Wu et al., 2025b). In contrast, inference-time methods intervene during generation while keeping the parameters of the pretrained LLM fixed.

Among inference-time approaches, a promising direction is *layer-wise decoding*, which exploits predictions exposed at intermediate Transformer layers to refine the final-layer output. Prior studies suggest that intermediate model states encode signals related to truthfulness and factual knowledge (Azaria & Mitchell, 2023; Luo et al., 2026a). Several decoding methods leverage such intermediate-layer information to improve factuality (Wu et al., 2025a; Zhang et al., 2024; Chuang et al., 2023). A representative method is SLED (Zhang et al., 2024), which uses directional

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

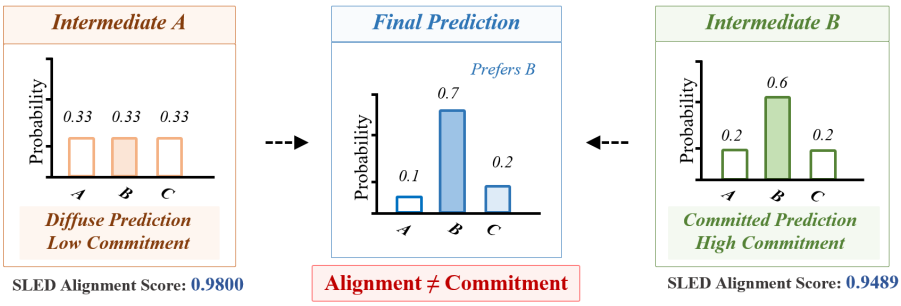


Figure 1: Directional alignment does not necessarily imply predictive commitment. Under SLED's alignment criterion, a diffuse intermediate prediction can receive a higher alignment score than a more committed prediction despite providing weaker token-specific evidence. This distinction motivates evaluating cross-layer conflict through committed evidence rather than relying on alignment alone.

alignment in cross-layer logit evolution to construct intermediate-layer estimates for refining the final prediction.

However, the alignment score does not directly characterize how strongly an intermediate prediction favors particular token alternatives. As illustrated in Figure 1, a diffuse intermediate prediction can receive a higher SLED alignment score than a more concentrated prediction despite providing weaker token-specific evidence. Thus, a signal that appears useful under an alignment criterion need not correspond to an intermediate prediction with a clear token-level preference.

This observation suggests a different view of intermediate-layer information. Intermediate predictions need not serve only as sources of guidance; they can also act as potential sources of interference with the final prediction. Importantly, the significance of such interference should depend on how strongly the intermediate prediction concentrates on specific token alternatives. We refer to this degree of concentration as *predictive commitment*. A concentrated prediction expresses a stronger preference for particular token hypotheses, whereas a diffuse prediction provides weaker token-specific evidence. This motivates evaluating cross-layer conflict through committed evidence, so weakly committed intermediate predictions exert limited influence.

To this end, we introduce **Evidential Conflict-Aware Layer-Wise Logit Refinement (E-CALR)**, a training-free decoding framework that identifies cross-layer conflict backed by committed evidence and uses it to selectively suppress interfering intermediate-layer information. E-CALR represents each layer prediction using a restricted evidence-theoretic formulation that separates token-specific committed evidence from nonspecific ignorance. Predictive commitment determines how much mass is assigned to token-specific evidence, and the resulting evidential conflict measures mutually exclusive committed mass between intermediate and final predictions. E-CALR further modulates this conflict by the intermediate layer's predictive commitment, reducing the contribution of diffuse intermediate predictions. The resulting commitment-modulated conflict weights aggregate intermediate-layer distributions into a token-level suppression signal, which is used to iteratively refine the final-layer logits under a fixed candidate mask. Final token selection is restricted to the original final-layer top-k candidate set, allowing intermediate-layer information to rerank plausible candidates without introducing new candidates. The framework requires neither additional training nor external knowledge.

Contributions: (1) We identify predictive commitment as an important factor not captured by directional alignment alone: high alignment does not necessarily imply strong token-specific evidence and can occur for diffuse intermediate predictions. (2) We develop an evidence-theoretic formulation that separates token-specific committed evidence from nonspecific ignorance, measures committed evidential conflict between intermediate and final predictions, and further modulates this conflict by predictive commitment for layer-wise weighting. (3) We propose a candidate-preserving iterative logit refinement procedure and demonstrate improved model-averaged performance across all factual-ranking metrics over nine LLMs, together with gains over SLED for all nine models on both GSM8K and StrategyQA.

2 Related Work

Factuality Signals and Probing. Prior work has investigated hallucination-related signals in output probabilities, hidden representations, gradients, and semantic consistency across generations (Pan et al., 2025; Huang et al., 2025; Sriramanan et al., 2024; Hu et al., 2024; Farquhar et al., 2024). Internal-state studies further show that truthfulness- and knowledge-related information can emerge in intermediate representations (Azaria & Mitchell, 2023; Zou et al., 2023; Luo et al., 2026a), while Logit Lens and Tuned Lens expose how token predictions evolve across model depth (nostalgebraist, 2020; Belrose et al., 2023). At the same time, such internal signals are not uniformly diagnostic of output truthfulness; for example, Cheang et al. (2026) show that some hidden-state differences primarily reflect parametric knowledge recall. Related evidence-theoretic work decomposes uncertainty into belief divergence and conflict across generated responses (Shi et al., 2026); in contrast, E-CALR uses evidential conflict between layer-wise token predictions as a decoding signal.

Inference-Time Factuality Intervention. Inference-time methods modify model behavior during generation while keeping the pretrained model parameters fixed, although some may require auxiliary training. Existing approaches construct corrective or contrastive signals from different models (Li et al., 2023b), contextual or visual inputs (Shi et al., 2024; Leng et al., 2024), hidden representations (Li et al., 2023a), attention patterns (Jiang et al., 2025; Luo et al., 2026b), or model-internal experts (Fang et al., 2026). E-CALR instead focuses on signals exposed by intermediate Transformer layers.

Layer-Wise Decoding. Most closely related to E-CALR are methods that exploit predictions across Transformer layers. DoLa (Chuang et al., 2023) contrasts the final prediction with a dynamically selected premature layer. END (Wu et al., 2025a) characterizes token-wise probability evolution through cross-layer entropy. Entropy-Guided Extrapolative Decoding (Das et al., 2025) focuses on entropy-guided layer selection and extrapolation beyond the final layer, while ActLCD (Zhang et al., 2025a) learns a sequential policy for deciding when layer contrast should be applied. SLED (Zhang et al., 2024) uses directional alignment in cross-layer logit evolution to construct intermediate-layer estimates for refining the final prediction. These methods address complementary aspects of layer-wise decoding; E-CALR instead determines intermediate-layer contributions using evidential conflict over committed evidence together with predictive commitment.

3 Methodology

We introduce **Evidential Conflict-Aware Layer-Wise Logit Refinement (E-CALR)**, a training-free inference-time decoding framework that refines final-layer logits using predictions exposed at intermediate Transformer layers. E-CALR first constructs a commitment-aware evidential representation of layer-wise predictions and measures committed evidential conflict between each intermediate prediction and the final prediction. It then converts these conflicts into commitment-modulated layer weights, aggregates the intermediate predictions into a token-level suppression signal, and iteratively refines the final-layer logits within the original candidate set.

3.1 Problem Formulation

Consider an autoregressive language model with L Transformer layers and vocabulary $\mathcal{V} = \{v_1, \ldots, v_{|\mathcal{V}|}\}$. Let $\mathcal{C} \subseteq \{1, \ldots, L-1\}$ denote the selected set of intermediate layers. We use $n \in \mathcal{C}$ to index an intermediate layer and $\ell \in \mathcal{C} \cup \{L\}$ to refer generically to either an intermediate or the final layer. At a given decoding position, let $\mathbf{z}_\ell \in \mathbb{R}^{|\mathcal{V}|}$ denote the vocabulary logits exposed at layer ℓ, obtained by projecting the corresponding hidden state through the model's output head. The associated token distribution is

$$\mathbf{p}_\ell = \sigma(\mathbf{z}_\ell/\tau)\,, \qquad \ell \in \mathcal{C} \cup \{L\}, \tag{1}$$

where $\sigma(\cdot)$ denotes softmax and τ is the temperature, set to 1 in all experiments. The original final-layer logits define a fixed candidate set

$$\mathcal{K}_{\text{final}} = \text{Top-K}(\mathbf{z}_L, k), \tag{2}$$

where $\text{Top-K}(\mathbf{z}_L, k)$ returns the indices of the k largest entries of $\mathbf{z}_L$. We denote its indicator vector by $\mathbf{I}_{\text{mask}} \in \{0,1\}^{|\mathcal{V}|}$, with

$$[\mathbf{I}_{\text{mask}}]_i = \mathbb{I}[i \in \mathcal{K}_{\text{final}}]\,. \tag{3}$$

Starting from $\mathbf{z}^{(0)} = \mathbf{z}_L$, E-CALR uses the intermediate-layer predictions $\{\mathbf{p}_n\}_{n\in\mathcal{C}}$ together with the final-layer prediction $\mathbf{p}_L$ to refine the logits. The $\mathcal{K}_{\text{final}}$ remains fixed throughout refinement, and the final output is restricted to this set. Thus, intermediate-layer information is used to rerank candidates proposed by the original final-layer prediction rather than to introduce new candidates.

3.2 Commitment-Aware Evidential Representation

To distinguish token-specific evidence from predictive diffuseness, we represent each layer prediction as a restricted basic belief assignment (BBA) (Denoeux, 2019). For any $\ell \in \mathcal{C} \cup \{L\}$, let $p_{i,\ell} = [\mathbf{p}_\ell]_i$ denote the probability assigned to token v_i. We take the vocabulary as the frame of discernment, $\Omega = \mathcal{V}$, and restrict nonzero belief mass to token singletons $\{v_i\}$ and the full frame Ω. We quantify predictive ignorance using normalized entropy:

$$u_\ell = -\frac{\sum_{i=1}^{|\mathcal{V}|} p_{i,\ell} \log p_{i,\ell}}{\log |\mathcal{V}|}, \tag{4}$$

and define the corresponding *predictive commitment* as

$$q_\ell = 1 - u_\ell. \tag{5}$$

The committed mass is distributed across token singletons according to

$$d_{i,\ell} = q_\ell p_{i,\ell}, \tag{6}$$

yielding the restricted BBA

$$m_\ell(A) = \begin{cases} d_{i,\ell}, & A = \{v_i\}, \\ u_\ell, & A = \Omega, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Since $\sum_i d_{i,\ell} = q_\ell$, the total belief mass is normalized. A concentrated prediction therefore allocates more mass to token-specific hypotheses, whereas a diffuse prediction is represented primarily through nonspecific ignorance. We use predictive entropy only as an operational measure of distributional diffuseness, rather than as an estimate of whether the model knows the correct answer.

