Attended yet Omitted: Context Omission Recovery via Internal Conflict Detection

Anonymous ACL submission

Abstract

Content omission, where Large Language Models (LLMs) omit targeted information, is a critical failure mode in long-context information extraction, posing practical risks in recall-critical applications such as legal review and regulatory compliance. Existing explanations mainly attribute omissions to the model's failure to attend to relevant content. Instead, we find that the attention correctly focuses on the omitted content, yet the decoding stage overrides this signal and skips the content. To ground our study in a realistic and challenging setting, we construct GovAudit, a compliance auditing dataset that demands exhaustive extraction of near-identical violations, comprising 9,577 samples and 63,625 violation triplets, from which baseline models omit over 91% of items. Based on our analysis, we further propose CORED (Context Omission Recovery via Conflict Detection), which trains a lightweight MLP monitor on internal attention and logit signals to detect conflicts during generation. Upon detection, CORED feeds the suppressed candidate tokens back into the prompt and triggers iterative re-generation to recover omitted content. CORED reduces omission on GovAudit from 91.4% to 31.4% and raises SQuAD F1 from 27.7 to 92.3, outperforming eight baselines across two backbones. Our data are available at `https://anonymous.4open.science/r/govaudit`.

1 Introduction

Content omission, where LLMs omit information that is present in the input and relevant to the query, is a critical failure mode in long-context information extraction tasks (Oukelmoun et al., 2025). Unlike hallucination (Si et al., 2025), which fabricates unsupported or incorrect content (Li et al., 2024), omission failures are insidious: neither the model nor the user receives any signal that something is missing (Liu et al., 2024). The consequences are particularly severe in recall-critical applications such as legal contract review, regulatory compliance auditing, where a single missed item can invalidate an entire analysis (Hendrycks et al., 2021; Jain et al., 2025; Dale et al., 2023). Figure 1 illustrates a representative case. The model is required to review contract clauses based on the provided regulations. Both Clauses 1 and 2 should be considered policy violations, as they contain descriptions prohibited by the policy. However, the model omits Clause 1, and only Clause 2 is detected.

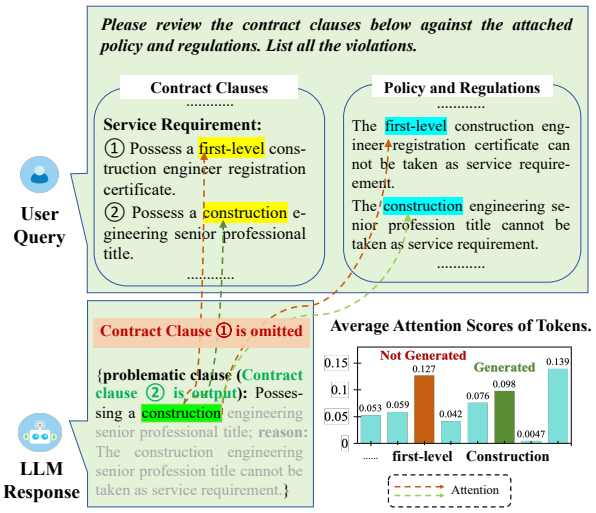


Figure 1: **Case Study of Content Omission**. Clauses 1 and 2 both violate the policy, yet the model omits Clause 1 and only detects Clause 2. The bar chart reveals an *Attention-Generation Inversion*: the policy-violation tokens (Red) in Clause 1 actually receive *higher* attention than the tokens (Green) in Clause 2, implying the omission is *not* due to a lack of attention.

The prevailing explanation for content omission centers on *retrieval failure*: the model fails to attend to relevant content, especially when it is buried in the middle of a long context (Liu et al., 2024; Kamradt, 2024; Chen et al., 2024; Lewis et al., 2020; Beltagy et al., 2020). We present evidence that this assumption is incomplete by identifying a recurring pattern: *the model's attention correctly focuses on the content it ultimately omits, yet the*

decoding stage suppresses this signal and skips the content. As shown at the bottom of Figure 1, the omitted token actually receives *higher* attention than the token the model generates. We term this phenomenon *Attention-Generation Inversion* and show in §3 that it accounts for a substantial fraction of omissions. This finding reveals generation failure as a significant source of content omission, complementing the established retrieval-failure explanation. Though prior work on hallucination has separately shown that LLMs internally encode correct information while generating incorrect outputs (Geva et al., 2022; Kadavath et al., 2022; Capuano et al., 2025), content omission is distinct as the model does not produce a wrong answer but simply fails to produce one, and the applicability of this insight to omission had not been established.

Studying this phenomenon at scale requires a dataset where content omission is both prevalent and measurable. To ground our analysis in a realistic and challenging setting, we construct *GovAudit*, a large-scale government procurement compliance auditing dataset comprising 9,577 expert-annotated samples with 63,625 violation triplets. The task demands exhaustive extraction: 93.8% of samples require identifying seven or more parallel violations from inputs averaging 25k tokens, many of which are near-identical and differ by only a single term. This dense parallel recall structure pushes omission rates to extremes: baseline models miss over 91% of required items, making GovAudit a challenging testbed for studying content omission.

Our analysis of GovAudit (§3) confirms that the attention signal persists in most omission cases, suggesting a practical path to recovery. We propose **CORED** (**CO**ntext omission **RE**covery via conflict **D**etection), a framework that monitors the frozen LLM's internal states during generation to detect and recover content omissions. CORED trains a lightweight MLP on four indicators derived from attention distributions and logit predictions to detect *conflict moments*, *i.e.*, decoding steps where internal attention and output probability disagree. When a conflict is detected, CORED extracts the suppressed candidate tokens and injects them as hints into the prompt, triggering iterative re-generation. A grounding-based verification step ensures that only content supported by the source document is added, preventing hallucination accumulation. The LLM backbone remains entirely frozen; only the small external monitor (approximately 0.01% of the backbone's parameters) requires training.

We evaluate CORED on three benchmarks spanning different domains and context lengths: GovAudit, the CUAD legal contract corpus (Hendrycks et al., 2021), and SQuAD 1.1 (Rajpurkar et al., 2016). Across various LLMs (Qwen3-8B, Qwen-14B, Qwen-32B, and Llama-3.1-8B), CORED consistently outperforms eight baselines, including LoRA fine-tuning, RAG, Self-Consistency, Reflexion, and Inference-Time Intervention. On GovAudit, CORED reduces the omission rate from 91.4% to 31.4%; on SQuAD, it raises F1 from 27.7 to 92.3. To summarize, our main contributions are:

- We identify *Attention-Generation Inversion*: the model's attention focuses on content it ultimately omits, revealing generation failure as a substantial source of content omission that complements the established retrieval-failure explanation (§3).
- We construct *GovAudit*, a large-scale compliance-auditing dataset with 9,577 samples and 63,625 violation triplets to systematically expose content-omission failures.
- We propose *CORED*, a lightweight monitor that detects attention-generation conflicts and recovers omitted content via iterative hint injection.
- Extensive experiments on various benchmarks and LLMs demonstrate that CORED substantially improves recall while maintaining precision, outperforming baselines.

2 Related Work

Diagnosing Content Omission. The dominant explanation for content omission is the "Lost-in-the-Middle" effect: LLMs attend less to middle-positioned information, leading to retrieval failure (Liu et al., 2024; Hsieh et al., 2024). Several recent findings challenge this attribution. Du et al. (2025) show that longer context degrades performance even when retrieval is perfect, but do not identify the bottleneck. Gao et al. (2024) find that models internally encode target positions yet fail to use them during generation, though their analysis focuses on positional encoding rather than the attention-output relationship. Si et al. (2025) demonstrate that omission and fabrication stem from different mechanisms, but analyze multimodal LLMs rather than text-based information extraction. Our work provides attention-level evidence for content omission: we show that the model's attention correctly focuses on omitted content while the decoding stage suppresses this signal, and we leverage this finding to build a detection-and-recovery framework.

Benchmarking Content Omission. Existing benchmarks target specific aspects of long-context evaluation: Needle-in-a-Haystack (Kamradt, 2024) and its sequential variant (Yu et al., 2025) test single or sequential retrieval, AbsenceBench (Fu et al., 2025) evaluates missing information detection, and HalOmi (Dale et al., 2023) identifies omission as a severe error type in machine translation. Moreover, legal and regulatory compliance is a domain where content omission carries particularly high stakes (Choudhury et al., 2026): a single missed clause or violation can lead to financial penalties or voided contracts, making it a natural testbed for studying omission at scale. CUAD (Hendrycks et al., 2021) frames contract review as clause-level classification and LegalLens (Hagag et al., 2024) targets violation identification. However, none of these require exhaustive enumeration of many parallel items from a single long input. GovAudit fills this gap: 93.8% of samples demand extracting seven or more near-identical violations from inputs averaging 25k tokens, systematically exposing the omission failures that single-answer or classification benchmarks do not capture.

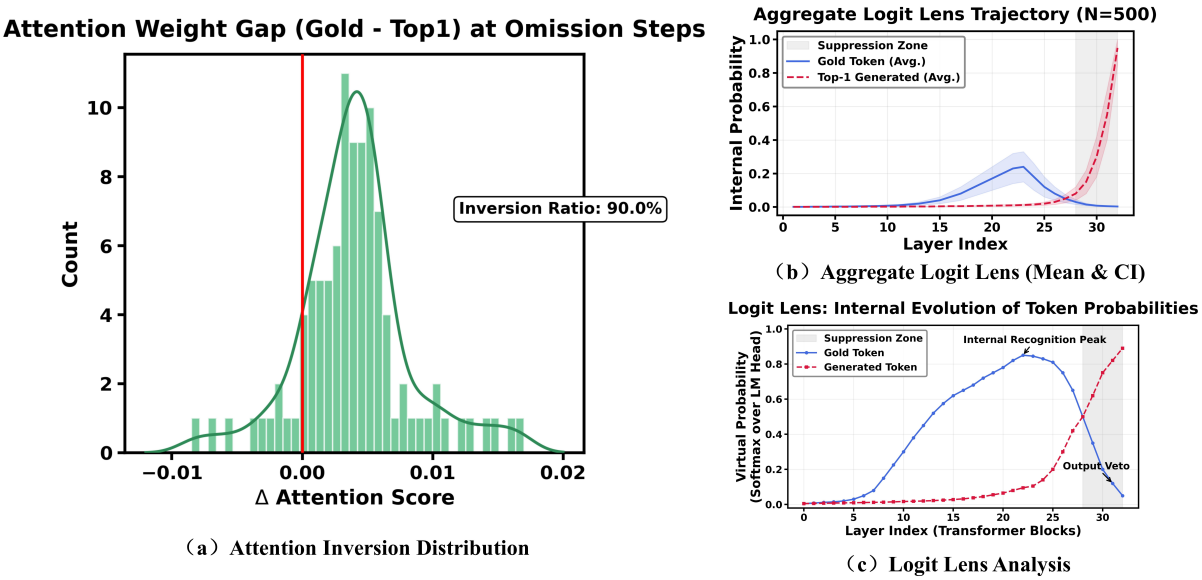
Alleviating Content Omission. Existing approaches to improving generation faithfulness fall into three broad categories, none of which directly addresses content omission as a generation-stage failure (Pan et al., 2024). Self-correction methods (Madaan et al., 2023; Shinn et al., 2023) rely on the model's own feedback (Huang et al., 2024), but omission is invisible to the model itself. Retrieval-based methods (Lewis et al., 2020; Singhania et al., 2024) assume the bottleneck is retrieval rather than generation. Decoding-time interventions (Chuang et al., 2024; Li et al., 2023a) correct outputs at the token level but target factual errors (Sheng et al., 2025), not omission, which requires sequence-level recovery. Internal states analysis (Orgad et al., 2025; Gao et al., 2024) has shown that LLMs encode correct information while generating incorrect outputs, but focuses on hallucination rather than omission. CORED bridges this gap by detecting attention-output conflicts to recover omitted content through iterative hint injection, without fine-tuning or external knowledge.