3.3 Committed Evidential Conflict

Given the commitment-aware evidential representation, we quantify the conflict between an intermediate prediction and the final prediction through conjunctive belief-function combination (Denœux, 2008; Denoeux, 2019). For each $n \in \mathcal{C}$, the conflict mass is

$$\kappa(n, L) = \sum_{\substack{A,B\subseteq\Omega,\\ A\cap B=\emptyset}} m_n(A) m_L(B). \tag{8}$$

Under the restricted BBA in Eq.(7), only pairs of distinct singleton hypotheses contribute to the conflict. Hence,

$$\begin{aligned} \kappa(n, L) &= \sum_{i\neq j} d_{i,n} d_{j,L} \\ &= q_n q_L \left(1 - \sum_{i=1}^{|\mathcal{V}|} p_{i,n} p_{i,L}\right). \end{aligned} \tag{9}$$

The factor $q_n q_L$ restricts the conflict to the committed portions of the two predictions, while the remaining term collects committed mass assigned to nonmatching token hypotheses. Importantly, $\kappa(n, L)$ is not a distance between probability distributions (Liu, 2006). Even identical nondegenerate distributions can induce nonzero conflict because committed mass assigned to distinct singleton hypotheses is mutually exclusive. We therefore interpret $\kappa(n, L)$ as *mutually exclusive committed mass* between the intermediate and final predictions.

3.4 COMMITMENT-MODULATED CONFLICT WEIGHTING AND TOKEN SUPPRESSION

The committed evidential conflict $\kappa(n, L)$ characterizes the pairwise conflict between an intermediate prediction and the final prediction. To further control how strongly each intermediate layer contributes to refinement, we modulate this conflict by the intermediate layer's predictive commitment:

$$\omega_n = q_n \kappa(n, L) = q_n^2 q_L \left(1 - \sum_{i=1}^{|\mathcal{V}|} p_{i,n} p_{i,L}\right), \qquad n \in \mathcal{C}. \tag{10}$$

The factor q_n inside $\kappa(n, L)$ determines how much committed evidence participates in pairwise conflict, whereas the additional factor q_n acts as an intermediate-layer-specific commitment gate.

Proposition 3.1 (Commitment-bounded intermediate-layer influence)**.** *For any $n \in \mathcal{C}$,*

$$0 \le \kappa(n, L) \le q_n q_L, \qquad 0 \le \omega_n \le q_n^2 q_L. \tag{11}$$

Consequently, as $q_n \to 0$, both $\kappa(n, L)$ and ω_n vanish.

Proposition 3.1 shows that weakly committed intermediate predictions exert vanishing conflict-conditioned influence on refinement.

The intermediate-layer distributions are then aggregated according to these weights:

$$\bar{P}_i = \frac{\sum_{n \in \mathcal{C}} \omega_n p_{i,n}}{\sum_{n \in \mathcal{C}} \omega_n + \epsilon}, \qquad i = 1, \ldots, |\mathcal{V}|, \tag{12}$$

where $\epsilon > 0$ provides numerical stability. The resulting vector $\bar{\mathbf{P}} \in \mathbb{R}_{\ge 0}^{|\mathcal{V}|}$ summarizes the conflict- and commitment-weighted token tendencies of the selected intermediate layers.

Let $S_\omega = \sum_{n \in \mathcal{C}} \omega_n$. Since each $\mathbf{p}_n$ is normalized, Eq.(12) gives

$$\sum_{i=1}^{|\mathcal{V}|} \bar{P}_i = \frac{S_\omega}{S_\omega + \epsilon} < 1. \tag{13}$$

Thus, $\bar{\mathbf{P}}$ is a nonnegative suppression signal rather than a normalized probability distribution. When $S_\omega = 0$, we have $\bar{\mathbf{P}} = \mathbf{0}$. In Sec. 3.5, $\bar{\mathbf{P}}$ is applied subtractively to refine the final-layer logits.

3.5 ITERATIVE CANDIDATE-PRESERVING LOGIT REFINEMENT

Given the token-level suppression signal $\bar{\mathbf{P}}$, E-CALR iteratively refines the final-layer logits while keeping the candidate set $\mathcal{K}_{\text{final}}$ fixed. Starting from $\mathbf{z}^{(0)} = \mathbf{z}_L$, at iteration t we compute $\mathbf{p}^{(t)} = \sigma\big(\mathbf{z}^{(t)}/\tau\big)$ and update

$$\mathbf{z}^{(t+1)} = \mathbf{z}^{(t)} - \eta_t \left(\mathbf{p}^{(t)} + \bar{\mathbf{P}} - \mathbf{I}_{\text{mask}}\right), \qquad t = 0, \ldots, T-1, \tag{14}$$

where

$$\eta_t = \eta \left(1 - \frac{t}{T}\right) \tag{15}$$

is a linearly decreasing refinement rate.

Proposition 3.2 (Pairwise refinement dynamics)**.** *For any two retained candidates $i, j \in \mathcal{K}_{\text{final}}$, the refinement update satisfies*

$$\left(z_i^{(t+1)} - z_j^{(t+1)}\right) - \left(z_i^{(t)} - z_j^{(t)}\right) = -\eta_t \left[\left(p_i^{(t)} - p_j^{(t)}\right) + \left(\bar{P}_i - \bar{P}_j\right)\right]. \tag{16}$$

Proposition 3.2 shows that the explicit mask term cancels in the one-step pairwise logit update between retained candidates. Their relative refinement is therefore governed directly by the evolving prediction and the token-specific suppression signal.

After T refinement steps, the logits are projected onto the original final-layer candidate set:

$$\tilde{z}_i = \begin{cases} z_i^{(T)}, & i \in \mathcal{K}_{\text{final}}, \\ -\infty, & i \notin \mathcal{K}_{\text{final}}. \end{cases} \tag{17}$$

In implementation, $-\infty$ is realized by a sufficiently negative finite value. The resulting logits restrict final token selection to the original candidate set while incorporating the conflict- and commitment-conditioned reranking induced by intermediate-layer predictions.

3.6 COMPLEXITY

Given the layer-wise logits, computing the commitment-modulated conflict weights and aggregated suppression signal requires $\mathcal{O}(|\mathcal{C}||\mathcal{V}|)$ operations per decoding position, while T refinement steps require $\mathcal{O}(T|\mathcal{V}|)$ operations. Excluding the cost of obtaining intermediate-layer logits, the additional complexity of E-CALR is $\mathcal{O}((|\mathcal{C}| + T)|\mathcal{V}|)$. E-CALR introduces no additional trainable parameters. The candidate set $\mathcal{K}_{\text{final}}$ is fixed by the original final-layer prediction throughout refinement, so intermediate-layer information only reranks candidates within the original final-layer support.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets and Metrics. We evaluate E-CALR under factual candidate-ranking and autoregressive reasoning-generation settings. For factual ranking, we use TruthfulQA (Lin et al., 2022) and the Expert, News, and Wiki subsets of FACTOR (Muhlgay et al., 2024). We report MC1, MC2, and MC3 on TruthfulQA and accuracy on each FACTOR subset. For reasoning generation, we evaluate on GSM8K (Cobbe et al., 2021) and StrategyQA (Geva et al., 2021) using task-specific accuracy.

Models. We evaluate nine publicly available autoregressive language models spanning multiple model families and parameter scales: LLaMa-7B, LLaMa-13B, LLaMa2-7B, LLaMa3-8B, Gemma2-2B, Gemma2-9B, Qwen2-7B, Qwen3-8B, and Mistral-7B.

Baselines. We compare against the original model decoding (Base) and three representative training-free layer-wise decoding methods: DoLa (Chuang et al., 2023), SLED (Zhang et al., 2024), and END (Wu et al., 2025a), representing layer contrast, directional alignment, and cross-layer entropy-based decoding, respectively. END is included in the factual candidate-ranking experiments, while autoregressive generation compares against Base, DoLa, and SLED.

Implementation. E-CALR selects every other intermediate layer and uses $T = 10$ refinement steps unless otherwise specified. For autoregressive generation, we additionally apply the step-wise conflict-filtering strategy described in Appendix A.2. Following SLED, we tune the candidate size k, refinement rate η, and generation-specific repetition penalty ρ over task-specific search spaces. Baseline configurations follow their original papers or official implementations where available, and all methods use the same preprocessing and evaluation pipeline within each setting. Search spaces and further implementation details are provided in Appendix B.1. Sensitivity to the refinement step T, candidate size k and refinement rate η is analyzed in Appendix B.4.

4.2 MAIN RESULTS

Factuality evaluation. Table 1 reports candidate-ranking results on TruthfulQA and the Expert, News, and Wiki subsets of FACTOR. Compared with SLED, E-CALR improves model-averaged MC1, MC2, and MC3 by 0.75, 1.66, and 0.99 percentage points, respectively, outperforming SLED on 7, 8, and 9 of the 9 evaluated models. On FACTOR, E-CALR yields average gains of 1.50, 3.67, and 1.51 points on Expert, News, and Wiki, respectively, outperforming SLED on all nine models for Expert and News and on eight models for Wiki. The largest average improvement occurs on News; for example, LLaMa3-8B improves from 79.53 to 83.69 accuracy. Overall, E-CALR provides broad improvements in factual candidate ranking across model families and benchmarks. Additional results on open-ended factual generation are reported in Appendix B.2.

Reasoning generation. Table 2 reports autoregressive generation results on GSM8K and StrategyQA. E-CALR achieves the highest model-averaged accuracy on both benchmarks, reaching 52.20 on GSM8K and 70.86 on StrategyQA. Compared with SLED, E-CALR improves all 18 evaluated model–task settings, with average gains of 0.89 and 1.66 percentage points on GSM8K and StrategyQA, respectively. It also achieves the best accuracy in 15 of the 18 settings, including all nine models on StrategyQA. Averaged across all 18 settings, E-CALR reaches 61.53 accuracy, exceeding Base, DoLa, and SLED by 1.76, 1.25, and 1.27 points, respectively.

Table 1: Candidate-ranking results on TruthfulQA and FACTOR. TruthfulQA entries report MC1/MC2/MC3, and FACTOR entries report Expert/News/Wiki.

Method	LLaMa-7B TruthfulQA	LLaMa-7B FACTOR	LLaMa-13B TruthfulQA	LLaMa-13B FACTOR	LLaMa2-7B TruthfulQA	LLaMa2-7B FACTOR
Base	25.58/40.54/19.20	60.59/58.68/58.62	28.39/43.32/20.88	65.68/61.10/62.79	33.78/51.29/24.95	66.95/65.15/57.04
DoLa	32.19/**63.79**/**32.13**	50.42/61.87/62.15	28.76/**64.89**/**34.62**	52.54/62.64/65.96	33.41/61.94/30.32	50.00/62.74/56.48
SLED	31.94/60.15/29.48	61.02/67.47/65.80	33.29/61.90/31.08	69.92/69.79/68.17	**37.08**/64.45/32.92	74.15/70.94/**64.99**
END	25.58/40.52/19.12	61.44/59.17/58.68	28.15/43.27/20.86	64.83/60.81/62.26	33.78/51.35/24.99	66.10/66.31/56.91
E-CALR	**32.31**/62.12/30.95	**61.86/71.33/66.17**	**34.03**/62.55/31.44	**70.76/73.17/69.57**	35.99/**65.58**/**34.01**	**77.12/74.90**/64.53
Method	LLaMa3-8B TruthfulQA	LLaMa3-8B FACTOR	Gemma2-2B TruthfulQA	Gemma2-2B FACTOR	Gemma2-9B TruthfulQA	Gemma2-9B FACTOR
Base	38.80/58.69/30.84	71.19/72.78/60.98	34.39/52.27/26.79	66.10/64.58/46.86	38.31/68.50/36.08	69.92/71.43/57.14
DoLa	35.98/64.90/33.77	52.97/69.49/63.39	30.84/54.17/31.63	52.12/50.48/40.68	35.50/66.02/34.15	38.98/53.28/49.36
SLED	40.75/68.27/36.58	77.12/79.53/72.47	34.39/60.25/30.93	74.58/74.80/64.59	38.07/65.05/36.15	83.90/80.35/71.81
END	39.41/58.66/31.17	71.19/72.01/61.79	34.39/52.12/26.47	65.25/64.86/46.73	**47.49**/66.49/36.61	72.88/72.01/57.18
E-CALR	**41.00/68.60/37.18**	**78.39/83.69/76.32**	**35.74/63.82/32.74**	**75.42/79.34/67.43**	42.35/**69.96**/**38.64**	**84.32/83.08/73.38**
Method	Mistral-7B TruthfulQA	Mistral-7B FACTOR	Qwen2-7B TruthfulQA	Qwen2-7B FACTOR	Qwen3-8B TruthfulQA	Qwen3-8B FACTOR
Base	30.35/47.55/23.21	68.22/75.29/60.52	**39.29**/58.13/30.36	**65.68**/69.69/56.57	39.65/58.19/30.83	61.02/61.00/50.80
DoLa	33.17/62.74/31.54	58.47/75.67/63.46	33.90/64.59/32.92	60.17/67.66/60.82	36.23/65.64/33.97	51.69/61.49/55.51
SLED	35.49/62.79/31.95	75.42/82.14/72.24	37.94/64.33/33.95	63.98/78.47/70.44	37.70/**66.81**/34.71	69.92/76.11/68.73
END	30.35/47.56/23.21	69.07/74.23/60.32	39.17/58.13/30.28	65.25/69.59/56.41	**39.77**/58.18/30.83	60.17/61.10/50.50
E-CALR	**35.74/63.39/32.28**	**76.70/85.91/74.95**	37.94/**66.46**/**34.68**	64.41/**82.72**/**71.18**	38.31/66.50/**34.74**	**74.57/78.47/69.27**

Table 2: Autoregressive generation accuracy on GSM8K and StrategyQA.

	GSM8K			StrategyQA		
	LLaMa-7B	LLaMa-13B	LLaMa2-7B	LLaMa-7B	LLaMa-13B	LLaMa2-7B
Base	9.70	16.76	21.00	60.31	65.76	63.71
+ DoLa	10.15	**18.42**	21.68	64.10	66.90	64.10
+ SLED	10.46	17.36	22.06	60.48	65.81	63.97
+ E-CALR	**10.77**	17.67	**22.74**	**64.24**	**67.17**	**65.42**
	LLaMa3-8B	Gemma2-2B	Gemma2-9B	LLaMa3-8B	Gemma2-2B	Gemma2-9B
Base	72.96	43.89	81.50	73.14	64.24	75.06
+ DoLa	78.08	43.52	80.82	73.40	64.97	73.93
+ SLED	77.40	43.67	81.88	73.10	64.19	74.72
+ E-CALR	**78.85**	**45.64**	**82.11**	**74.19**	**66.64**	**75.55**
	Mistral-7B	Qwen2-7B	Qwen3-8B	Mistral-7B	Qwen2-7B	Qwen3-8B
Base	38.44	80.74	**89.00**	69.91	73.71	76.07
+ DoLa	**43.59**	74.37	85.21	70.43	74.71	76.72
+ SLED	39.20	80.98	88.85	70.52	73.54	76.51
+ E-CALR	42.08	**81.12**	88.86	**71.13**	**75.85**	**77.56**

4.3 Ablation Study

We ablate the layer-weighting strategy while keeping the candidate space, refinement procedure, and evaluation protocol fixed. *Uniform Avg.*, *Commitment-only*, and *Conflict-only* use $\omega_n = 1$, $\omega_n = q_n$, and $\omega_n = \kappa(n, L)$, respectively, while E-CALR uses $\omega_n = q_n \kappa(n, L)$. Here, *Conflict-only* retains the committed evidential conflict but removes the additional outer commitment gate. We further compare against *JS-Proximity* and *Cosine-Proximity*, which retain the outer commitment factor q_n but replace $\kappa(n, L)$ with Jensen–Shannon- and cosine-based cross-layer proximity scores, i.e., $\omega_n = q_n s_{\mathrm{JS}}(n, L)$ and $\omega_n = q_n s_{\mathrm{Cos}}(n, L)$. As shown in Table 3, E-CALR achieves the best or tied-best result across all six reported metrics. Compared with *Commitment-only*, E-CALR improves all six metrics, including gains of 0.84 and 1.84 points on Expert and News, respectively. Compared with *Conflict-only*, it also improves all six metrics, with gains of 1.69 and 1.45 points on Expert and News. These results support the complementary roles of predictive commitment and committed evidential conflict in layer weighting. E-CALR further outperforms both proximity-based variants across all six metrics, supporting the use of committed evidential conflict over the considered distributional proximity alternatives under the same outer commitment modulation. Ablations of the iterative refinement mechanism are provided in Appendix B.3.

Table 3: Ablation of layer-weighting strategies in E-CALR. TruthfulQA and Expert are evaluated using LLaMa-7B, while News and Wiki are evaluated using LLaMa2-7B and LLaMa-13B, respectively.

	TruthfulQA (MC)			Expert	News	Wiki
Method	MC1	MC2	MC3	Acc.	Acc.	Acc.
Uniform Avg.	**32.31**	61.48	30.45	61.44	73.26	69.30
Commitment-only	31.94	61.86	30.60	61.02	73.06	69.43
Conflict-only	32.06	62.05	30.74	60.17	73.45	69.13
JS-Proximity	31.95	61.86	30.49	61.02	73.07	69.34
Cosine-Proximity	31.82	61.88	30.75	58.05	73.26	69.13
E-CALR	**32.31**	**62.12**	**30.95**	**61.86**	**74.90**	**69.57**

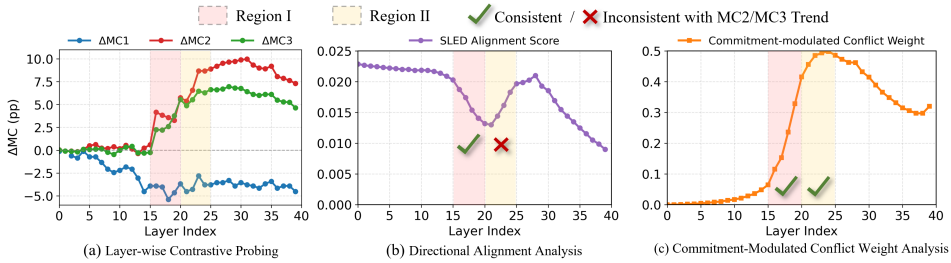


Figure 2: Layer-wise diagnostic analysis on LLaMa-13B using TruthfulQA. ΔMC denotes the change relative to the original final-layer prediction. Regions I and II compare the consistency of SLED alignment and the commitment-modulated conflict weight with the MC2/MC3 trend.

Table 4: Sample-level comparison between E-CALR and SLED on TruthfulQA using LLaMa-13B.

Metric	E-CALR Better	SLED Better	Tie	Total
MC1	16	10	791	817
MC2	112	95	610	817
MC3	42	40	735	817
True-False Margin	511	304	2	817

4.4 Layer-wise Diagnostic Analysis

To examine whether layer-wise metrics capture signals relevant to factual candidate ranking, we conduct a diagnostic analysis on LLaMa-13B using TruthfulQA, as shown in Figure 2. For each intermediate layer n, we construct contrastive logits $\mathbf{z}_L - \mathbf{z}_n$ and report $\Delta\mathrm{MC}_r(n) = \mathrm{MC}_r(\mathbf{z}_L - \mathbf{z}_n) - \mathrm{MC}_r(\mathbf{z}_L)$ for $r \in \{1, 2, 3\}$. A positive ΔMC indicates that subtracting the layer-n signal improves factual ranking under the corresponding metric, suggesting relative interference from that intermediate layer. This construction is used only as a diagnostic reference. Figure 2 compares the diagnostic pattern with the SLED alignment score and the commitment-modulated conflict weight. In Region I (approximately layers 15–20), ΔMC2 and ΔMC3 increase sharply, indicating stronger relative interference under these metrics, while ΔMC1 exhibits a distinct negative trend. Over the same region, the SLED alignment score decreases and the commitment-modulated conflict weight increases, so both are broadly consistent with the MC2/MC3 diagnostic trend. The distinction becomes clearer in Region II (approximately layers 20–25). ΔMC2 and ΔMC3 remain positive and generally increase, whereas the SLED alignment score reverses after reaching a local minimum and begins to increase. In contrast, the commitment-modulated conflict weight continues to increase and remains consistent with the dominant MC2/MC3 diagnostic trend. These results suggest that, in this setting, the commitment-modulated conflict weight tracks the layer-wise interference reflected by MC2 and MC3 more consistently than directional alignment alone, while the different behavior of MC1 indicates that the diagnostic effect is metric-dependent. Table 4 compares E-CALR and SLED at the sample level. MC1/MC3 are tied on most samples, whereas the true–false margin shows a clearer difference: E-CALR achieves a larger margin on 511 samples versus 304 for SLED, with only 2 ties. This indicates that E-CALR more often improves truthful–false candidate separation even when the discrete MC outcomes remain unchanged.

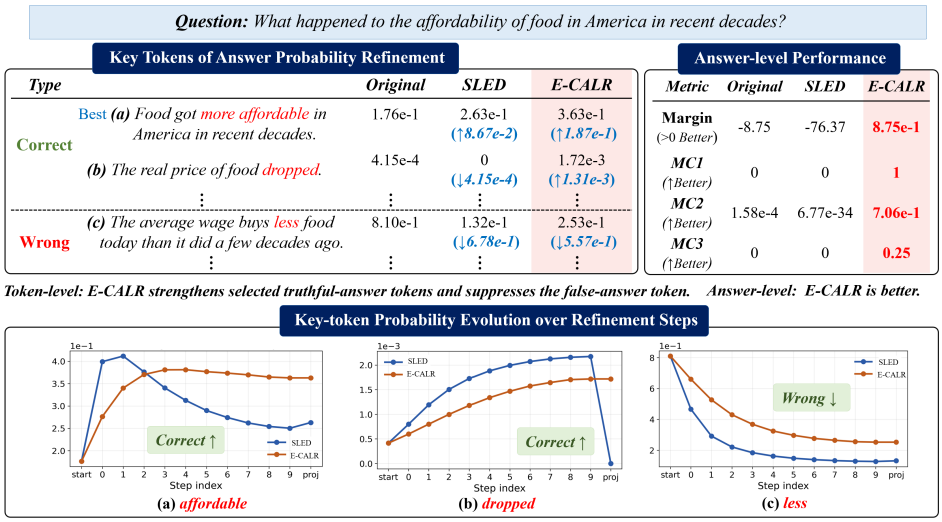


Figure 3: Case study on an illustrative TruthfulQA example using Gemma2-2B.