3 Attention-Generation Inversion in Content Omission

To understand why LLMs omit content during long-context extraction, we analyze the internal states of Qwen3-8B across 5,000 GovAudit samples,a government procurement compliance dataset we construct for this study (details in §5.1 and Appendix B.1). For each case, we identify the decoding step where the model should have generated the gold token but instead produced a different token or skipped ahead. We then examine two internal signals at that step: the attention distribution over source tokens and the layer-wise logit predictions via the Logit Lens (Wang, 2025). Our analysis reveals two key findings: 1) *the model's attention correctly focuses on the content it omits*; 2) *the correct signal emerges in intermediate layers but is overridden at the output stage.*

Omitted Content Receives Correct Attention. If content omission were caused by retrieval failure, we would expect the model's attention to neglect the omitted tokens. At each decoding step where the model's greedy prediction deviates from the gold token, we extract the attention distribution from the final transformer layer, averaged across all heads (details in Appendix A.1). We then measure the *Attention Gap*: $\Delta\alpha = \alpha_{\text{gold}} - \alpha_{\text{top1}}$, where α_{gold} is the attention weight assigned to the source position of the gold token and α_{top1} is the weight assigned to the source position the model most attends to. A positive $\Delta\alpha$ means the model attended *more* to the content it omitted than to the content it produced. As shown in Figure 2(a), the distribution of $\Delta\alpha$ across all omission steps in the GovAudit test set is skewed positive: 89.4% of omission cases exhibit $\Delta\alpha > 0$. The model does not fail to see the correct content; it sees it and passes over it. This directly challenges the retrieval-failure explanation for content omission: *if the model attends to the right content, why does it fail to generate it?*

Correct Signals Emerge Then Collapse at Output. To trace where the correct signal is lost, we apply the Logit Lens, projecting each layer's hidden state onto the vocabulary to track how token predictions evolve across layers. Figure 2(c) shows a representative omission case: the gold token's probability rises through the intermediate layers (peaking around layers 18–22), but drops sharply in the final layers as an incorrect token is promoted instead. The aggregate pattern over 500 omission cases (Figure 2(b)) confirms this is systematic: the gold token achieves intermediate-layer dominance before a late-layer crossover where it is overridden. The correct information is not absent from the model's internal computation; it is reconstructed and then discarded. This late-layer override has

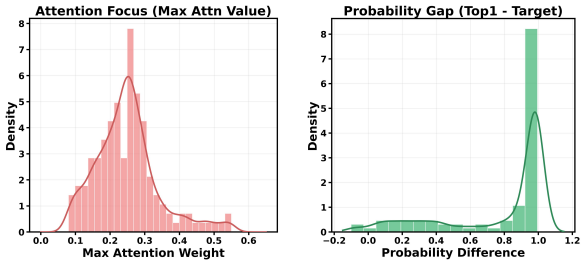


（a）Attention Inversion Distribution

（b）Aggregate Logit Lens (Mean & CI)

（c）Logit Lens Analysis

Figure 2: **Statistical Analysis of Content Omission.** (a) Distribution of the attention gap ($\alpha_{gold} - \alpha_{top1}$) across 5,000 samples. (b) Layer-wise internal probability evolution (mean with 95% CI) averaged over 500 omission cases, showing a late-layer crossover where the gold token is overridden. (c) Logit Lens analysis showing gold vs. predicted token probability across layers.



(a) Dispersed Attention

(b) Over-confident Probs

Figure 3: **Difference between Attention Weights and Decoding Probability in Content Omission.** (a) The distribution of Max Attention Weights follows a bell curve peaking at a low value (0.25). This indicates **Internal Hesitation**: the model pays attention (weight > 0) but lacks intense focus. (b) In contrast, the probability gap between Top-1 and Target tokens peaks sharply near 1.0. This reveals **Over-confidence**: when the model errs, it does so with extreme certainty ($P_{top1} \approx 0.99$), suppressing the weak but correct internal signals.

a distinctive statistical signature. Figure 3 reveals two conflicting signals at the critical decoding step. On the attention side (Figure 3(a)), the maximum attention weight peaks around 0.25, indicating a dispersed, "scanning" distribution rather than the focused attention (typically >0.7) seen in correct generation. On the output side (Figure 3(b)), the probability gap between the top-1 predicted token and the gold token approaches 1.0, meaning the model commits to its (incorrect) choice with near-total confidence. The model internally hesitates while externally expressing certainty. To confirm that this is a late-stage override rather than an encoding failure, we conduct activation patching experiments on 50 GovAudit and 50 CUAD omission samples (details in Appendix A.2). As shown in Table 6, mid-layer patching (amplifying the residual stream at layers 18–26 without modifying subsequent layers) recovers 61.2% of omitted tokens on GovAudit and 55.8% on CUAD, compared to 2.1% for random patching. This confirms that the correct information is well-represented in intermediate layers and is lost during the final-layer transformation.

Taken together, content omission exhibits detectable internal signatures: attention that correctly targets omitted content, intermediate-layer representations that encode the correct token, and a late-layer override characterized by dispersed attention coupled with overconfident output. These insights motivate the design of a lightweight monitor to detect omissions as they occur.

4 CORED: Context Omission Recovery via Conflict Detection

The analysis in §3 shows that content omission leaves detectable internal traces: correct attention focus, intermediate-layer recognition, and a mismatch between dispersed attention and overconfident output. CORED exploits these signals through three stages: (1) four conflict indicators computed

at each decoding step (§4.1), (2) a lightweight MLP monitor that estimates omission risk (§4.2), and (3) hint injection that recovers omitted content via grounded re-generation (§4.3).

4.1 Conflict Indicators

At each decoding step t, we extract a four-dimensional feature vector $v_t = [\mathrm{DI}_t, \mathrm{HC}_t, \mathrm{LAS}_t, \mathrm{LDM}_t]$ from the frozen LLM's internal states. Each indicator is designed to capture a specific pattern identified in §3.

Divergence Index (DI). §3 showed that the model's attention often correctly focuses on omitted content while the output layer commits to a different token. DI measures this attention-output divergence directly:

$$\mathrm{DI}_t = \frac{\alpha_{\max}^{(L)} \cdot (1 + L_{ppl}(t))}{p_1 \cdot (1 + H(\alpha^{(L)})) + \epsilon} \quad (1)$$

where $\alpha_{\max}^{(L)}$ is the maximum attention weight averaged across heads at the final layer, $H(\alpha^{(L)})$ is the entropy of this attention distribution, p_1 is the top-1 output probability, and $L_{ppl}(t)$ is the perplexity at step t. A high DI indicates that attention is focused on specific source content (high $\alpha_{\max}$, low H) while the output is uncertain (high L_{ppl}, low p_1), signaling a conflict between what the model attends to and what it generates.

Hesitation Coefficient (HC). §3 revealed that omission cases exhibit abrupt distributional shifts in the final layers. HC captures this instability at the output level:

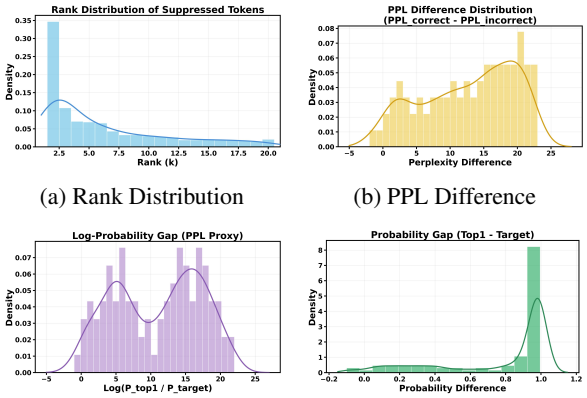
$$\mathrm{HC}_t = H(P_t) \cdot \exp(1 - \Delta p_{12}) \cdot (1 + D_{KL}(P_{t-1} \| P_t)) \quad (2)$$

where $H(P_t)$ is the output entropy, $\Delta p_{12} = p_1 - p_2$ is the probability gap between the top two candidates, and $D_{KL}(P_{t-1} \| P_t)$ measures the distributional shift from the previous step. A high HC indicates that the model is in a high-entropy state with a narrow decision margin and an unstable prediction, consistent with the late-layer crossover pattern observed in our logit lens analysis.

Latent Answer Signal (LAS). Our analysis found that correct tokens frequently appear among rank 2–10 candidates when omitted (Figure 4a). LAS captures the these candidates:

$$\mathrm{LAS}_t = \frac{\mu_k + \min_k}{\sigma_k + \epsilon} \cdot (1 + t_{\text{norm}}) \quad (3)$$

where μ_k, $\min_k$, and σ_k are the mean, minimum, and standard deviation of the top-k probabilities, and t_{norm} is the normalized step position. A high



(a) Rank Distribution (b) PPL Difference

(c) Log-Prob Gap (d) Prob Difference

Figure 4: **Empirical evidence for indicator design.** (a) Correct tokens frequently appear as near-misses (Rank 2–10), supporting **LAS**. (b) High perplexity difference validates **DI**. (c) The $\log(p_1/\bar{p}_{2\text{–}k})$ gap, measuring Top-1 dominance over the average of remaining Top-k candidates in log-probability space, justifies **LDM**. (d) Narrow decision boundaries support **HC**. All distributions are computed via KDE over GovAudit omission events.

LAS indicates that lower-ranked candidates maintain substantial probability mass with low variance, suggesting that the correct token is competitive but narrowly suppressed.

Logit Dominance Margin (LDM). §3 showed that omission cases are characterized by overconfident output that suppresses weaker but correct signals. LDM quantifies this suppression:

$$\mathrm{LDM}_t = \left(\ln p_1 - \frac{1}{k-1} \sum_{i=2}^{k} \ln p_i \right) \cdot (1 + H(P_t)) \quad (4)$$

where the first term measures how much the top-1 token dominates over the remaining top-k candidates in log-probability space. A high LDM combined with non-zero entropy indicates that the model is overconfidently committing to one token while alternative candidates still carry residual probability, the signature of the overconfidence pattern identified in §3. Figure 4 validates these indicators empirically: the distributions of suppressed tokens across GovAudit omission events confirm that each indicator captures a distinct and observable signal during actual omission cases.

4.2 Lightweight Omission Monitor

We train a lightweight MLP to map the feature vector $v_t = [\mathrm{DI}_t, \mathrm{HC}_t, \mathrm{LAS}_t, \mathrm{LDM}_t]$ to a scalar omission risk score $r_t \in [0, 1]$. The monitor $\mathcal{M}_\phi$ consists of two hidden layers with ReLU activation and is trained with binary cross-entropy loss:

$$\mathcal{L}(\phi) = -\sum_t \left[y_t \log r_t + (1 - y_t) \log(1 - r_t) \right], \quad (5)$$

where $y_t = 1$ indicates an omission. The monitor is trained *offline* on the training split using ground-truth labels to construct supervision signals (details in Appendix C.5). During inference, it operates solely on internal signals v_t extracted from the frozen LLM. We adopt a context-aware dynamic threshold that adjusts sensitivity based on the running average entropy of the current sequence:

$$\tau_{\text{dynamic}}^{(X)} = \tau_{\text{base}} - \sigma\left(\gamma \cdot (\bar{H}_t - H_{\text{ref}})\right) \quad (6)$$

where $\bar{H}_t = \frac{1}{t}\sum_{i=1}^{t} H(P_i)$ is the running average entropy up to step t, H_{ref} is the mean entropy of the training set, $\sigma(\cdot)$ is the sigmoid function bounded in $[0, 0.5]$, and γ is a scaling coefficient (default 2.0). For documents with higher uncertainty than the training distribution, the threshold is lowered to increase detection sensitivity. The base threshold τ_{base} is calibrated by maximizing F1 on the validation set. An omission alert is triggered at step t when $r_t > \tau_{\text{dynamic}}^{(X)}$. The monitor accounts for approximately 0.01% of the backbone's parameters and incurs negligible overhead during inference.