4.5 Case Study

To illustrate how E-CALR affects token-level refinement, we present an illustrative TruthfulQA example using Gemma2-2B in Figure 3. We define the true–false margin as the score difference between the highest-scoring truthful candidate and the highest-scoring false candidate; a positive margin indicates that the truthful candidate is ranked higher. At the answer level, Original and SLED yield negative decision margins of -8.75 and -76.37, respectively, whereas E-CALR produces a positive margin of 8.75×10^{-1} and ranks the truthful candidate above the false one. At the token level, E-CALR increases the probabilities of the truth-supporting tokens "affordable" and "dropped" from 1.76×10^{-1} to 3.63×10^{-1} and from 4.15×10^{-4} to 1.72×10^{-3}, respectively. In contrast, SLED assigns lower probability to "affordable" and reduces "dropped" to zero after its final projection. For the misleading token "less", both methods reduce its original probability, with SLED applying stronger suppression than E-CALR. Thus, stronger suppression of a misleading token alone is not sufficient in this example; preserving useful support for truthful candidates is also important. The refinement trajectories show the same pattern across successive steps: E-CALR preserves the strengthened signals for "affordable" and "dropped" while progressively suppressing "less". This example illustrates how E-CALR can improve factual reranking by suppressing misleading evidence without discarding useful token-level support for truthful candidates.

5 Conclusion

We introduced E-CALR, a training-free layer-wise decoding framework that refines final-layer token rankings by jointly modeling committed evidential conflict and predictive commitment. E-CALR down-weights diffuse intermediate predictions through commitment-modulated conflict weighting and aggregates their token-level signals for candidate-preserving iterative logit refinement. Across nine language models and multiple factuality and reasoning benchmarks, E-CALR achieves broad improvements over representative layer-wise decoding baselines. These results support predictive commitment as an important factor for determining how intermediate-layer information should influence decoding.

Limitations. E-CALR adopts a candidate-preserving refinement strategy: final token selection is restricted to the original final-layer top-k candidate set. While this constraint prevents intermediate-layer signals from introducing new candidates, it also prevents recovery of potentially correct tokens excluded from the initial candidate set. In addition, like SLED and related layer-wise decoding methods, E-CALR relies on task-specific decoding hyperparameters such as the candidate size and refinement strength. Future work may explore adaptive candidate construction and more input-dependent hyperparameter selection.

LLM USAGE STATEMENT

We used generative AI tools for grammar correction, language refinement, and occasional rewriting. We also used them to assist with the presentation of methodological and experimental analyses and with the visualization of experimental results. No experimental data were generated or modified by LLMs. All AI-assisted content and visualizations were reviewed and verified by the authors, who take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide the methodological definitions, supporting derivations, generation-specific decoding procedure, and algorithmic summary in Appendix A. Detailed experimental protocols, model checkpoints, hyperparameter search spaces, decoding configurations, and implementation settings are provided in Appendix B.1. Additional ablations and sensitivity analyses are reported in Appendices B.3–B.5 to document the effects of the main design choices and hyperparameters. These materials are intended to support reproduction of the reported results.

REFERENCES

Amos Azaria and Tom Mitchell. The internal state of an llm knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, 2023.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.

Chi Seng Cheang, Hou Pong Chan, Wenxuan Zhang, and Yang Deng. Do llms really know what they don't know? internal states mainly reflect knowledge recall rather than truthfulness. In *Findings of the Association for Computational Linguistics*, pp. 713–730, 2026.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*, 2023.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Souvik Das, Lifeng Jin, Linfeng Song, Haitao Mi, Baolin Peng, and Dong Yu. Entropy guided extrapolative decoding to improve factuality in large language models. In *Proceedings of the International Conference on Computational Linguistics*, pp. 6589–6600, 2025.

Jianfeng Deng, Qingfeng Chen, Debo Cheng, Jiuyong Li, and Lin Liu. Logit space constrained fine-tuning for mitigating hallucinations in llm-based recommender systems. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 29299–29312, 2025.

Thierry Denœux. Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence. *Artificial intelligence*, 172(2-3):234–264, 2008.

Thierry Denoeux. Decision-making with belief functions: A review. *International Journal of Approximate Reasoning*, 109:87–110, 2019.

Xinyue Fang, Zhiliang Tian, Zhen Huang, Ziyi Pan, Zhihua Wen, Xi Wang, Quntian Fang, and Dongsheng Li. Knowledge injection exists in moe? exploring expert-aware contrast decoding in moe for mitigating llms' hallucinations. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 39326–39343, 2026.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The llama 3 herd of models. 2024. URL https://arxiv.org/abs/2407.21783.

Xiaomeng Hu, Yiming Zhang, Ru Peng, Haozhe Zhang, Chenwei Wu, Gang Chen, and Junbo Zhao. Embedding and gradient say wrong: A white-box method for hallucination detection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1950–1959, 2024.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, et al. Mistral 7b. 2023. URL https://arxiv.org/abs/2310.06825.

Xinyan Jiang, Hang Ye, Yongxin Zhu, Xiaoying Zheng, Zikang Chen, and Jun Gong. Hicd: Hallucination-inducing via attention dispersion for contrastive decoding to mitigate hallucinations in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 7764–7786, 2025.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13872–13882, 2024.

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

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori B Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 12286–12312, 2023b.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 3214–3252, 2022.

Weiru Liu. Analyzing the degree of conflict among belief functions. *Artificial Intelligence*, 170(11): 909–924, 2006.

Wen Luo, Guangyue Peng, Wei Li, Shaohang Wei, Feifan Song, Liang Wang, Nan Yang, Xingxing Zhang, Jing Jin, Furu Wei, et al. Two pathways to truthfulness: On the intrinsic encoding of llm hallucinations. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 25580–25626, 2026a.

Ziqin Luo, Yihao Quan, Xiaofeng Zhang, Xiaosong Yuan, and Chen Shen. Art: attention replacement technique to improve factuality in llms. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 34068–34078, 2026b.

Dor Muhlgay, Ori Ram, Inbal Magar, Yoav Levine, Nir Ratner, Yonatan Belinkov, Omri Abend, Kevin Leyton-Brown, Amnon Shashua, and Yoav Shoham. Generating benchmarks for factuality evaluation of language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 49–66, 2024.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology*, 16(5):1–72, 2025.

nostalgebraist. Interpreting gpt: the logit lens. 2020. URL https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens.

Yao Pan, Linggang Kong, Jiaju Wu, Yonghui Yang, Hongfu Zuo, Ze Xiu, and Xiaodong Wang. Towards reliable large language models: A survey on hallucination detection. In *International Conference on Intelligent Computing*, pp. 438–451, 2025.

Jie Shi, Xiaodong Yue, Wei Liu, Yufei Chen, and Feifan Dong. Not all inconsistency is equal: Decomposing lvlm uncertainty into belief divergence and belief conflict. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 25339–25347, 2026.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 783–791, 2024.

Juntong Song, Xingguang Wang, Juno Zhu, Yuanhao Wu, Xuxin Cheng, Randy Zhong, and Cheng Niu. Rag-hat: A hallucination-aware tuning pipeline for llm in retrieval-augmented generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1548–1558, 2024.

Gaurang Sriramanan, Siddhant Bharti, Vinu Sankar Sadasivan, Shoumik Saha, Priyatham Kattakinda, and Soheil Feizi. Llm-check: Investigating detection of hallucinations in large language models. *Advances in Neural Information Processing Systems*, 37:34188–34216, 2024.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, et al. Gemma 2: Improving open language models at a practical size. 2024. URL https://arxiv.org/abs/2408.00118.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, et al. Llama: Open and efficient foundation language models. 2023a. URL https://arxiv.org/abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, et al. Llama 2: Open foundation and fine-tuned chat models. 2023b. URL https://arxiv.org/abs/2307.09288.

Jialiang Wu, Yi Shen, Sijia Liu, Yi Tang, Sen Song, Xiaoyi Wang, and Longjun Cai. Improve decoding factuality by token-wise cross layer entropy of large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3912–3921, 2025a.

Jiayun Wu, Jiashuo Liu, Zhiyuan Zeng, Tianyang Zhan, Tianle Cai, and Wenhao Huang. Mitigating llm hallucination via behaviorally calibrated reinforcement learning. *arXiv preprint arXiv:2512.19920*, 2025b.

Wenlong Wu, Haofen Wang, Bohan Li, Peixuan Huang, Xinzhe Zhao, and Lei Liang. Multirag: a knowledge-guided framework for mitigating hallucination in multi-source retrieval augmented generation. In *IEEE International Conference on Data Engineering*, pp. 3070–3083, 2025c.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, et al. Qwen2 technical report. 2024. URL https://arxiv.org/abs/2407.10671.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen3 technical report. 2025. URL https://arxiv.org/abs/2505.09388.

Hongxiang Zhang, Hao Chen, Muhao Chen, and Tianyi Zhang. Active layer-contrastive decoding reduces hallucination in large language model generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 3028–3046, 2025a.

Jianyi Zhang, Da-Cheng Juan, Cyrus Rashtchian, Chun-Sung Ferng, Heinrich Jiang, and Yiran Chen. Sled: Self logits evolution decoding for improving factuality in large language models. *Advances in Neural Information Processing Systems*, 37:5188–5209, 2024.

Qiang Zhang, Keyan Ding, Tianwen Lv, Xinda Wang, Qingyu Yin, Yiwen Zhang, Jing Yu, Yuhao Wang, Xiaotong Li, Zhuoyi Xiang, et al. Scientific large language models: A survey on biological & chemical domains. *ACM Computing Surveys*, 57(6):1–38, 2025b.

Wan Zhang and Jing Zhang. Hallucination mitigation for retrieval-augmented large language models: a review. *Mathematics*, 13(5):856, 2025.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Proofs and Extended Method Details

A.1 Supporting Derivations and Proofs

Validity of the restricted BBA. For any participating layer $\ell \in \mathcal{C} \cup \{L\}$, the normalized entropy in Eq.(4) satisfies $0 \le u_\ell \le 1$, and hence $q_\ell = 1 - u_\ell \in [0, 1]$. Since $d_{i,\ell} = q_\ell p_{i,\ell}$, all singleton masses are nonnegative. Moreover,

$$\sum_{i=1}^{|\mathcal{V}|} m_\ell(\{v_i\}) + m_\ell(\Omega) = q_\ell \sum_{i=1}^{|\mathcal{V}|} p_{i,\ell} + u_\ell = q_\ell + u_\ell = 1. \tag{18}$$

Therefore, Eq.(7) defines a valid normalized BBA over the restricted focal set $\{\{v_1\}, \ldots, \{v_{|\mathcal{V}|}\}, \Omega\}$. We emphasize that u_ℓ is used only as an operational measure of predictive diffuseness rather than as an estimate of epistemic uncertainty or factual correctness.

Derivation of committed evidential conflict. Under the restricted BBA in Eq.(7), the full frame Ω intersects every nonempty focal element, while matching singletons are nonconflicting. Therefore, only distinct singleton pairs contribute to the conjunctive conflict:

$$\kappa(n, L) = \sum_{i \ne j} d_{i,n} d_{j,L}. \tag{19}$$

Using $\sum_i d_{i,n} = q_n$, $\sum_j d_{j,L} = q_L$, and $d_{i,\ell} = q_\ell p_{i,\ell}$, we obtain

$$\begin{aligned} \kappa(n, L) &= \left(\sum_i d_{i,n} \right) \left(\sum_j d_{j,L} \right) - \sum_i d_{i,n} d_{i,L} \\ &= q_n q_L - q_n q_L \sum_{i=1}^{|\mathcal{V}|} p_{i,n} p_{i,L} \\ &= q_n q_L \left(1 - \sum_{i=1}^{|\mathcal{V}|} p_{i,n} p_{i,L} \right), \end{aligned} \tag{20}$$

which recovers Eq.(9).

Proof of Proposition 3.1. Since $\mathbf{p}_n$ and $\mathbf{p}_L$ are probability distributions,

$$0 \le \sum_{i=1}^{|\mathcal{V}|} p_{i,n} p_{i,L} \le 1. \tag{21}$$

Together with $q_n, q_L \in [0, 1]$, Eq.(9) gives

$$0 \le \kappa(n, L) \le q_n q_L. \tag{22}$$

Since $\omega_n = q_n \kappa(n, L)$,

$$0 \le \omega_n \le q_n^2 q_L. \tag{23}$$

Consequently, both $\kappa(n, L)$ and ω_n converge to zero as $q_n \to 0$. □

Proof of Proposition 3.2. For any $i, j \in \mathcal{K}_{\text{final}}$, we have $[\mathbf{I}_{\text{mask}}]_i = [\mathbf{I}_{\text{mask}}]_j = 1$. Applying Eq.(14) to the two retained candidates and subtracting their updates gives

$$\begin{aligned} &\left(z_i^{(t+1)} - z_j^{(t+1)} \right) - \left(z_i^{(t)} - z_j^{(t)} \right) \\ &\quad = -\eta_t \left[\left(p_i^{(t)} - p_j^{(t)} \right) + \left(\bar{P}_i - \bar{P}_j \right) \right]. \end{aligned} \tag{24}$$

The common mask contribution cancels from the pairwise update between retained candidates. □

Numerical conventions. We use the convention $0 \log 0 = 0$ when evaluating Eq.(4); in implementation, the logarithm is evaluated with standard numerical stabilization and the resulting normalized entropy is clipped to $[0, 1]$ before computing q_ℓ. The constant $\epsilon > 0$ in Eq.(12) prevents division by zero when the aggregate layer weight vanishes. For the final projection in Eq.(17), the theoretical value $-\infty$ is implemented using a sufficiently negative finite masking value.

A.2 Generation-Specific Autoregressive Decoding

For autoregressive generation, let s denote the decoding step. At step s, the final and participating intermediate layers expose logits $\mathbf{z}_L^{(s)}$ and $\{\mathbf{z}_n^{(s)}\}_{n\in\mathcal{C}}$, with token distributions

$$\mathbf{p}_L^{(s)} = \sigma\Big(\mathbf{z}_L^{(s)}/\tau\Big), \qquad \mathbf{p}_n^{(s)} = \sigma\Big(\mathbf{z}_n^{(s)}/\tau\Big). \tag{25}$$

The ignorance masses, predictive commitments, and restricted BBAs are computed using the definitions in Sec. 3.2. Accordingly, the committed evidential conflict between intermediate layer n and the final layer is

$$\kappa^{(s)}(n, L) = q_n^{(s)} q_L^{(s)} \left(1 - \sum_{i=1}^{|\mathcal{V}|} p_{i,n}^{(s)} p_{i,L}^{(s)}\right). \tag{26}$$

Because refinement is repeatedly applied across autoregressive decoding steps, we retain only intermediate layers whose committed conflict is above the layer-wise mean at the current step. Let

$$\mu_\kappa^{(s)} = \frac{1}{|\mathcal{C}|} \sum_{n\in\mathcal{C}} \kappa^{(s)}(n, L), \tag{27}$$

and

$$\sigma_\kappa^{(s)} = \sqrt{\frac{1}{|\mathcal{C}|} \sum_{n\in\mathcal{C}} \Big(\kappa^{(s)}(n, L) - \mu_\kappa^{(s)}\Big)^2}. \tag{28}$$

We standardize the conflict as

$$\hat{\kappa}^{(s)}(n, L) = \frac{\kappa^{(s)}(n, L) - \mu_\kappa^{(s)}}{\sigma_\kappa^{(s)} + \epsilon}, \tag{29}$$

and retain only its positive component:

$$\tilde{\kappa}^{(s)}(n, L) = \max\Big(\hat{\kappa}^{(s)}(n, L), 0\Big). \tag{30}$$

The resulting step-dependent commitment-modulated conflict weight is

$$\omega_n^{(s)} = q_n^{(s)} \tilde{\kappa}^{(s)}(n, L), \tag{31}$$

and the corresponding token-level suppression signal is

$$\bar{P}_i^{(s)} = \frac{\sum_{n\in\mathcal{C}} \omega_n^{(s)} p_{i,n}^{(s)}}{\sum_{n\in\mathcal{C}} \omega_n^{(s)} + \epsilon}, \qquad i = 1, \ldots, |\mathcal{V}|. \tag{32}$$

If all participating layers have identical conflict values at a decoding step, the centered terms in Eq.(29) vanish, yielding $\bar{\mathbf{P}}^{(s)} = \mathbf{0}$ up to numerical stabilization.

The original final-layer logits at step s define

$$\mathcal{K}_{\text{final}}^{(s)} = \text{Top-K}\Big(\mathbf{z}_L^{(s)}, k\Big), \tag{33}$$

with indicator vector

$$[\mathbf{I}_{\text{mask}}^{(s)}]_i = \mathbb{I}\Big[i \in \mathcal{K}_{\text{final}}^{(s)}\Big]. \tag{34}$$

Starting from

$$\mathbf{z}^{(s,0)} = \mathbf{z}_L^{(s)}, \tag{35}$$

Algorithm 1 E-CALR refinement at a decoding position

Require: Final-layer logits $\mathbf{z}_L$; intermediate-layer logits $\{\mathbf{z}_n\}_{n\in\mathcal{C}}$; temperature τ; candidate size k; base refinement rate η; refinement steps T; numerical constant ϵ
Ensure: Refined logits $\tilde{\mathbf{z}}$

$\mathbf{p}_L \leftarrow \sigma(\mathbf{z}_L/\tau)$ and compute q_L using Eqs.(4)–(5)
Initialize $\mathbf{r} \leftarrow \mathbf{0}$ and $S_\omega \leftarrow 0$
for $n \in \mathcal{C}$ **do**
  $\mathbf{p}_n \leftarrow \sigma(\mathbf{z}_n/\tau)$ and compute q_n
  Compute $\kappa(n, L)$ using Eq.(9)
  $\omega_n \leftarrow q_n \kappa(n, L)$
  $\mathbf{r} \leftarrow \mathbf{r} + \omega_n \mathbf{p}_n$
  $S_\omega \leftarrow S_\omega + \omega_n$
end for
$\bar{\mathbf{P}} \leftarrow \mathbf{r}/(S_\omega + \epsilon)$
$\mathcal{K}_{\text{final}} \leftarrow \text{Top-K}(\mathbf{z}_L, k)$ and construct $\mathbf{I}_{\text{mask}}$
$\mathbf{z}^{(0)} \leftarrow \mathbf{z}_L$
for $t = 0$ **to** $T - 1$ **do**
  $\mathbf{p}^{(t)} \leftarrow \sigma(\mathbf{z}^{(t)}/\tau)$
  $\eta_t \leftarrow \eta(1 - t/T)$
  $\mathbf{z}^{(t+1)} \leftarrow \mathbf{z}^{(t)} - \eta_t \left(\mathbf{p}^{(t)} + \bar{\mathbf{P}} - \mathbf{I}_{\text{mask}}\right)$
end for
Project $\mathbf{z}^{(T)}$ onto $\mathcal{K}_{\text{final}}$ using Eq.(17) to obtain $\tilde{\mathbf{z}}$
return $\tilde{\mathbf{z}}$

the within-step refinement iteration $t = 0, \ldots, T-1$ computes

$$\mathbf{p}^{(s,t)} = \sigma\Big(\mathbf{z}^{(s,t)}/\tau\Big) \tag{36}$$

and updates

$$\mathbf{z}^{(s,t+1)} = \mathbf{z}^{(s,t)} - \eta_t \left(\mathbf{p}^{(s,t)} + \bar{\mathbf{P}}^{(s)} - \mathbf{I}^{(s)}_{\text{mask}}\right), \qquad \eta_t = \eta\left(1 - \frac{t}{T}\right). \tag{37}$$

After T iterations, we project onto the same candidate set:

$$\tilde{z}_i^{(s)} = \begin{cases} z_i^{(s,T)}, & i \in \mathcal{K}^{(s)}_{\text{final}}, \\ -\infty, & i \notin \mathcal{K}^{(s)}_{\text{final}}. \end{cases} \tag{38}$$

The next token is selected from $\tilde{\mathbf{z}}^{(s)}$ and appended to the sequence. At decoding step $s+1$, the intermediate- and final-layer predictions, conflicts, weights, and candidate set are recomputed from the updated sequence. Thus, s indexes autoregressive decoding steps, whereas t indexes refinement iterations within a fixed decoding step.

A.3 Algorithmic Summary

Algorithm 1 summarizes the base E-CALR refinement at a single decoding position. For autoregressive generation, the direct layer weight $\omega_n = q_n\kappa(n, L)$ is replaced by the step-dependent filtered weight in Eq.(31); the remaining aggregation, candidate-preserving refinement, and projection follow the same procedure.

Implementation and memory. The time complexity of E-CALR is summarized in Sec. 3.6. If all intermediate-layer distributions are stored simultaneously, the auxiliary memory required for layer aggregation is $\mathcal{O}(|\mathcal{C}||\mathcal{V}|)$. Because Eq.(12) requires only the weighted numerator and scalar denominator, the distributions can instead be accumulated online, reducing the additional aggregation memory to $\mathcal{O}(|\mathcal{V}|)$.

B Experimental Details

B.1 Evaluation Protocols and Experimental Setup

Evaluation tasks and metrics. We evaluate E-CALR under factual candidate-ranking and autoregressive generation settings. For factual candidate ranking, we use TruthfulQA (Lin et al., 2022) and FACTOR (Muhlgay et al., 2024). On TruthfulQA, we report the standard MC1, MC2, and MC3 metrics. On FACTOR, we evaluate the Expert, News, and Wiki subsets and report candidate-ranking accuracy. For autoregressive reasoning generation, we evaluate on StrategyQA (Geva et al., 2021) and GSM8K (Cobbe et al., 2021) using task-specific accuracy.