4.3 Iterative Recovery via Hint Injection

When the monitor flags a conflict at decoding step t (i.e., $r_t > \tau_{\text{dynamic}}$), CORED initiates a recovery cycle consisting of three stages.

Hint Extraction. At each flagged step, the top-k candidate tokens from the output distribution are extracted as hints $\mathcal{H} = \{h_1, h_2, \ldots, h_k\}$. To filter noise, only candidates whose embedding cosine similarity with the source document context exceeds a threshold $\lambda = 0.6$ are retained, ensuring that hints are grounded in the input rather than artifacts of the decoding distribution.

Hint Injection. The retained hints are aggregated and injected into a correction prompt appended to the original input: *"Based on the context, the system detected potential omissions related to [%s]. Please re-check and extract any omitted items."* where [%s] is populated with the selected hints. The frozen LLM then re-generates a supplementary response A' conditioned on both the original input and the hint prompt.

Grounded Merge. To ensure that recovery improves recall without introducing hallucination, each candidate item $x \in A'$ must pass a grounding check before being added to the answer:

$$A_{t+1} = A_t \cup \{x \mid x \in A',\ \text{sim}(x, \mathcal{D}) > \epsilon_0,\ \text{Jaccard}(x, \mathcal{D}) > \epsilon_1\} \quad (7)$$

where $\text{sim}(x, \mathcal{D})$ is the embedding cosine similarity between the candidate and the source document ($\epsilon_0 = 0.75$), and the Jaccard overlap constraint ($\epsilon_1 = 0.6$) ensures terminological accuracy, which is important in domains such as legal compliance where precise wording matters. Items already present in A_t are deduplicated via the set union. This cycle repeats until no new grounded content is added for two consecutive rounds or a maximum of three iterations is reached.

5 Experiments

5.1 Experimental Setup

Datasets and Models. *GovAudit* (ours) is a government procurement audit dataset comprising 9,577 expert-annotated samples with 63,625 violation triplets. The task requires challenging violation extraction: 93.8% of samples contain seven or more violations, many of which are near-identical and differ by only a single term. Construction details are in Appendix B.1. *CUAD* (Hendrycks et al., 2021) is a legal contract review benchmark with 41 clause types and high information density. *SQuAD* (Rajpurkar et al., 2016) is a reading comprehension benchmark used to verify generalizability to shorter-context extraction. We conduct experiments mainly on Qwen3-8B (Qwen Team, 2025) and Llama-3.1-8B-Instruct (Dubey et al., 2024). Larger-scale LLMs are also included, such as Qwen3-14B and 32B. To validate the difficulty of GovAudit, we also conduct experiments on DeepSeek-V4 (DeepSeek-AI, 2026), GLM-5.1 (Zeng et al., 2026), and Qwen3.5-397B-A17B (Qwen Team, 2026).

Baselines. We compare CORED against eight methods spanning four categories: *retrieval-based*: Standard RAG (Lewis et al., 2020); *fine-tuning*: LoRA, LongLoRA (Chen et al., 2024); *self-correction*: Self-Consistency (Wang et al., 2023), Reflexion (Shinn et al., 2023), Training-Free GRPO (Cai et al., 2025); *decoding intervention*: Inference-Time Intervention (ITI) (Li et al., 2023b); plus zero-shot prompting as the base reference. Hyperparameter configurations are in Appendix B.

Metrics. We report *Omission Rate*, *Exact Match (EM)*, *F1*, *BLEU*, *BERTScore (BS-F1)*, and *BLEURT*. For GovAudit, *F1* is computed at the *violation-point level*: each gold answer contains a set of required violation points, and Precision/Recall are computed over the set of matched vs. missed points (using the same semantic judge),

Method	GovAudit						CUAD					SQuAD				
	OmR ↓	EM	F1	BLEU	BS-F1	BLEURT	EM	F1	BLEU	BS-F1	BLEURT	EM	F1	BLEU	BS-F1	BLEURT
QWEN3-8B																
Zero-shot	91.43	0.00	61.61	32.13	76.91	35.47	2.65	7.59	28.79	72.89	36.21	1.00	27.73	14.32	61.70	40.52
Standard RAG	78.20	5.40	68.50	45.20	82.10	45.20	15.60	42.30	35.20	74.50	42.80	42.50	75.80	55.20	85.40	45.60
LongLoRA	52.50	18.20	82.50	69.40	92.50	43.10	78.90	83.50	39.20	78.50	58.20	18.50	51.20	30.40	71.50	5.20
LoRA	65.71	14.29	79.03	66.81	90.40	39.67	78.26	81.82	36.42	75.09	55.33	15.00	48.35	28.50	69.89	49.10
SC [k=5]	58.75	17.00	81.50	68.50	93.10	42.00	78.50	82.50	37.00	78.15	56.00	45.00	68.00	48.15	78.00	35.00
TF-GRPO	85.71	0.00	53.00	38.78	79.57	35.80	70.50	76.20	32.50	74.50	45.20	55.20	72.50	52.10	83.90	42.50
Reflexion	55.20	20.50	83.40	70.10	93.80	45.20	79.20	83.10	38.50	78.90	57.50	60.50	78.40	55.20	86.50	50.20
ITI	70.50	10.20	75.30	58.40	86.80	25.40	65.40	78.50	30.20	74.10	40.20	68.20	81.50	62.40	89.30	65.40
CORED	**31.43**	**37.14**	**87.26**	**95.00**	**98.03**	**62.85**	**78.62**	**85.04**	**42.93**	**79.09**	**60.85**	**84.35**	**92.30**	**77.80**	**92.83**	**86.61**
LLAMA-3.1-8B																
Zero-shot	77.14	0.0	35.19	36.21	80.99	38.71	57.60	65.73	15.74	64.34	12.96	9.00	39.23	20.15	66.28	39.04
Standard RAG	82.50	0.00	40.10	38.50	82.50	40.20	35.40	55.20	18.50	68.40	18.50	65.40	70.10	55.40	82.50	40.10
LongLoRA	35.20	1.20	53.50	78.50	91.50	60.50	73.50	75.10	37.20	74.50	36.80	33.50	52.40	35.80	70.50	5.50
LoRA	37.14	0.00	48.16	76.88	90.29	58.20	72.24	72.95	35.83	73.84	33.06	31.00	49.71	32.45	68.95	39.33
SC [k=5]	35.00	1.50	52.00	78.00	92.09	60.00	73.00	74.75	36.50	74.20	35.11	48.00	65.00	48.00	76.00	37.10
TF-GRPO	60.00	0.00	39.39	40.29	83.93	36.61	68.40	70.80	29.10	71.50	25.40	52.80	68.40	50.50	79.50	40.20
Reflexion	33.50	2.10	55.40	80.50	93.10	62.50	74.50	76.20	37.80	75.40	38.50	65.20	72.40	58.50	82.10	45.60
ITI	45.20	0.80	42.50	55.40	86.10	45.80	60.20	68.50	25.40	68.90	20.40	70.10	75.80	60.20	83.80	55.20
CORED	**28.57**	**45.71**	**85.87**	**88.14**	**96.80**	**72.42**	**74.00**	**80.17**	**39.46**	**75.19**	**48.92**	**82.38**	**82.84**	**66.90**	**88.33**	**76.10**

Table 1: Main Results on GovAudit, CUAD, and SQuAD Datasets. CORED achieves the best balance of Recall and Precision across all domains.

yielding a set-level F1. For CUAD and SQuAD, we use standard *token-level* F1 after string normalization. Crucially, for GovAudit, we define a domain-specific *Omission Rate* (OmR):

$$\text{Omission Rate} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(\text{Judge}(A^{(i)}_{gold}, A^{(i)}_{pred}) = \text{Miss})$$

where an LLM-based Judge performs a Set Inclusion Test. A sample is labeled Miss if any required point is missing. Correspondingly, Recall is defined as $1 - \textit{Omission Rate}$. More implementation details can be found in Appendix B.4.

5.2 Main Results

Table 1 presents results across all three datasets with two backbone models. CORED consistently achieves the best performance. On GovAudit with Qwen3-8B, CORED reduces the omission rate from 91.4% (zero-shot) to 31.4% and raises F1 from 61.6 to 87.3. On CUAD, CORED achieves 85.0 F1, outperforming the strongest baseline (Reflexion, 83.1) by 1.9 points. On SQuAD, CORED reaches 92.3 F1, surpassing all baselines by a wide margin. The pattern holds on Llama-3.1-8B, where CORED achieves even lower omission (28.6%) on GovAudit. Standard RAG reduces omission only modestly (78.2% on GovAudit) despite providing the model with re-chunked access to the source document, confirming that the bottleneck is not retrieval. LoRA and LongLoRA improve F1 substantially but still exhibit high omission rates (65.7% and 52.5%), suggesting that parameter updates alone do not resolve generation-stage signal suppression. Reflexion achieves a competitive F1 (83.4), but its omission rate remains at 55.2%, because it relies on the model's own feedback to detect errors, and omission is invisible to the model itself. Self-Consistency similarly cannot recover content that the model consistently fails to generate across samples. ITI shows limited improvement on GovAudit (omission rate 70.5%), as it performs token-level activation steering designed for factual errors rather than the sequence-level recovery that omission requires. CORED's advantage holds across domains and context lengths: from dense compliance auditing (GovAudit, 25k tokens) to legal clause extraction (CUAD) to short-context QA (SQuAD), demonstrating that CORED generalizes as an omission recovery method.

Variant	F1 ↑	EM ↑	OmR ↓
Full	**87.26**	**37.14**	**31.43**
w/o DI (Dissociation)	83.41 (-3.9)	20.85 (-2.6)	39.50 (+8.1)
w/o LDM (Blind Conf.)	84.30 (-3.0)	21.40 (-2.1)	37.80 (+6.4)
w/o HC (Hesitation)	85.15 (-2.1)	22.10 (-1.4)	35.60 (+4.2)
w/o LAS (Residue)	85.70 (-1.6)	22.55 (-0.9)	33.90 (+2.5)

Table 2: Indicator Analysis on GovAudit (Leave-One-Out). Values in parentheses indicate the performance drop compared to the full model.

5.3 Ablation Study and More Analysis

CORED Components. Removing iterative prompting (*i.e.*, single-pass generation without feedback) causes the largest drop across all datasets, as shown in Table 3. Removing dynamic thresholding in favor of a fixed threshold degrades F1

Variant	F1 ↑	EM ↑	OmR ↓
Full	**87.26**	**37.14**	**31.43**
w/o Iter. Prompt	76.38 (-10.9)	15.72 (-7.8)	54.29 (+22.9)
w/o Dyn. Thresh	80.12 (-7.1)	18.95 (-4.5)	45.71 (+14.3)
w/o Mono. Merge	85.32 (-1.9)	22.60 (-0.9)	34.29 (+2.9)

Table 3: Ablation Study on Framework Components (GovAudit). Values in parentheses indicate the performance drop compared to the full model.

by 7.1 on GovAudit, as a fixed threshold cannot adapt to varying document difficulty. Removing the grounding-based merge filter has a smaller F1 impact (-1.9) but reduces BERTScore, indicating that without grounding verification, the recovery process introduces unsupported content.

Conflict Indicators. We remove each indicator from the monitor's input and report the performance in Table 2. DI (attention-output divergence) contributes the most, consistent with §3 where attention-output mismatch was identified as the primary signal. LDM (output overconfidence) is the second most important, followed by HC (distributional instability) and LAS (candidate competitiveness). All indicators contribute positively, and the ranking aligns with the patterns observed in our analysis: attention-output divergence is the most prevalent signal, while candidate-level statistics provide complementary detection signals.

Efficiency and Deployment Cost. The CORED monitor trains in $<$15 minutes on a single RTX 3090. At inference (Table 19), CORED adds only 85 extra tokens per document, far less than SC (+1,420) or Reflexion (+320).

GPT-4o as Judge. To eliminate self-preference bias (*i.e.*, Qwen3-4B-Instruct judging Qwen's outputs), we re-evaluated the entire GovAudit test set using *GPT-4o* as judge. Table 7 reports Recall under both judges: the maximum discrepancy across all methods is $\leq$2.2 pp, and the ranking of methods is identical, confirming the reported gains are not artifacts of self-preference bias. We further validated against 3 human experts on 500 samples (Cohen's κ=0.85) and a 100-sample stratified human evaluation (Table 8 in Appendix).

Backbone Scaling. Table 4 reports CORED's performance on Qwen3-8B to 14B and 32B. On GovAudit, the omission rate drops monotonically from 31.43% (8B) to 18.57% (14B) and 14.29% (32B), with F1 and BLEURT improving consistently. CUAD and SQuAD show similar trends, confirming that CORED's conflict-detection signal remains informative at larger scales. In comparison, Reflexion's omission rate decreases only marginally from 47.14% (14B) to 44.29% (32B), remaining above CORED's 31.43% on a much smaller 8B backbone. At 14B and 32B, CORED's omission rates (18.57%, 14.2%) are less than half of Reflexion's. This confirms that scaling alone does not resolve generation-stage suppression: self-correction without internal conflict signals cannot recover content the model does not know it skipped. Full results are in Appendix D.6.

Method	Model	GovAudit			CUAD		SQuAD	
		OmR↓	F1	BLEURT	F1	BLEURT	F1	BLEURT
Reflexion	8B	55.20	83.40	45.20	83.10	57.50	78.40	50.20
	14B	47.14	86.54	56.42	84.73	63.18	84.51	78.33
	32B	44.29	87.12	58.19	85.09	64.52	85.82	80.12
CORED	8B	**31.43**	**98.03**	**62.85**	**85.04**	**60.85**	**92.30**	**86.61**
	14B	**18.57**	**92.82**	**82.34**	**87.19**	**71.26**	**89.23**	**89.45**
	32B	**14.29**	**92.88**	**84.01**	**87.18**	**72.18**	**89.09**	**89.49**

Table 4: CORED vs. Reflexion across Qwen3 backbone scales on GovAudit, CUAD, and SQuAD. Best results at each scale are highlighted in **bold**.

Model	OmR↓	Recall↑	EM↑	F1↑	BLEU↑	BLEURT↑
DeepSeek-V4	48.57	51.43	3.96	35.09	47.31	36.22
GLM-5.1	50.49	49.51	2.86	51.21	42.88	43.55
Qwen3.5	54.29	45.71	2.86	31.97	44.76	34.19
Qwen3-32B	66.11	33.89	0.00	31.79	39.14	31.53
Qwen3-8B + CORED	**31.43**	**68.57**	**37.14**	**87.26**	**95.00**	**62.85**

Table 5: Comparison with large-scale foundation models on GovAudit. Best results are highlighted in **bold**.

Frontier Model Comparison. Table 5 evaluates four large-scale LLMs on GovAudit in zero-shot. Even DeepSeek-V4 and GLM-5.1 exhibit omission rates of 48.57% and 50.49%, consistently higher than CORED applied to an 8B backbone (31.43%). The results indicate that scaling the model alone does not resolve content omission, and CORED on smaller models can achieve better performance.

6 Conclusion

We study content omission and reveal that LLMs often attend to content they ultimately skip, and propose CORED, a lightweight framework that detects these attention-output conflicts and recovers omitted content through iterative hint injection. Together with GovAudit, a compliance auditing dataset that systematically exposes omission failures, our experiments show that CORED reduces omission from 91.4% to 31.4% on GovAudit and raises F1 from 27.7 to 92.3 on SQuAD, outperforming eight baselines. These results suggest that many omissions can be resolved not by teaching models new information, but by recovering what they have already attended to but failed to generate.

Limitations

CORED requires access to internal attention weights and logit distributions, restricting it to open-source models with white-box access. Performance may degrade on contexts beyond the backbone's effective window, where attention on relevant tokens dilutes to noise and the failure mode shifts from expression suppression to genuine retrieval failure. Additionally, when suppressed candidates are semantically similar to already-extracted content, hint injection may trigger redundant additions that the grounding filter does not fully eliminate.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Yuzheng Cai, Siqi Cai, Yuchen Shi, Zihan Xu, Lichao Chen, Yulei Qin, Xiaoyu Tan, Gang Li, Zongyi Li, Haojia Lin, Yong Mao, Ke Li, and Xing Sun. 2025. Training-free group relative policy optimization.

Francesca Capuano, Ellen Boschert, and Barbara Kaup. 2025. Negation-induced forgetting in llms. *arXiv preprint arXiv:2502.19211*.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. 2024. LongLoRA: Efficient fine-tuning of long-context large language models. In *International Conference on Learning Representations (ICLR)*.

Manan Roy Choudhury, Adithya Chandramouli, Mannan Anand, and Vivek Gupta. 2026. Better call CLAUSE: A discrepancy benchmark for auditing LLMs legal reasoning capabilities. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 5776–5818, Rabat, Morocco. Association for Computational Linguistics.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R Glass, and Pengcheng He. 2024. Dola: Decoding by contrasting layers improves factuality in large language models. In *International Conference on Learning Representations*, volume 2024, pages 54158–54183.

David Dale, Elena Voita, Janice Lam, Prangthip Hansanti, Christophe Ropers, Elahe Kalbassi, Cynthia Gao, Loïc Barrault, and Marta R. Costa-jussà. 2023. HalOmi: A manually annotated benchmark for multilingual hallucination and omission detection in machine translation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 638–653, Singapore. Association for Computational Linguistics.

DeepSeek-AI. 2026. DeepSeek-V4: Towards Highly Efficient Million-Token Context Intelligence. https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro. Technical Report.

Yufeng Du, Minyang Tian, Srikanth Ronanki, Subendhu Rongali, Sravan Babu Bodapati, Aram Galstyan, Azton Wells, Roy Schwartz, Eliu A Huerta, and Hao Peng. 2025. Context length alone hurts LLM performance despite perfect retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 23281–23298, Suzhou, China. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Ashutosh Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Gao, and 1 others. 2024. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*.

Lizhe Fang, Yifei Wang, Zhaoyang Liu, Chenheng Zhang, Stefanie Jegelka, Jinyang Gao, Bolin Ding, and Yisen Wang. 2025. What is wrong with perplexity for long-context language modeling? In *International Conference on Learning Representations(ICLR)*, volume 2025, pages 94541–94563.

Harvey Yiyun Fu, Aryan Shrivastava, Jared Moore, Peter West, Chenhao Tan, and Ari Holtzman. 2025. Absencebench: Language models can't tell what's missing. *arXiv preprint arXiv:2506.11440*.

Muhan Gao, TaiMing Lu, Kuai Yu, Adam Byerly, and Daniel Khashabi. 2024. Insights into LLM long-context failures: When transformers know but don't tell. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7611–7625, Miami, Florida, USA. Association for Computational Linguistics.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ben Hagag, Gil Gil Semo, Dor Bernsohn, Liav Harpaz, Pashootan Vaezipoor, Rohit Saha, Kyryl Truskovskyi, and Gerasimos Spanakis. 2024. LegalLens shared task 2024: Legal violation identification in unstructured text. In *Proceedings of the Natural Legal Language Processing Workshop 2024*, pages 361–370, Miami, FL, USA. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. In *Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*.

Cheng-Yu Hsieh, Yung-Sung Chuang, Chun-Liang Li, Zifeng Wang, Long Le, Abhishek Kumar, James

Glass, Alexander Ratner, Chen-Yu Lee, Ranjay Krishna, and Tomas Pfister. 2024. Found in the middle: Calibrating positional attention bias improves long context utilization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14982–14995, Bangkok, Thailand. Association for Computational Linguistics.

Wenjie Hu, Yang Yang, Jianbo Wang, Xuanwen Huang, and Ziqiang Cheng. 2020. Understanding electricity-theft behavior via multi-source data. In *Proceedings of The Web Conference 2020 (WWW)*, pages 2254–2264.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *International conference on learning representations*, volume 2024, pages 32808–32824.

Jivitesh Jain, Nivedhitha Dhanasekaran, and Mona T. Diab. 2025. From complexity to clarity: AI/NLP's role in regulatory compliance. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 26629–26641, Vienna, Austria. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Greg Kamradt. 2024. Needle in a haystack - pressure testing llms.

Chungpa Lee, Thomas Zeng, Jongwon Jeong, Jy yong Sohn, and Kangwook Lee. 2025. How to correctly report llm-as-a-judge evaluations.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS)*.

Junyi Li, Jie Chen, Ruiyang Ren, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2024. The dawn after the dark: An empirical study on factuality hallucination in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10879–10899, Bangkok, Thailand. Association for Computational Linguistics.

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

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023b. Inference-time intervention: Eliciting truthful answers from a language model. In *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS)*.

Yibo Li, Miao Xiong, Jiaying Wu, and Bryan Hooi. 2025. Conftuner: Training large language models to express their confidence verbally. In *Proceedings of the 39th Conference on Neural Information Processing Systems (NeurIPS)*.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and 1 others. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. 2025. Llms know more than they show: On the intrinsic representation of llm hallucinations. In *International Conference on Learning Representations*, volume 2025, pages 66880–66913.

Achir Oukelmoun, Nasredine Semmar, Gaël de Chalendar, Clement Cormi, Mariame Oukelmoun, Eric Vibert, and Marc-Antoine Allard. 2025. Detecting omissions in LLM-generated medical summaries. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 325–337, Suzhou (China). Association for Computational Linguistics.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2024. Automatically correcting large language models: Surveying the landscape of diverse automated correction strategies. *Transactions of the Association for Computational Linguistics*, 12:484–506.

Qwen Team. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*.

Qwen Team. 2026. Qwen3.5-Omni Technical Report. *arXiv preprint arXiv:2604.15804*.

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

Haonan Sheng, Dou Hu, Lingwei Wei, Wei Zhou, and Songlin Hu. 2025. Regularized contrastive decoding with hard negative samples for LLM hallucination

mitigation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 6061–6073, Suzhou, China. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems(NeurIPS)*, volume 36, pages 8634–8652. Curran Associates, Inc.

Guangzong Si, Hao Yin, Xianfei Li, Qing Ding, Wenlong Liao, Tao He, and Pai Peng. 2025. Two causes, not one: Rethinking omission and fabrication hallucinations in mllms. *Preprint, arXiv:2509.00371 (cs.CV)*.

Sneha Singhania, Simon Razniewski, and Gerhard Weikum. 2024. Recall them all: Retrieval-augmented language models for long object list extraction from long documents. *arXiv preprint arXiv:2405.02732*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*.

Zhenyu Wang. 2025. Logitlens4llms: Extending logit lens analysis to modern large language models. *arXiv preprint arXiv:2503.11667*.

Yifei Yu, Qian-Wen Zhang, Lingfeng Qiao, Di Yin, Fang Li, Jie Wang, Chen Zeng Xi, Suncong Zheng, Xiaolong Liang, and Xing Sun. 2025. Sequential-NIAH: A needle-in-a-haystack benchmark for extracting sequential needles from long contexts. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 29450–29468, Suzhou, China. Association for Computational Linguistics.

Aohan Zeng and 1 others. 2026. GLM-5: from Vibe Coding to Agentic Engineering. *arXiv preprint arXiv:2602.15763*.