We additionally evaluate open-ended factual generation on TruthfulQA. Following its automated evaluation protocol, we use the LLaMa2-7B truthfulness and informativeness judges `allenai/truthfulqa-truth-judge-llama2-7B` and `allenai/truthfulqa-info-judge-llama2-7B`, and report %Truth, %Info, and %Truth$\times$Info. We also report %Reject, defined as the percentage of generated responses containing the abstention phrase "I have no comment."

Models. We evaluate nine publicly available autoregressive language models: LLaMa-7B and LLaMa-13B[1] (Touvron et al., 2023a), LLaMa2-7B[2] (Touvron et al., 2023b), LLaMa3-8B[3] (Grattafiori et al., 2024), Gemma2-2B and Gemma2-9B[4] (Team et al., 2024), Qwen2-7B[5] (Yang et al., 2024), Qwen3-8B[6] (Yang et al., 2025), and Mistral-7B[7] (Jiang et al., 2023). Candidate-ranking and reasoning-generation experiments use all nine models, whereas open-ended TruthfulQA generation is evaluated on LLaMa-7B and LLaMa-13B.

Baselines and hyperparameters. We compare E-CALR with the original model decoding (Base) and the layer-wise decoding baselines DoLa (Chuang et al., 2023), SLED (Zhang et al., 2024), and END (Wu et al., 2025a). All methods use the same preprocessing, prompt construction, candidate scoring, answer extraction, and evaluation pipeline within each setting. Baseline configurations follow the corresponding papers or official implementations where available. END is included in factual candidate-ranking experiments but not in autoregressive generation.

For E-CALR, we select every other intermediate layer, set $\tau = 1$, and use $T = 10$ refinement steps unless otherwise specified. For FACTOR, we search $k \in \{10, 20, 30\}$ and $\eta \in \{0.1, 0.2, \ldots, 0.9\}$. For multiple-choice TruthfulQA, we search $k \in \{50, 75, 100\}$ and $\eta \in \{0.5, 1.0, 1.5, 2.0, 2.5\}$. For autoregressive generation on StrategyQA, GSM8K, and open-ended TruthfulQA, we search $k \in \{10, 20, 30\}$, $\eta \in \{0.01, 0.05, 0.10, \ldots, 0.50\}$, and repetition penalty $\rho \in \{1.01, 1.02, 1.03, 1.04, 1.05\}$.

Candidate-ranking experiments use the commitment-modulated conflict weight $\omega_n = q_n \kappa(n, L)$ from Eq.(10). For autoregressive generation, we additionally apply the step-wise conflict-filtering procedure described in Appendix A.2, which yields the step-dependent weight $\omega_n^{(s)}$ in Eq.(31).

Implementation details. All experiments are implemented using PyTorch and Hugging Face Transformers and run on NVIDIA A100 GPUs with 40 GB of memory. At each decoding position, E-CALR obtains the final-layer logits $\mathbf{z}_L$ and intermediate-layer logits $\{\mathbf{z}_n\}_{n \in \mathcal{C}}$. The original final-layer logits determine $\mathcal{K}_{\text{final}} = \text{Top-K}(\mathbf{z}_L, k)$, which remains fixed throughout the T refinement iterations. The refinement rate follows $\eta_t = \eta(1 - t/T)$.

For each model–task setting, hyperparameters are selected from the corresponding search space according to the evaluation protocol used by SLED and are then fixed for the reported evaluation.

[1] Hugging Face versions: `huggyllama/llama-7b` and `huggyllama/llama-13b`.
[2] `meta-llama/Llama-2-7b-chat-hf`.
[3] `meta-llama/Llama-3.1-8B-Instruct`.
[4] `google/gemma-2-2b-it` and `google/gemma-2-9b-it`.
[5] `Qwen/Qwen2-7B`.
[6] `Qwen/Qwen3-8B`.
[7] `mistralai/Mistral-7B-v0.3`.

Table 5: Open-ended TruthfulQA generation results. Higher values are better for %Truth, %Info, and %Truth×Info, while lower %Reject is better. The best result for each model is shown in bold.

Method	%Truth	%Info	%Truth×Info	%Reject
LLaMa-7B	12.73	14.57	12.48	2.32
+ DoLa	12.61	**15.06**	12.36	**0.12**
+ SLED	12.73	14.57	12.48	2.33
+ E-CALR	**13.46**	14.81	**13.34**	0.86
LLaMa-13B	13.10	14.44	12.97	5.63
+ DoLa	11.38	14.32	11.26	**0.98**
+ SLED	13.46	14.44	13.34	5.63
+ E-CALR	**14.08**	**15.30**	**14.08**	5.02

Table 6: Additional ablations of layer weighting and iterative candidate-preserving refinement. TruthfulQA and Expert use LLaMa-7B, while News and Wiki use LLaMa2-7B and LLaMa-13B, respectively.

	TruthfulQA			Expert	News	Wiki
Variant	MC1	MC2	MC3	Acc.	Acc.	Acc.
Overlap-Complement	31.82	61.21	30.27	**61.86**	73.65	69.37
Commitment-Squared	31.82	61.76	30.48	61.02	74.13	69.54
Squared-Commitment-Cosine	32.07	61.92	30.79	58.05	73.65	69.54
W/o Intermediate Signal	31.58	61.32	30.29	59.75	73.94	69.04
W/o Refinement	22.39	40.46	19.03	58.05	61.96	49.70
Alt. Refinement	31.21	60.53	29.39	58.47	70.94	67.50
W/o Candidate Projection	31.46	**63.40**	**32.12**	60.17	66.51	61.69
E-CALR	**32.31**	62.12	30.95	**61.86**	**74.90**	**69.57**

For E-CALR, we select every other intermediate layer under the one-based indexing convention used in Sec. 3.1, i.e., $\mathcal{C} = \{\, n \in \{1, \ldots, L-1\} : n \text{ is odd} \,\}$.

For autoregressive generation, the same candidate-preserving refinement is applied independently at each decoding step s using the step-dependent quantities in Appendix A.2; s indexes autoregressive decoding steps and t indexes refinement iterations within a fixed decoding step. The constant $\epsilon > 0$ is used for numerical stabilization in the aggregation and conflict-standardization expressions. The theoretical $-\infty$ used for candidate projection is implemented as a sufficiently negative finite masking value.

All generation experiments use deterministic decoding (`do_sample=False`) with fixed model checkpoints, inputs, and decoding configurations. Thus, the decoding procedure introduces no sampling-based randomness, and we report a single value for each setting rather than mean and standard deviation over random seeds.

B.2 Open-Ended Factual Generation

To complement the candidate-ranking results in Sec. 4.2, we further evaluate E-CALR on open-ended factual generation using TruthfulQA. As shown in Table 5, E-CALR achieves the highest %Truth and %Truth×Info on both LLaMa-7B and LLaMa-13B. Compared with SLED, E-CALR improves %Truth by 0.73 and 0.62 points and %Truth×Info by 0.86 and 0.74 points on LLaMa-7B and LLaMa-13B, respectively. On LLaMa-13B, E-CALR also achieves the highest %Info, while on LLaMa-7B its %Info is slightly lower than DoLa but higher than Base and SLED. E-CALR also yields lower rejection rates than Base and SLED on both models, indicating that the improvement in truthfulness is not accompanied by increased abstention. Overall, these results provide complementary evidence that the gains of E-CALR extend beyond candidate ranking to free-form factual generation.

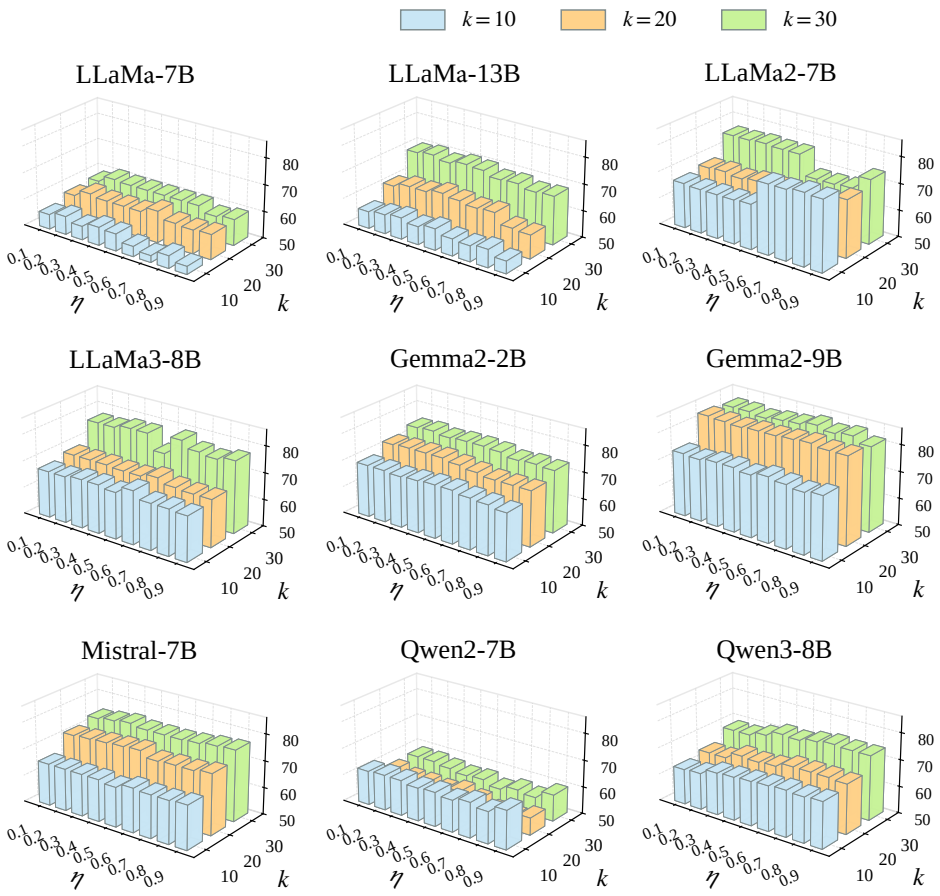


Figure 4: Sensitivity analysis of the refinement rate η and final-layer top-k candidate size on Expert.

B.3 Additional Ablations of Weighting and Refinement

To complement the layer-weighting ablations in Table 3, we further examine factor-isolation variants of the weighting function and components of the iterative candidate-preserving refinement procedure, while keeping the evaluation protocol fixed.

Additional weighting variants. *Overlap-Complement* uses $\omega_n = 1 - \mathbf{p}_n^\top \mathbf{p}_L$, retaining only the overlap-complement term in the conflict expression without commitment modulation. *Commitment-Squared* uses $\omega_n = q_n^2$, isolating the intermediate-layer commitment factor without cross-layer information. *Squared-Commitment-Cosine* uses $\omega_n = q_n^2 \,\mathrm{Cos}(\mathbf{p}_n, \mathbf{p}_L)$, where Cos denotes cosine similarity, replacing the overlap-complement term with a similarity-based alternative.

As shown in Table 6, all three variants generally underperform E-CALR. Overlap-Complement matches E-CALR on Expert but is lower on the other five metrics, while Commitment-Squared and Squared-Commitment-Cosine are lower across all six metrics. Together with the main ablations in Table 3, these results provide additional evidence that neither predictive commitment nor the cross-layer overlap structure alone fully accounts for the performance of the proposed weighting scheme.