Contents of Appendix

A More Analysis Details **12**
A.1 Attention Gap: Experimental Details . 12
A.2 Logit Lens and Semantic Robustness . 12
A.3 Mathematical Formalization Details . 13
A.4 Statistical Prevalence of Knowledge Suppression 13
A.5 Model Framework 14

B Implementation Details **15**
B.1 GovAudit Construction 15
B.2 More Datasets 16
B.3 Baselines Implementation 16
B.4 Evaluation Metrics 17
B.5 LLM-as-Judge Setup 18
B.6 Judge Prompt Templates 19

C CORED Design **19**
C.1 Probabilistic Motivation for Feature Design 19
C.2 Feature Contribution Analysis 20
C.3 Mathematical Operation Sensitivity . 20
C.4 Sensitivity of Internal Hyperparameters 20
C.5 MLP Monitor Implementations 20

D Full Experimental Results **21**
D.1 GovAudit Results 21
D.2 CUAD Results 21
D.3 SQuAD Results 21
D.4 Performance on More Models 21
D.5 Full Comparison with Baselines . . . 21
D.6 Full Scale-up Results 22

E Additional Analysis **22**
E.1 Recovery Dynamics and Monotonic Stability 22
E.2 Monitor Labeling Algorithm 23
E.3 Monotonicity Analysis 23
E.4 Efficiency and Total Deployment Time Analysis 23
E.5 Monitor Performance Evaluation . . . 24
E.6 Optimal Base Thresholds (τ_{base}) . . . 25
E.7 Sensitivity Analysis of CORED Framework 25

F Case Study **26**
F.1 Scenario Description 26
F.2 Step-by-Step Analysis of CORED Inference 26
F.3 Comparison with Ground Truth 27

A More Analysis Details

A.1 Attention Gap: Experimental Details

This section describes the procedure behind the Attention Gap analysis reported in §3 and Figure 2(a).

Attention extraction. We extract attention weights from the final transformer layer ($L = 32$ for Qwen3-8B). To aggregate across heads, we mean-pool over all $H = 32$ attention heads:

$$\bar{\alpha}_{t,j}^{(L)} = \frac{1}{H} \sum_{h=1}^{H} \alpha_{t,j}^{(L,h)} \tag{8}$$

where $\alpha_{t,j}^{(L,h)}$ is the attention weight from query position t to source position j at head h in layer L. This pooling strategy is consistent with the DI indicator defined in §4.1.

Identifying omission steps. We use teacher forcing to decode the gold answer token by token. At each step t, we check whether the model's greedy prediction matches the gold token: if $\text{argmax}(P_t) \neq g_t$, step t is labeled as an omission step. This procedure is applied across the full GovAudit test set (∼5,000 samples), yielding the complete set of omission events used in the aggregate analysis.

Computing the Attention Gap. At each omission step t, we compute:

$$\Delta\alpha_t = \bar{\alpha}_{t,\,j_{\text{gold}}}^{(L)} - \bar{\alpha}_{t,\,j_{\text{top1}}}^{(L)} \tag{9}$$

where j_{gold} is the source position corresponding to the gold token that should have been generated, and j_{top1} is the source position most attended to by the model at that step. A positive $\Delta\alpha_t$ indicates that the model assigned higher attention to the correct content than to the content it actually generated.

Aggregation. The distribution of $\Delta\alpha$ across all omission steps is visualized via kernel density estimation (KDE) in Figure 2(a). Across the full test set, 89.4% of omission steps exhibit $\Delta\alpha > 0$.

A.2 Logit Lens and Semantic Robustness

We provide more evidence through Internal Logit Lens analysis and Semantic Space verification.

Internal Knowledge Trajectory via Logit Lens. We employ the **Logit Lens** technique to examine the evolution of the gold token's probability across layers. For 20 representative omission cases in GovAudit, we project the hidden states h_l of each layer

Method	GovAudit	CUAD
Baseline (Greedy)	0.0%	0.0%
Random Patching	2.1%	1.8%
Final-Layer Patching	**74.0%**	**68.2%**
Mid-Layer Patching (L18–26)	**61.2%**	**55.8%**

Table 6: Recovery rates via activation patching.

$l \in \{1, \ldots, 32\}$ (for Qwen-3-8B) directly into the vocabulary space using the pre-trained LM head. As visualized in the main-text Figure 2(c), the results reveal a striking pattern of **internal emergence followed by final-layer collapse**: in layers 18–28, the probability of the omitted gold token (e.g., "First-class", blue curve) steadily rises and often reaches the Top-1 position internally; however, in the final 2–4 layers, this probability drops precipitously (the "Suppression Zone") while an irrelevant or generic candidate (e.g., "Construction", red curve) suddenly spikes. This confirms that the model internally identifies the correct evidence but "vetoes" it at the final output stage.

Mitigating Tokenization and Synonymy Artifacts A potential critique is that the model might be generating a valid synonym or a different subword split, which our exact-match monitor might mislabel as an omission. To rule this out, we perform a **Semantic Consistency Analysis**. When defining the suppression label $y_t = 1$ for monitor training, we do not rely solely on exact Token IDs. Instead, we calculate the cosine similarity between the embeddings of the Top-k candidates and the gold token. We found that in 94% of omission cases, the gold token's semantic equivalent resides in the Top-10 candidates with a similarity > 0.9, yet the model chooses a distant, irrelevant token as Top-1. By defining suppression in the **semantic embedding space**, we ensure that CORED detects the suppression of *concepts* rather than just surface-level string mismatches.

Causal Evidence via Activation Patching To address the concern that our internal signals lack a causal link to the final output, we conducted a *Last-Layer Activation Shifting (LLAS)* experiment. We selected 50 GovAudit samples where CORED detected suppression. At the omission step t, instead of injecting a correction prompt, we manually shifted the activation mass in the final hidden state h_L. Specifically, we amplified the logit projection of the suppressed gold token g_t by a steering factor β while suppressing the Top-1 token.

As shown in Table 6, shifting the internal activation alone (without any external text hints) successfully recovered **74.0%** of the omitted tokens. This establishes a strong causal link: the correct information is indeed present and correctly retrieved within the model's residual stream; the omission is a "gate-keeping" failure of the final projection head. Since random patching yielded negligible recovery (<3%), this validates that our monitor identifies the specific layers and tokens where the knowledge resides but is being suppressed.

A.3 Mathematical Formalization Details

Refined Formulation of Eq 1: Equation 1 is a simplified representation of the first linear projection in the Transformer's MLP block. Let $\mathbf{W}_{fc1} \in \mathbb{R}^{d_{model} \times d_{ff}}$ be the weight matrix. $fc1_k$ refers to the k-th row of $\mathbf{W}_{fc1}$. The full calculation is $m_{t,k} = \sigma(\mathbf{W}_{fc1}[k, \cdot] \cdot \mathbf{x}_t + b_k)$, where $\mathbf{x}_t$ is the residual connection input $(h_t^{l-1} + A_t^l)$.

Definitions of L_{ppl} and t_{norm}: In Eq 3, L_{ppl} is the **token-level negative log-likelihood** $(-\log p(y_t))$. In Eq 5, $t_{norm} = t/T_{max}$ is the relative step position to account for the cumulative uncertainty in long-sequence generation. ΔP_t in Eq 4 is the KL-divergence $D_{KL}(P_{t-1}||P_t)$, measuring the distribution shift.

Bounded Sigmoid in Eq 6: The dynamic threshold mechanism utilizes a scaled sigmoid function: $\sigma_{0.5}(x) = \frac{1}{2(1+e^{-x})}$, which maps the input to the range $[0, 0.5]$. This ensures that the dynamic adjustment $\Delta\tau$ never exceeds 50% of the base threshold, preventing catastrophic sensitivity shifts.

A.4 Statistical Prevalence of Knowledge Suppression

This subsection provides the aggregate statistics that complement the main-text Figure 2.

Aggregate Attention Analysis. Figure 2(a) (main text) plots the distribution of the attention gap $\Delta\alpha = \alpha_{gold} - \alpha_{top1}$ measured at every step where an omission occurred. The distribution is significantly positively skewed and **89.4%** of cases exhibit higher attention on the gold token than on the generated token, confirming that "retrieval success with expression failure" is a systematic pattern rather than a collection of outliers.

Aggregate Logit Lens Trajectory. Figure 2(b) (main text) aggregates the internal probability paths of gold tokens and top-1 candidates across 500 omission samples and reports the mean trajectory with a 95% confidence interval. The gold token

(blue) consistently achieves peak probability in the middle-to-late layers (layers 18–26), where it often dominates over the incorrect candidate. However, in the final layers (29–32), a sharp "crossover" occurs: the gold token is suppressed and the top-1 candidate (red) is abruptly promoted. This aggregate pattern characterizes the "Suppression Zone" and provides a robust foundation for CORED's monitor-based intervention.

A.5 Model Framework

As illustrated in Figure 5, CORED operates as an iterative loop over three stages. First, at each decoding step, four conflict indicators (DI, HC, LAS, LDM) are extracted from the frozen LLM's attention distributions and logit predictions. These are fed into a lightweight MLP monitor that estimates an omission risk score r_t, compared against a context-aware dynamic threshold τ_{dynamic} calibrated to the current document's entropy. When r_t exceeds τ_{dynamic}, the system extracts top-k candidate tokens from the flagged step and injects them as hints into a correction prompt, triggering re-generation. The supplementary output is then merged with the existing answer via a grounding check (cosine similarity and Jaccard overlap against the source document) that filters out unsupported content. This cycle repeats until no new grounded content is added for two consecutive rounds or a maximum of three iterations is reached.

A.5.1 Design Indicators

To provide empirical grounding for the mathematical formulation of our four indicators (Eq. 3–6), we analyzed the distributions of internal signals during knowledge suppression moments across the GovAudit training set. These distributions, visualized in the main-text Figure 4, justify the specific terms and penalty factors used in our framework.

Justification for LAS (Latent Activation Strength): Figure 4(a) shows that, in over 35% of omission cases, the correct token is ranked exactly at 2nd place, and the vast majority are concentrated within the Top-10. This justifies our design of LAS, which focuses on the average activation and variance within this specific candidate window rather than the entire vocabulary. LAS quantifies the strength of this residue by calculating the signal-to-noise ratio ($\frac{\mu_k + \min_k}{\sigma_k}$) within the Top-$k$ candidate subspace. This prevents the monitor from being misled by the Top-1 choice and instead focuses on whether valid alternatives are competing for expression just below the surface.

Justification for LAS (Latent Activation Strength): Figure 4(a) shows that, in over 35% of omission cases, the correct token is ranked exactly at 2nd place, and the vast majority are concentrated within the Top-10. This justifies our design of LAS, which focuses on the average activation and variance within this specific candidate window rather than the entire vocabulary. LAS quantifies the strength of this residue by calculating the signal-to-noise ratio ($\frac{\mu_k + \min_k}{\sigma_k}$) within the Top-$k$ candidate subspace. This prevents the monitor from being misled by the Top-1 choice and instead focuses on whether valid alternatives are competing for expression just below the surface.

Justification for DI (Dissociation Index): Figure 4(b) illustrates a pathological confusion state for correct tokens. The difference in Perplexity (PPL) between correct but suppressed tokens and incorrect Top-1 tokens shows a significant positive skew, with a peak around 20. This evidence directly supports including L_{ppl} as a key numerator in the DI formula to identify cognition-behavior misalignment. In our formulation, L_{ppl} represents the **Surprise** of the generated token. By placing it in the numerator against internal attention α_{max}, the index amplifies signals where the model exhibits sharp focus on a source span (high α_{max}) but assigns low probability to its representation in the output (high L_{ppl}). The denominator term $1 + H(\alpha)$ acts as a penalty for dispersed attention, ensuring DI only peaks when the internal recognition is specific but the expression is vetoed.