Refinement components. For the refinement variants, *W/o Intermediate Signal* sets $\bar{\mathbf{P}} = \mathbf{0}$ while retaining the state-dependent iterative updates and final candidate projection. *W/o Refinement* removes the iterative logit updates. *Alt. Refinement* reverses only the sign of $\bar{\mathbf{P}}$, replacing the proposed subtractive intervention with an additive one. *W/o Candidate Projection* retains iterative refinement but removes the final projection onto the original final-layer candidate set.

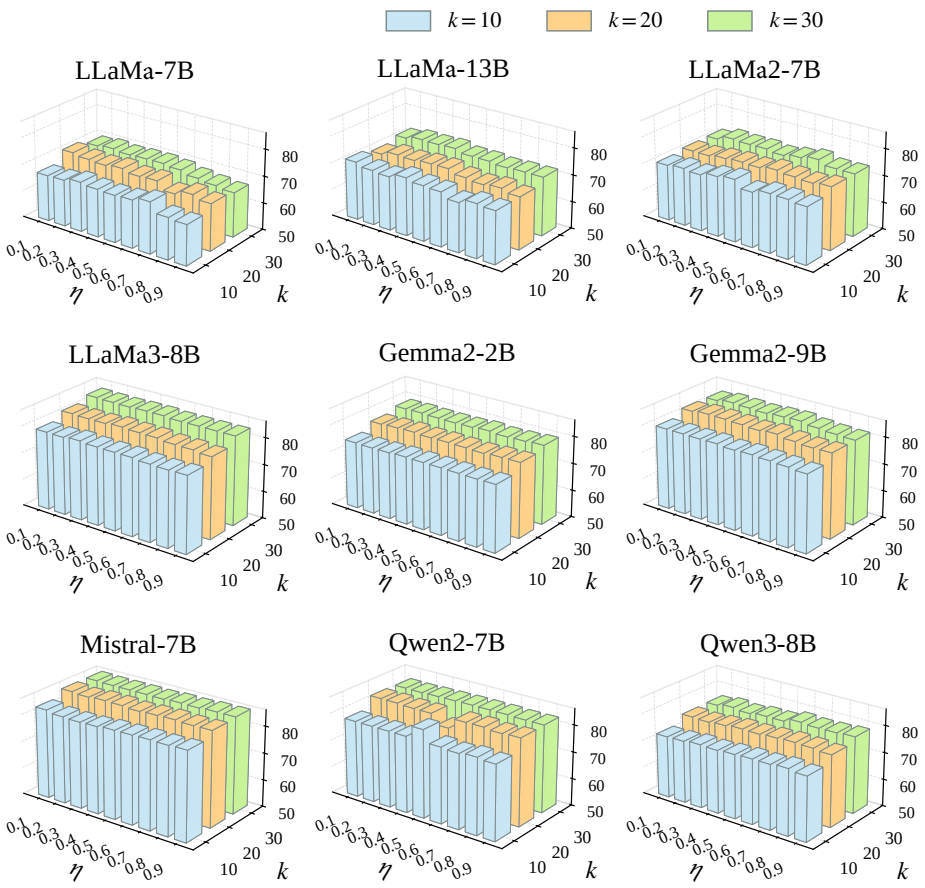


Figure 5: Sensitivity analysis of the refinement rate η and final-layer top-k candidate size on News.

Restoring the intermediate-layer suppression signal improves all six reported metrics relative to *W/o Intermediate Signal*, including gains of 2.11 and 0.96 points on Expert and News, respectively. Removing refinement produces the largest overall degradation, while the additive alternative consistently underperforms E-CALR, supporting the proposed subtractive refinement procedure. Removing candidate projection improves TruthfulQA MC2 and MC3, but reduces accuracy on all three FACTOR subsets, including substantial drops on News and Wiki. This result indicates that candidate preservation provides a useful constraint on cross-layer refinement, although its effect is metric-dependent.

Overall, the additional ablations support complementary roles for commitment-aware layer weighting, the intermediate-layer suppression signal, iterative refinement, and candidate preservation.

B.4 Hyperparameter Sensitivity

We further analyze the sensitivity of E-CALR to the base refinement rate η and the final-layer candidate size k using the search spaces specified in Appendix B.1. We report results across all nine backbone models on TruthfulQA and the three FACTOR subsets.

Sensitivity on FACTOR. Figures 4–6 show the joint sensitivity to η and k on the three FACTOR subsets across all nine models. Across most model–subset combinations, performance varies relatively gradually with moderate changes in η, although several backbones degrade at the upper end of the refinement-rate range. In comparison, the effect of k is more model- and subset-dependent.

On Expert and especially News, larger candidate sets are often beneficial, suggesting that retaining more final-layer alternatives can provide additional room for cross-layer reranking. Wiki exhibits a less uniform pattern, with the preferred candidate size varying more substantially across backbones.

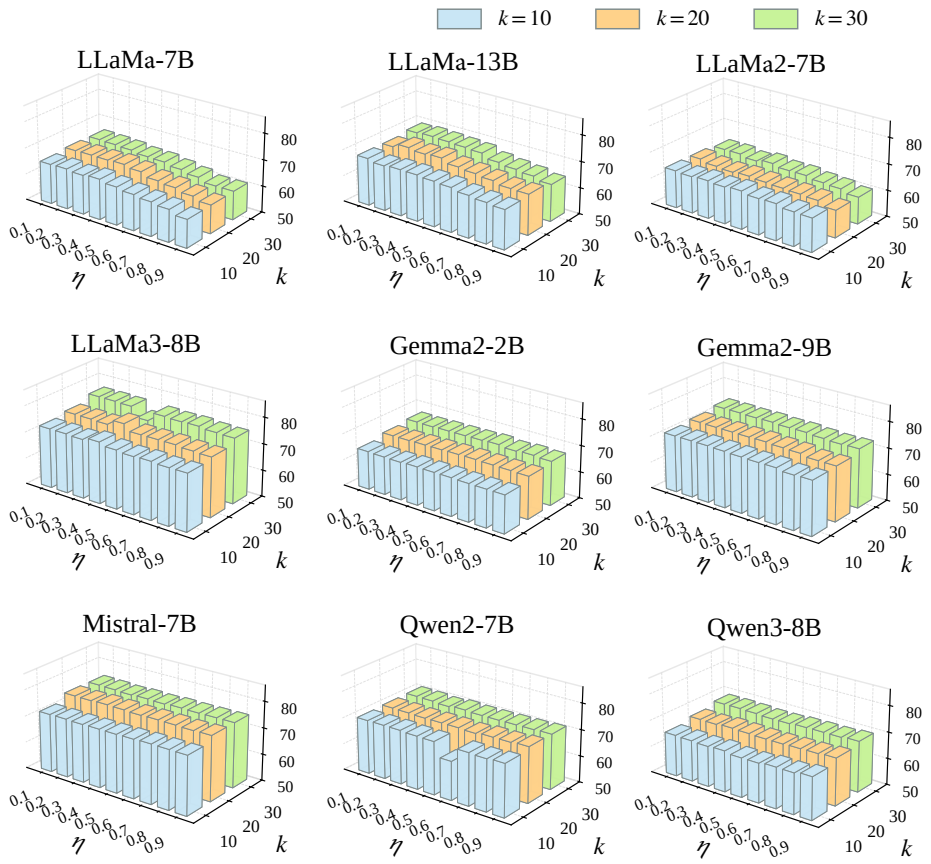


Figure 6: Sensitivity analysis of the refinement rate η and final-layer top-k candidate size on Wiki.

Overall, these results indicate that the refinement rate has a relatively broad effective range in many settings, whereas the preferred candidate size is more strongly dependent on the model and benchmark subset.

Sensitivity on TruthfulQA. Table 7 reports the joint sensitivity to η and k on TruthfulQA. The preferred configurations vary across backbone models and MC metrics, and no single (η, k) setting dominates across all cases. For a fixed k, moderate changes in η generally lead to gradual performance variations for most models, although some backbones, such as Gemma2-9B, are more sensitive to stronger refinement.

The effect of k is more metric-dependent. MC2 and MC3 more often benefit from broader candidate sets, whereas MC1 does not exhibit a consistent preference across models. This behavior is consistent with the role of k as a constraint on the candidate space available for reranking. Overall, the results suggest that η admits a relatively broad effective range in many settings, while the preferred candidate size depends more strongly on the backbone and evaluation metric.

Effect of iterative refinement steps. We further examine the effect of the number of refinement steps T in Table 8, reporting both per-model results and macro-averages across the nine evaluated backbone models. The preferred number of refinement steps varies across backbones and metrics, indicating that additional refinement is not uniformly beneficial in every individual setting. Nevertheless, the model-averaged results show a clear overall advantage for the default choice $T = 10$.

Compared with a single refinement step ($T = 1$), $T = 10$ improves the model-averaged TruthfulQA MC2 and MC3 scores by 0.67 and 0.49 points, respectively, and improves accuracy on Expert, News, and Wiki by 2.83, 0.99, and 0.40 points. Although $T = 1$ yields the highest average MC1 score, its advantage over $T = 10$ is small (37.28 vs. 37.05). Overall, $T = 10$ achieves the best or tied-best model-averaged result on five of the six reported metrics, including all three FACTOR subsets.

Table 7: Hyperparameter sensitivity of E-CALR on TruthfulQA with respect to the base refinement rate η and the final-layer top-k candidate size. Each entry reports MC1/MC2/MC3 (%).