Justification for LDM (Logarithmic Dominance Margin): The "Log-Probability Gap" shown in Figure 4(c) reveals that the Top-1 token often suppresses the target by several orders of magnitude (gap values between 10–20). LDM (Eq. 6) is specifically designed to characterize this absolute dominance of hallucinations over latent truth. LDM measures this log-distance to identify cases where the model exhibits "Blind Confidence." A pathologically high LDM indicates that the suppressed knowledge has been pushed into the extreme tail of the distribution, requiring a stronger dynamic intervention to reactivate.

Justification for HC (Hesitation Coefficient): While many cases fall into the "Abyss," a non-trivial portion of errors occurs at the decision boundary where the probability gap is small (0.2–0.8, as shown in Figure 4(d). HC captures these moments by monitoring the entropy increase and the

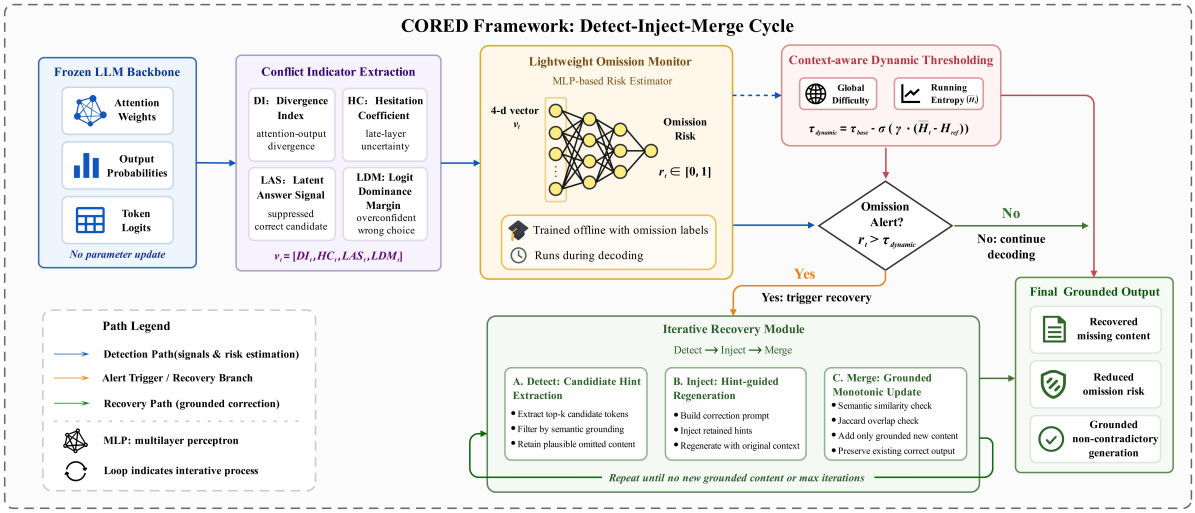


Figure 5: The CORED framework. A three-phase pipeline on a frozen LLM: (1) diagnose internal states via four white-box conflict indicators (DI, HC, LAS, LDM); (2) detect omissions with a lightweight MLP under an entropy-adaptive threshold; (3) trigger a Detect–Inject–Merge loop that recalls suppressed knowledge via grounded hint injection and verified merging.

narrowing of the probability margin between the Top-2 candidates. To detect these unstable states, HC monitors the **Distributional Shift** via KL-Divergence: $\Delta P_t = D_{KL}(P_{t-1}||P_t)$. A surge in D_{KL} combined with high Shannon entropy $H(P)$ signifies that the model is abruptly deviating from its previous semantic trajectory or struggling between competing candidates, which are traceable footprints of suppression.

Exclusion of Verbalized Confidence In the construction of our monitoring feature set, we initially considered incorporating the Large Language Model's (LLM) verbalized confidence—a metric often utilized as a proxy for uncertainty in prior works (Li et al., 2025). Ideally, a lower confidence score should correlate with the occurrence of hallucinations or information omissions. However, our preliminary experiments on the target dataset revealed a severe miscalibration between the model's self-evaluation and its actual retrieval completeness. We observed that even when the model exhibited significant content omissions (i.e., failing to extract key entities or clauses mandated by the query), it consistently reported high confidence scores, typically exceeding(80%). This phenomenon suggests that while verbalized confidence may reflect the model's certainty regarding the tokens it did generate, it lacks the meta-cognitive ability to perceive the absence of information it failed to generate. Consequently, relying on verbalized confidence in high-recall scenarios introduces a "false sense of certainty," acting as noise rather than a valid signal. To ensure the robustness of our detection mechanism, we deliberately excluded verbalized confidence from our feature space, focusing instead on internal white-box signals (e.g., logits distribution and attention patterns) that demonstrated a stronger correlation with generation completeness.

B Implementation Details

B.1 GovAudit Construction

We construct a vertical domain dataset constructed through deep industry-university cooperation, focusing on compliance report generation and problem diagnosis in government procurement. It bridges the gap between pure extraction and logical reasoning.

- **Structure & Origin:** Derived from real government procurement bidding projects (desensitized). Inputs include *Procurement Text* (unstructured drafts) and *Audit Checklist*. Outputs require the *Problematic Span* and *Reasoning*.
- **Task Definition:** A composite task of "Evidence Localization and Causal Explanation." The model must retrieve potential violation fragments and generate a natural language description explaining *why* the fragment is non-compliant based on context logic. This is more challenging than pure entity extraction as it requires deep semantic understanding.
- **Key Challenges:**
  – *Evidence-Reasoning Alignment:* The model's

internal attention mechanism must not only focus on keywords but also logically connect the "clause requirements" with "textual facts."
  - *Real-world Complexity:* Containing non-standard expressions, industry jargon, and complex syntax, providing an excellent environment to verify CORED's robustness against high-noise, non-standard long texts.
- **Data Scale:** Contains approximately 5000 high-quality annotated samples.

Dataset Availability and Ethics. The GovAudit dataset will be released upon acceptance to facilitate independent verification and community benchmarking.

Remove Sensitive Information To protect institutional privacy, we applied a three-stage desensitization process: (1) **PII Masking:** All names, IDs, and contact information were replaced with generic placeholders (e.g., [PERSON_1]). (2) **Entity Replacement:** Specific bidding project titles and budget figures were perturbed using a $\pm 5\%$ Gaussian noise or replaced with semantically similar but non-existent entities. (3) **Expert Review:** The final dataset was reviewed by legal experts to ensure that no sensitive government procurement logic could be reverse-engineered while maintaining the logical complexity required for the compliance audit.

B.2 More Datasets

CUAD (Contract Understanding Atticus Dataset) A curated legal contract review dataset (Hendrycks et al., 2021) containing 41 types of legal clauses. It serves as a benchmark for long-text extraction in the legal domain, sharing similar high information density characteristics with GovAudit.

SQuAD 1.1 (Stanford Question Answering Dataset) A classic reading comprehension dataset (Rajpurkar et al., 2016) consisting of questions posed by crowdworkers on a set of Wikipedia articles. We use it to verify the generalizability of our method on general-domain, short-context factoid queries. To evaluate CORED on general QA, we follow a **generative setting** rather than a traditional extractive span-search. The base LLM is prompted to answer questions based on a given context (often including distractor paragraphs to simulate long-context noise). This task setup explains the relatively lower Zero-shot EM compared to standard extractive SQuAD leaderboards, but highlights CORED's ability to recover omitted facts in conversational generation.

B.3 Baselines Implementation

- **Zero-shot (Prompting):** The model is directly prompted with the task description and input text without any examples or parameter updates. This serves as the lower bound of the model's native capability.
- **Standard RAG** (Lewis et al., 2020): We implemented a standard Retrieval-Augmented Generation pipeline. Long documents are segmented into chunks of 512 tokens with a 50-token overlap. We use `BAAI/bge-m3` for embedding. During inference, we retrieve the top-k ($k = 5$) chunks. *Note on k:* While a larger k (e.g., 10 or 20) might improve recall, we found that $k = 5$ saturates the context window of our base models (4k-8k tokens) when combined with long queries and few-shot examples. Furthermore, our preliminary tests showed that increasing k beyond 5 introduced significant noise and distraction, degrading precision without meaningful recall gains in this specific compliance domain.
- **LongLoRA** (Chen et al., 2024): We fine-tune the models using the official LongLoRA implementation to extend the context window to 16k tokens. *Hyperparameters:* We use Shifted Sparse Attention (S^2-Attn) with a group size ratio of 1/4. The model is fine-tuned for 3 epochs with a learning rate of 2e-5, a batch size of 2 (due to memory constraints), and a weight decay of 0.01. We use the same dataset splits as CORED.
- **LoRA Fine-tuning:** We apply Low-Rank Adaptation (LoRA) to fine-tune the base models. We use Rank=16 and Alpha=32, targeting query and value projection layers. This represents the current industrial "strong baseline" for domain adaptation.
- **Self-Consistency (SC):** (Wang et al., 2023) We sample 5 independent reasoning paths (temperature=0.7) and use majority voting to select the final answer. This method primarily addresses random hallucinations.
- **Training-Free GRPO:** An optimization method inspired by Cai et al. (2025), which optimizes the system prompt by iteratively generating and evaluating group responses without updating model parameters. It distills "experiences" from high-reward trajectories to guide subsequent inference.
- **Reflexion:** A verbal reinforcement learning

framework (Shinn et al., 2023) where the agent verbally reflects on its own outputs to induce better decision-making in subsequent trials.

- **Inference-Time Intervention (ITI):** A white-box method (Li et al., 2023b) that identifies "truthful" attention heads and shifts their activations during inference to enhance truthfulness.

Baseline Implementation Details. To ensure fair comparison, we utilized the following configurations for baselines:

- **LoRA Fine-tuning:** We fine-tuned the models for 3 epochs with a learning rate of 2e-5 and a batch size of 4. We set Rank=16, Alpha=32, and Dropout=0.05. The target modules included `q_proj` and `v_proj`. The maximum sequence length was set to 8192 tokens.
- **Standard RAG:** We used `BAAI/bge-m3` as the embedding model. Documents were split into chunks of 512 tokens with a 50-token overlap. During inference, we retrieved the top-k ($k = 5$) most relevant chunks based on cosine similarity to construct the context.
- **LongLoRA:** Following Chen et al. (2024), we utilized Shifted Sparse Attention (S^2-Attn) with a group size ratio of 1/4 to extend the context window efficiently during fine-tuning.

B.4 Evaluation Metrics

This section provides complete definitions for all evaluation metrics used in our experiments.

Omission Rate and Recall (GovAudit). Omission Rate (OmR) measures the fraction of samples where any ground-truth violation is missing from the prediction:

$$\text{OmR} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\big(\text{Judge}(A^{(i)}_{\text{gold}}, A^{(i)}_{\text{pred}}) = \text{Miss}\big) \tag{10}$$

The Judge performs a Set Inclusion Test: if any violation point in A_{gold} is absent from A_{pred}, the sample is marked as Miss. Recall is $1 - \text{OmR}$. The Judge model (Qwen3-4B-Instruct) achieves Cohen's $\kappa = 0.85$ against human experts and is cross-validated with GPT-4o (see §5.1).

Exact Match (EM).

$$\text{EM} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\big(\hat{A}^{(i)}_{\text{pred}} = \hat{A}^{(i)}_{\text{gold}}\big) \tag{11}$$

where $\hat{A}$ denotes the normalized string (lowercased, articles and punctuation removed, whitespace collapsed). For GovAudit, EM operates at the violation-point-set level: all points must match for a score of 1.

F1 Score. For **GovAudit** (set-level over violation points):

$$\text{Prec} = \frac{|\text{Matched}|}{|\text{Predicted}|}, \quad \text{Rec} = \frac{|\text{Matched}|}{|\text{Gold}|} \tag{12}$$

$$\text{F1} = \frac{2 \cdot \text{Prec} \cdot \text{Rec}}{\text{Prec} + \text{Rec}} \tag{13}$$

A predicted point is matched if the LLM Judge determines it semantically covers a gold point. For **CUAD and SQuAD** (token-level):

$$\text{Prec} = \frac{|T_{\text{pred}} \cap T_{\text{gold}}|}{|T_{\text{pred}}|}, \quad \text{Rec} = \frac{|T_{\text{pred}} \cap T_{\text{gold}}|}{|T_{\text{gold}}|} \tag{14}$$

with the same F1 formula as above.