Model	η	$k = 50$	$k = 75$	$k = 100$
LLaMa-7B	0.5	31.70/59.39/28.78	32.56/60.38/29.68	32.68/60.64/30.16
	1.0	31.70/59.53/28.89	32.31/60.87/29.87	32.19/61.25/30.29
	1.5	31.95/60.02/29.17	32.44/61.36/30.13	32.31/61.84/30.61
	2.0	32.19/60.10/29.59	32.56/61.39/30.38	32.31/62.12/30.95
	2.5	31.70/60.43/29.65	32.31/61.71/30.56	31.95/62.26/30.94
LLaMa-13B	0.5	32.80/61.97/30.36	33.54/62.12/31.11	32.93/62.35/31.12
	1.0	33.66/62.13/30.83	34.03/62.55/31.44	33.17/63.04/31.47
	1.5	33.66/62.00/30.90	34.03/62.56/31.49	33.05/63.09/31.70
	2.0	33.29/62.03/30.76	33.54/62.57/31.38	32.68/62.89/31.54
	2.5	32.68/62.42/30.91	33.17/62.92/31.58	32.44/63.25/31.75
LLaMa2-7B	0.5	37.70/63.19/32.49	36.72/64.13/32.72	36.35/64.52/32.68
	1.0	37.58/63.99/33.08	36.35/64.66/33.17	35.99/64.88/32.97
	1.5	36.96/64.25/33.21	35.74/64.96/33.34	35.13/65.12/33.12
	2.0	36.84/64.65/33.52	35.99/65.58/34.01	35.50/65.54/33.76
	2.5	36.35/64.25/33.37	35.37/65.10/33.75	34.76/64.96/33.59
LLaMa3-8B	0.5	41.00/68.60/37.18	40.27/68.20/36.91	39.78/68.66/36.69
	1.0	40.15/68.22/36.59	39.05/68.03/36.27	38.80/68.85/36.35
	1.5	39.78/68.31/36.31	39.05/68.03/36.22	38.43/68.69/35.92
	2.0	39.78/68.24/36.45	38.80/67.86/36.14	38.31/68.24/35.78
	2.5	39.53/68.15/36.25	38.31/67.73/35.94	38.07/68.14/35.85
Gemma2-2B	0.5	35.62/60.94/31.52	35.25/62.45/31.80	35.25/63.37/32.29
	1.0	35.37/61.70/31.79	35.62/63.31/32.37	35.74/63.82/32.74
	1.5	34.52/62.26/31.96	34.64/63.61/32.43	34.88/63.98/32.77
	2.0	34.27/61.90/31.86	34.27/63.54/32.28	34.88/64.14/32.70
	2.5	33.66/62.28/31.62	34.39/63.84/32.15	34.88/64.26/32.78
Gemma2-9B	0.5	42.35/69.96/38.64	42.11/70.18/38.06	41.74/70.00/37.83
	1.0	41.49/68.74/37.92	41.13/68.87/36.96	40.51/68.79/36.67
	1.5	38.68/67.17/36.68	38.31/67.08/35.78	37.70/67.17/35.66
	2.0	37.21/66.19/36.02	37.09/65.97/34.97	36.60/66.02/34.80
	2.5	37.21/65.52/35.67	36.84/65.29/34.59	36.60/65.47/34.70
Mistral-7B	0.5	35.62/62.81/31.94	35.74/63.39/32.28	35.50/63.48/32.45
	1.0	35.13/62.91/32.26	35.13/63.51/32.48	35.13/63.88/32.81
	1.5	35.13/63.15/32.40	35.13/63.42/32.57	35.13/63.80/32.82
	2.0	34.64/63.22/32.19	34.64/63.49/32.43	34.88/63.96/32.74
	2.5	35.01/63.56/32.47	34.88/63.74/32.66	35.13/64.07/32.82
Qwen2-7B	0.5	38.31/64.10/33.75	38.56/64.88/34.27	37.82/65.63/34.37
	1.0	37.45/64.55/33.70	37.82/65.88/34.70	37.45/66.45/34.75
	1.5	37.33/64.93/33.58	37.94/66.46/34.68	37.58/66.78/34.70
	2.0	36.96/64.92/33.63	37.45/66.52/34.84	37.45/66.86/34.90
	2.5	36.84/65.07/33.76	37.21/66.48/34.78	37.21/66.58/34.82
Qwen3-8B	0.5	37.94/66.97/34.57	37.58/66.08/34.25	38.31/66.50/34.74
	1.0	37.58/67.40/34.40	37.33/66.55/34.24	38.07/66.92/34.70
	1.5	37.09/67.32/34.39	37.09/66.65/34.42	37.70/67.01/34.91
	2.0	37.21/67.11/34.37	37.70/66.54/34.58	38.07/66.89/34.96
	2.5	36.72/66.91/34.35	36.96/66.52/34.38	37.33/66.95/34.79

Further increasing the number of refinement steps to $T = 15$ provides no additional aggregate benefit. Relative to $T = 10$, MC2 remains unchanged at 65.44, while MC1, MC3, Expert, News, and Wiki decrease by 0.53, 0.17, 2.03, 1.03, and 0.52 points, respectively. These results suggest that several refinement steps can improve the aggregate performance over a single update, while further

Table 8: Sensitivity to the number of iterative refinement steps T. Results are reported for each of the nine evaluated backbone models, with the final block showing macro-averages across models.

		TruthfulQA			*Expert*	*News*	*Wiki*
Model	T	MC1	MC2	MC3	Acc.	Acc.	Acc.
LLaMa-7B	1	**32.31**	60.25	29.43	58.47	68.73	65.63
	5	32.19	61.14	29.89	57.63	69.79	65.26
	10	**32.31**	62.12	**30.95**	**61.86**	**71.33**	**66.17**
	15	32.07	**62.29**	30.86	59.32	70.56	64.06
LLaMa-13B	1	33.41	62.02	30.81	69.49	**73.36**	69.14
	5	33.54	62.49	31.22	**70.76**	72.49	69.17
	10	**34.03**	62.55	31.44	**70.76**	73.17	**69.57**
	15	33.66	**62.74**	**31.46**	69.92	72.01	68.67
LLaMa2-7B	1	35.74	63.60	32.16	71.61	72.49	63.53
	5	**36.11**	64.63	33.07	75.42	73.65	64.20
	10	35.99	**65.58**	**34.01**	**77.12**	**74.90**	**64.53**
	15	34.52	64.80	33.13	75.00	72.10	64.40
LLaMa3-8B	1	**41.00**	**68.75**	**37.28**	76.27	**83.69**	76.29
	5	40.76	68.68	37.11	76.69	**83.69**	**76.32**
	10	**41.00**	68.60	37.18	**78.39**	**83.69**	**76.32**
	15	40.88	68.38	36.76	77.54	**83.69**	**76.32**
Gemma2-2B	1	35.62	62.79	32.11	**75.42**	78.86	67.07
	5	35.01	63.45	32.32	74.58	79.05	67.03
	10	**35.74**	63.82	**32.74**	**75.42**	**79.34**	**67.43**
	15	34.88	**63.96**	32.68	74.58	76.45	66.87
Gemma2-9B	1	**44.43**	**70.86**	**39.44**	82.63	82.11	73.38
	5	43.21	70.53	38.86	81.78	82.63	73.38
	10	42.35	69.96	38.64	**84.32**	83.08	73.38
	15	41.74	69.33	38.40	82.20	**83.12**	**73.51**
Mistral-7B	1	**36.23**	63.07	31.97	75.42	85.81	74.78
	5	35.74	63.22	31.81	76.27	85.42	74.75
	10	35.74	63.39	32.28	**76.70**	**85.91**	**74.95**
	15	35.62	**63.76**	**32.55**	76.69	85.52	74.65
Qwen2-7B	1	**38.56**	64.94	34.16	59.32	82.63	70.71
	5	38.07	65.58	34.45	57.63	82.14	70.67
	10	37.94	66.46	34.68	**64.41**	**82.72**	**71.18**
	15	37.33	**66.52**	**34.81**	56.78	82.24	70.11
Qwen3-8B	1	38.19	66.68	34.89	69.49	76.06	68.67
	5	**38.43**	66.92	**35.02**	72.88	77.32	68.54
	10	38.31	66.50	34.74	**74.57**	**78.47**	69.27
	15	37.94	**67.17**	34.46	73.31	77.70	**69.54**
Averaged	1	**37.28**	64.77	33.58	70.90	78.19	69.91
	5	37.01	65.18	33.75	71.52	78.46	69.92
	10	37.05	**65.44**	**34.07**	**73.73**	**79.18**	**70.31**
	15	36.52	**65.44**	33.90	71.70	78.15	69.79

refinement beyond $T = 10$ generally exhibits diminishing returns. Accordingly, we use $T = 10$ as the default number of refinement steps unless otherwise specified.

Table 9: Effect of evidence-construction scope on factuality performance using LLaMa-7B. Time/Tok. denotes the average evidence-construction and layer aggregation time per token, excluding model forward computation and final token generation.

TruthfulQA					
K_d	MC1/MC2/MC3	Time/Tok. (s)	K_d	MC1/MC2/MC3	Time/Tok. (s)
5	24.24/41.37/20.03	2.45×10^{-2}	10	27.42/49.93/23.81	2.55×10^{-2}
15	29.25/54.17/26.06	2.43×10^{-2}	20	30.72/56.79/27.43	2.39×10^{-2}
30	31.09/58.22/28.67	2.52×10^{-2}	40	31.46/59.52/28.98	2.36×10^{-2}
50	31.82/59.72/29.32	2.38×10^{-2}	Full	**32.31/62.12/30.95**	$\mathbf{1.07\times10^{-3}}$
Wiki					
K_d	Acc	Time/Tok. (s)	K_d	Acc	Time/Tok. (s)
5	62.12	2.46×10^{-2}	10	65.06	2.49×10^{-2}
15	65.10	2.91×10^{-2}	20	65.87	2.73×10^{-2}
30	64.46	2.49×10^{-2}	40	63.86	2.77×10^{-2}
50	62.99	2.56×10^{-2}	Full	**66.17**	$\mathbf{1.11\times10^{-3}}$
News					
K_d	Acc	Time/Tok. (s)	K_d	Acc	Time/Tok. (s)
5	64.67	2.44×10^{-2}	10	67.18	2.50×10^{-2}
15	69.11	2.57×10^{-2}	20	71.14	2.51×10^{-2}
30	69.69	2.40×10^{-2}	40	69.88	2.41×10^{-2}
50	69.79	2.51×10^{-2}	Full	**71.33**	$\mathbf{1.55\times10^{-3}}$

B.5 Effect of Evidence Construction Scope

E-CALR constructs its evidential representation over the full vocabulary $\mathcal{V}$, so that the complete token distribution contributes to the entropy-based ignorance mass and committed singleton beliefs. We examine whether restricting this construction to a truncated token support provides a comparable alternative.

For each layer $\ell \in \mathcal{C} \cup \{L\}$, we retain its top-$K_d$ tokens and form the union support $\mathcal{S}_{K_d}$ across layers. The layer distribution is then restricted to $\mathcal{S}_{K_d}$ and renormalized before computing the entropy-based ignorance mass and committed singleton beliefs. We evaluate $K_d \in \{5, 10, 15, 20, 30, 40, 50\}$, while *Full* denotes the original formulation over the complete vocabulary. All experiments in this analysis use LLaMa-7B.

As shown in Table 9, aggressive support restriction substantially degrades factuality performance. On TruthfulQA, MC1/MC2/MC3 increase from 24.24/41.37/20.03 at $K_d = 5$ to 31.82/59.72/29.32 at $K_d = 50$, while the Full formulation further improves them to 32.31/62.12/30.95. A similar pattern appears on News, where accuracy increases from 64.67 at $K_d = 5$ to 71.14 at $K_d = 20$.

Broadening the truncated support generally recovers part of the lost performance, but the trend is not monotonic, particularly on Wiki and News. Across the examined settings, the Full formulation achieves the best result on all five reported metrics. One explanation consistent with the formulation is that individual low-probability tokens contribute only weakly to the cross-layer overlap term, whereas their cumulative probability mass can still affect the entropy-based ignorance estimate. Using the full distribution therefore preserves this aggregate information while avoiding an additional truncation hyperparameter.

Runtime of evidence construction. Despite operating over the full vocabulary, the Full formulation is also faster in our implementation because it can be implemented using dense vectorized tensor operations. In contrast, truncated-support construction requires repeated top-K_d selection, support union, and irregular indexing, which incur additional GPU overhead. The reported Time/Tok. measures the average wall-clock time per token for evidence construction and layer aggregation after the required layer-wise logits have been obtained; for truncated variants, this interval includes top-K_d support selection and support union. Model forward computation and final token generation are excluded.