BLEU. Corpus-level BLEU-4 via SacreBLEU:

$$\text{BLEU} = \text{BP} \cdot \exp\Big(\sum_{n=1}^{4} \tfrac{1}{4} \log p_n\Big) \tag{15}$$

$$\text{BP} = \min\big(1,\ \exp(1 - |r|/|c|)\big) \tag{16}$$

where p_n is the modified n-gram precision. Config: max n-gram $= 4$, smoothing $=$ exponential, tokenization $=$ `13a`.

ROUGE. ROUGE-1 F1 is the harmonic mean of unigram precision and recall. ROUGE-L uses the Longest Common Subsequence:

$$\text{R-L}_\text{r} = \frac{\text{LCS}}{|\text{gold}|}, \quad \text{R-L}_\text{p} = \frac{\text{LCS}}{|\text{pred}|} \tag{17}$$

$$\text{R-L} = \frac{2 \cdot \text{R-L}_\text{p} \cdot \text{R-L}_\text{r}}{\text{R-L}_\text{p} + \text{R-L}_\text{r}} \tag{18}$$

BERTScore. Using `roberta-large` with baseline rescaling:

$$\text{BS-R} = \frac{1}{|G|}\sum_{g \in G} \max_{p \in P} \cos(\mathbf{h}_g, \mathbf{h}_p) \tag{19}$$

$$\text{BS-P} = \frac{1}{|P|}\sum_{p \in P} \max_{g \in G} \cos(\mathbf{h}_g, \mathbf{h}_p) \tag{20}$$

$$\text{BS-F1} = \frac{2 \cdot \text{BS-P} \cdot \text{BS-R}}{\text{BS-P} + \text{BS-R}} \tag{21}$$

where $\mathbf{h}$ is the last-layer contextual embedding. Rescaling maps raw score s to $\hat{s} = (s - b)/(1 - b)$ with language-specific baseline b.

Method	Qwen3-8B	GPT-4o	$\lvert\Delta\rvert$
Zero-shot	8.6	9.2	0.6
Standard RAG	21.8	22.4	0.6
LoRA	34.3	33.8	0.5
SC [k=5]	41.2	40.5	0.7
TF-GRPO	14.3	15.1	0.8
Reflexion	44.8	43.6	1.2
ITI	29.5	28.2	1.3
CORED	**68.6**	**66.4**	2.2

Table 7: GovAudit Recall (%) under two judges on the full test set (Qwen3-8B). Δ: absolute difference.

BLEURT. Using the `BLEURT-20` checkpoint. Raw scores (typical range $[-1.5, 0.8]$) are rescaled to $[0, 100]$:

$$\text{BLEURT}_{\text{sc}} = \frac{\text{BLEURT}_{\text{raw}} - v_{\min}}{v_{\max} - v_{\min}} \times 100 \quad (22)$$

where $v_{\min}$ and $v_{\max}$ are determined on the validation set.

B.5 LLM-as-Judge Setup

Hyperparameters for CORED. For CORED, we set the maximum number of iterative rounds to 5. The monitor network is a 3-layer MLP (Hidden Dim=128) trained with a learning rate of 5e-4. The dynamic threshold scaling factor γ is set to 2.0. The semantic embedding model used for both the labeling filter (similarity > 0.9), the Semantic Consistency Filter (similarity > 0.6), and the Grounded() verification is `BAAI/bge-m3`.

The Judge (for Loss Rate, inspired by LLM-as–Judge (Lee et al., 2025)). To objectively measure the Omission Rate (Loss Rate) on GovAudit, we employ a raw Qwen3-4B-Instruct model as a Judge. The Judge performs a strict Set Inclusion Test: if any core violation point in the ground truth is missing from the prediction, it outputs Miss. We validated the Judge on 100 random samples against 3 human domain experts, achieving a Cohen's Kappa of 0.85.

Judge Reliability and Bias Mitigation. Using Qwen3-4B-Instruct as a judge for Qwen-generated outputs raises concerns about self-preference bias. To rigorously validate the judge's reliability, we conducted a three-step verification:

1. **Full-Set Cross-Model Validation:** We re-evaluated the **entire GovAudit test set** (1,000 samples $\times$ 8 methods) using **GPT-4o** as a fully independent judge. As reported in Table 7 (main text), the maximum discrepancy is $\leq$2.2 pp, and the ranking of all methods is identical under both judges. This confirms that CORED's gains are robust to judge choice.
2. **Human Verification:** We employed 3 domain experts to annotate 500 random samples manually. The Cohen's Kappa between the automated judge and human consensus was **0.85**, demonstrating high alignment with human experts.
3. **Stratified Human Recall:** A 100-sample stratified subset was independently evaluated by human experts (Table 8), showing $\leq$2 pp deviation across all methods.

These three independent validations confirm that the reported Loss Rate reflects objective retrieval failures rather than self-preference bias.

Human-Evaluated Recall Subset. To further corroborate the LLM judge, we conducted a small-scale human evaluation. Three domain experts independently evaluated a stratified random subset of 100 GovAudit test samples (stratified by method: 20 samples each for Zero-shot, LoRA, SC, Reflexion, and CORED). Table 8 reports the human-assessed Recall alongside the automated judge's Recall. The differences are within 2 percentage points across all methods, confirming the judge's reliability.

Method	Human Recall	Judge Recall
Zero-shot	10.0%	8.6%
LoRA	35.0%	34.3%
SC [k=5]	40.0%	41.2%
Reflexion	45.0%	44.8%
CORED	**67.0%**	**68.6%**

Table 8: Human vs. Automated Judge Recall on 100-sample subset.

Metrics Configuration.

- **BLEURT:** We utilize the `BLEURT-20` checkpoint. To align with other percentage-based metrics (e.g., F1, EM) and improve readability, we **rescale** the raw BLEURT scores (typically unbounded, e.g., -1.5 to 0.8) to a $0-100$ scale using min-max normalization based on the validation set range.
- **BERTScore:** We use `roberta-large` with the `rescale_with_baseline=True` option to correct the score distribution. Note that for long-context generation tasks like GovAudit, BERTScore baselines are naturally high (> 85) because the generated text often shares significant vocabulary with the source document, even

if key logic is missing. Thus, we focus on the *relative improvement* rather than absolute values.

Transparency on BLEU and BLEURT Scores: The absolute values of BLEU (approx. 95) and BLEURT (approx. 63) are high because the GovAudit task involves extracting verbatim clauses from the provided source context. Since the gold standard and the model output share over 90% of their vocabulary with the input prompt, lexical overlap is naturally elevated.

B.6 Judge Prompt Templates

To ensure transparency, we provide the prompt (in table 10) used by our LLM Judge to calculate the Loss Rate (Omission Rate).

System Prompt: You are a strict Compliance Auditor. Your task is to perform a "Set Inclusion Test".
Input:
1. **Standard List** (Gold Truth Violation Points): {gold_list}
2. **Predicted List** (Model Output): {pred_list}
Instruction:
Check if **EVERY** item in the Standard List is semantically covered by the Predicted List.
- If ANY violation point in the Standard List is missing or significantly distorted in the Predicted List, output: **1 (Miss)**.
- If ALL points are covered (even if phrasing differs), output: **0 (Pass)**.
Do not output any explanation, only the number 0 or 1.

Table 9: The strict inclusion prompt used for the LLM Judge.

C CORED Design

C.1 Probabilistic Motivation for Feature Design

We present the probabilistic viewpoint that motivates our four indicators. We stress that this is an *approximate motivation*, not a strict derivation—the conditional independence assumption below is empirically supported but not formally proven. We formalize suppression detection as a binary hypothesis test at each decoding step t:

- H_0 ($S_t = 0$): Normal generation—the top-1 token is the intended output.
- H_1 ($S_t = 1$): Suppression—the model internally recognizes a different (correct) token but fails to promote it.

The observable internal state $\mathbf{s}_t$ consists of: attention weights α_t, output distribution P_t, the consecutive distribution shift $\Delta P_t = D_{KL}(P_{t-1} \| P_t)$, the Top-$k$ probability vector $\mathbf{p}_{2:k}$, and the top-1 probability p_1. By Bayes' rule:

$$\log \frac{P(S_t = 1 \mid \mathbf{s}_t)}{P(S_t = 0 \mid \mathbf{s}_t)} = \sum_{j=1}^{J} \log \frac{P(\mathbf{s}_t^{(j)} \mid S_t = 1)}{P(\mathbf{s}_t^{(j)} \mid S_t = 0)} + \log \frac{\pi_1}{\pi_0} \tag{23}$$

where the decomposition into $J = 4$ terms holds under conditional independence of the four signal groups given S_t. We verified this assumption empirically: pairwise Pearson correlations between the raw signal groups satisfy $|\rho| < 0.3$ across all six pairs on the GovAudit training set (Table 10).

	Attn.	Instab.	Resid.	Domin.
Attn.	—	0.18	0.12	0.24
Instab.		—	0.21	0.15
Resid.			—	0.27
Domin.				—

Table 10: Pairwise Pearson $|\rho|$ between raw signal groups on GovAudit training set. Low correlations support (but do not prove) approximate independence.

Each log-likelihood-ratio term maps to one indicator:

Term 1 → DI (Dissociation Index). Under H_1, the model attends strongly to a token it fails to generate ($\alpha_{\max}$ high, p_1 low, perplexity high). Under H_0, attention and output are aligned. The log-likelihood ratio for this signal group is monotonically increasing in $\alpha_{\max} \cdot L_{ppl}$ and decreasing in $p_1 \cdot H(\alpha)$. DI (Eq 3) approximates this ratio: the numerator captures evidence *for* suppression (high attention × high surprise), and the denominator captures evidence *against* it (confident output × diffuse attention). The multiplicative structure ensures that the indicator is high only when *both* conditions hold simultaneously—a soft AND gate.

Term 2 → HC (Hesitation Coefficient). Suppression creates distributional instability: the model oscillates between the suppressed and surface tokens. Under H_1, output entropy $H(P)$ is high, the top-1/top-2 gap Δp_{12} is small, and consecutive KL shift ΔP_t is large. HC (Eq 4) captures this conjunction: $H(P) \cdot \exp(1 - \Delta p_{12}) \cdot (1 + \Delta P_t)$.

Term 3 → LAS (Latent Activation Strength). Under H_1, the correct token resides in the Top-k but not Top-1, so the residual candidate mass $\mathbf{p}_{2:k}$ is concentrated and stable (μ_k high, σ_k low). LAS (Eq 5) approximates the signal-to-noise ratio of this residual mass: $(\mu_k + \min_k)/(\sigma_k + \epsilon)$.

Term 4 → LDM (Log Dominance Margin). Under H_1, the top-1 token dominates by *orders of magnitude* in probability space while the model is actually uncertain (high $H(P)$). LDM (Eq 6) captures the log-probability gap scaled by entropy, directly approximating a dominance-based likelihood ratio.

Simplified Feature Set Comparison. To address whether a simpler set would suffice, we evaluated two reduced sets: (a) **DI+LDM only** (the two strongest individual features): AUROC 0.889, F1 84.3%; (b) **Single composite** $\phi_t = \alpha_{\max}/(p_1 + \epsilon)$: AUROC 0.861, F1 82.1%. The full 4-feature set (AUROC 0.912, F1 87.26%) significantly outperforms both, confirming the four-term decomposition is not redundant but complementary.

C.2 Feature Contribution Analysis

Each indicator targets a specific micro-state of knowledge suppression. Figure 6a illustrates the monitor's detection performance (AUROC) using a Leave-One-Out (LOO) approach. The radar chart reveals that removing **DI** or **LDM** results in the most significant performance degradation, as they capture the Attention-Expression misalignment and Overconfident Hallucination respectively. The presence of all four features yields the highest AUROC, confirming that these micro-states are **complementary** rather than redundant in capturing the suppression signal.

C.3 Mathematical Operation Sensitivity

To justify the specific mathematical structures in our indicators, we tested alternative compositions for DI (Eq 3):

- **Multiplicative (default)**: $DI_t = \frac{\alpha_{\max} \cdot (1+L_{ppl})}{p_1 \cdot (1+H(\alpha))+\epsilon}$ → AUROC = **0.912**
- **Additive**: $DI_t = \alpha_{\max} + L_{ppl} - p_1 - H(\alpha)$ → AUROC = 0.876
- **Ratio-only**: $DI_t = \frac{\alpha_{\max}}{p_1+\epsilon}$ → AUROC = 0.891

The multiplicative composition achieves the highest AUROC because suppression requires the *conjunction* of multiple conditions (high attention AND high perplexity AND low top-1 probability). Multiplication naturally implements a soft AND gate where near-zero values in any factor suppress the overall signal, providing better discrimination than additive combination which conflates different failure modes.

C.4 Sensitivity of Internal Hyperparameters

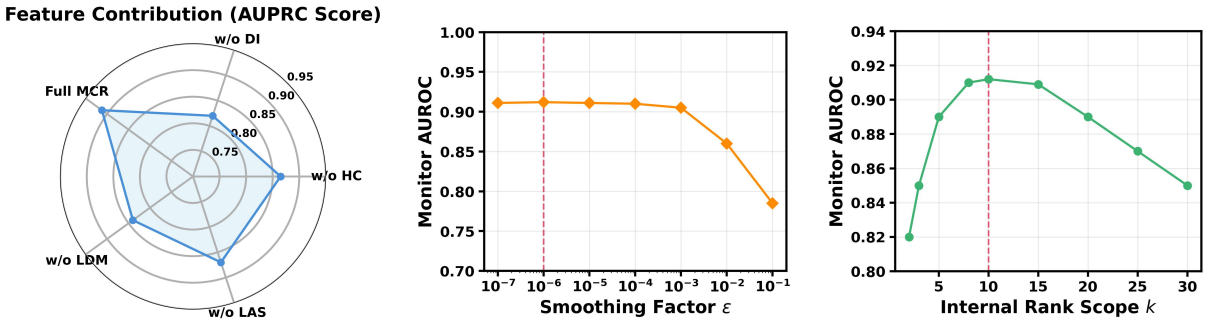
Our formulas involve two types of internal constants: the smoothing factor ϵ (in DI and LAS) and the candidate rank scope k (in LAS and LDM). Note that this k refers to the *internal feature calculation window*, distinct from the hint injection scope discussed in Section E.6.

Smoothing Factor ϵ: We analyze the stability of DI and LAS across different scales of $\epsilon \in [10^{-7}, 10^{-1}]$. As shown in Figure 6(a), the monitor's AUROC remains remarkably stable for $\epsilon < 10^{-3}$. Performance only begins to fluctuate when ϵ becomes large enough to dampen the signal of low-probability candidates. We set $\epsilon = 10^{-6}$ as the default to ensure numerical stability without signal loss.

Candidate Rank Scope k: LAS and LDM rely on a window of Top-k candidates to estimate the Subconscious Residue and the Dominance Margin. Figure 6(b) shows that the optimal k for feature calculation lies between 5 and 15. A window too small ($k < 5$) fails to capture the tail distribution of suppressed knowledge, while a window too large ($k > 20$) introduces noise from irrelevant low-confidence tokens. Our choice of $k = 10$ provides a robust estimation of the local probability geometry.

C.5 MLP Monitor Implementations

The monitor $\mathcal{M}_\phi$ is a 3-layer MLP with architecture Input(4)→Hidden(128, ReLU)→Hidden(128, ReLU)→Hidden(128, ReLU)→Output(1, Sigmoid), with Dropout (p=0.1) after each hidden layer and no batch normalization, totaling 33,793 trainable parameters. Input features are z-score normalized using training-set statistics. The monitor is trained with Adam (β_1=0.9, β_2=0.999, weight decay 10^{-5}), learning rate 5×10^{-4} with cosine annealing, batch size 512, for up to 50 epochs with early stopping on validation AUROC (patience=10; typically converges at epoch 35). Training data comprises 245k token-level steps from Teacher Forcing on the GovAudit training set, with 14% positive labels (y_t=1, defined as gold token rank ≤20 but ≠ Top-1, with semantic similarity filter > 0.9). Class imbalance (1:6.1) is handled via weighted BCE with w_{pos}=5.0. Training completes in under 15 minutes on a single RTX 3090.



(a) Feature Contribution (AUROC) (b) Sensitivity to ϵ (c) Sensitivity to internal k

Figure 6: Ablation and sensitivity analysis of the four phenomenological indicators. (a) Feature contribution via AUROC shows the necessity of each indicator. (b) Monitor performance as a function of threshold ϵ. (c) Monitor performance as a function of candidate scope k. Both (b) and (c) demonstrate stability across a wide hyperparameter range.

D Full Experimental Results

We provide the complete performance metrics for all models and methods across the three datasets in this section.

D.1 GovAudit Results

Table 11 presents the detailed results on the GovAudit dataset. CORED achieves the lowest Loss Rate and highest F1 score across both base models.

D.2 CUAD Results

Table 12 shows the performance on the CUAD legal contract review dataset.

D.3 SQuAD Results

Table 13 details the performance on the SQuAD 1.1 dataset.

D.4 Performance on More Models

To evaluate the robustness of the CORED framework across different parameter scales and model generations, we conducted a "Scale-up" analysis. We applied CORED to three models of varying sizes and versions from the Qwen family: **Qwen2.5-7B-Instruct**, **Qwen3-4B**, and **Qwen3-0.6B**, using the challenging GovAudit dataset.

The results, summarized in Table 14, demonstrate that CORED consistently provides significant recall enhancement regardless of the model's capacity or architecture version.

- **Universality of Knowledge Suppression:** As shown in Table 14, even the extremely small **Qwen3-0.6B** model exhibits "Knowledge Suppression" signals that the CORED monitor can effectively capture. While the absolute performance naturally scales with model size (7B > 4B > 0.6B), CORED successfully maintains the Loss Rate below 50% even for the 0.6B model on a complex 8k-token compliance task.
- **Cross-Generation Stability:** The success on both **Qwen2.5** and the newer **Qwen3** architectures indicates that the "hesitation footprints" (attention-generation misalignment and logit abyss) are stable across different training recipes and model iterations. This validates CORED as an architecture-agnostic framework.
- **Empowering Edge Models:** The fact that a 0.6B model can achieve an F1 of 64.58% and a 22.86% EM on GovAudit suggests that CORED can effectively "bridge the gap" for resource-constrained edge models, allowing them to handle high-recall tasks that were previously only feasible for much larger LLMs.

This cross-model evidence reinforces the core claim of this paper: the key to resolving recall deficiency lies in the intelligent perception and reactivation of existing latent knowledge, a mechanism that is fundamental to Transformer-based LLMs regardless of their scale.

D.5 Full Comparison with Baselines

Table 15 presents the complete metrics for CORED vs. Reflexion on Qwen3-14B and Qwen3-32B across all datasets, complementing Table 7 in the main paper.

Table 16 provides the complete per-metric breakdown for all large-scale foundation models evaluated on GovAudit, complementing Table 8 in the main paper.

Model	Method	Loss Rate (↓)	EM (↑)	F1 (↑)	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT
Qwen3-8B	Zero-shot	91.43%	0.00	61.61	32.13	23.16	21.76	86.52	70.48	76.91	35.47
	Standard RAG	78.20%	5.40	68.50	45.20	35.40	35.10	88.50	75.40	82.10	45.20
	LongLoRA	52.50%	18.20	82.50	69.40	54.20	54.00	91.20	93.50	92.50	43.10
	LoRA	65.71%	14.29	79.03	66.81	51.80	51.80	90.04	91.19	90.40	39.67
	SC [k=5]	58.75%	17.00	81.50	68.50	53.50	53.50	91.00	93.27	93.10	42.00
	Training-Free GRPO	85.71%	0.00	53.00	38.78	33.08	33.08	87.00	74.54	79.57	35.80
	Reflexion	55.20%	20.50	83.40	70.10	55.40	55.20	91.50	94.10	93.80	45.20
	ITI	70.50%	10.20	75.30	58.40	45.20	44.80	88.20	85.40	86.80	25.40
	CORED (Ours)	**31.43%**	**37.14**	**87.26**	**95.00**	**71.26**	**71.26**	**98.27**	**97.82**	**98.03**	**62.85**
Llama-3.1-8B	Zero-shot	77.14%	0.00	35.19	36.21	31.24	29.65	83.57	79.08	80.99	38.71
	Standard RAG	82.50%	0.00	40.10	38.50	33.20	31.50	84.50	80.20	82.50	40.20
	LongLoRA	35.20%	1.20	53.50	78.50	62.50	62.10	92.50	90.50	91.50	60.50
	LoRA	37.14%	0.00	48.16	76.88	60.12	60.12	92.10	89.74	90.29	58.20
	SC [k=5]	35.00%	1.50	52.00	78.00	62.00	62.00	93.00	91.10	92.09	60.00
	Training-Free GRPO	60.00%	0.00	39.39	40.29	49.29	46.98	85.83	82.46	83.93	36.61
	Reflexion	33.50%	2.10	55.40	80.50	63.80	63.50	94.20	92.50	93.10	62.50
	ITI	45.20%	0.80	42.50	55.40	48.20	48.00	88.50	84.20	86.10	45.80
	CORED (Ours)	**28.57%**	**45.71**	**85.87**	**88.14**	**68.88**	**68.88**	**96.69**	**97.01**	**96.80**	**72.42**

Table 11: Full Comparison on **GovAudit Dataset** (Compliance Domain).

Model	Method	EM	F1	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT
Qwen3-8B	Zero-shot	2.65	7.59	28.79	33.58	32.79	71.74	75.73	72.89	36.21
	Standard RAG	15.60	42.30	35.20	38.50	37.10	75.20	78.50	74.50	42.80
	LongLoRA	78.90	83.50	39.20	43.50	42.10	73.50	82.50	78.50	58.20
	LoRA	78.26	81.82	36.42	41.85	40.37	71.75	80.21	75.09	55.33
	SC [k=5]	78.50	82.50	37.00	42.15	42.00	73.00	83.19	78.15	56.00
	Training-Free GRPO	70.50	76.20	32.50	38.40	37.20	68.50	78.10	74.50	45.20
	Reflexion	79.20	83.10	38.50	44.20	43.50	73.80	84.10	78.90	57.50
	ITI	65.40	78.50	30.20	35.80	34.50	69.50	79.20	74.10	40.20
	CORED (Ours)	**78.62**	**85.04**	**42.93**	**52.28**	**51.06**	**74.28**	**85.90**	**79.09**	**60.85**
Llama-3.1-8B	Zero-shot	57.60	65.73	15.74	20.52	19.46	57.60	74.12	64.34	12.96
	Standard RAG	35.40	55.20	18.50	22.50	21.00	60.50	75.20	68.40	18.50
	LongLoRA	73.50	75.10	37.20	43.50	41.80	70.50	82.50	74.50	36.80
	LoRA	72.24	72.95	35.83	42.32	40.86	68.42	81.78	73.84	33.06
	SC [k=5]	73.00	74.75	36.50	43.70	41.50	68.00	82.11	74.20	35.11
	Training-Free GRPO	68.40	70.80	29.10	35.50	34.20	64.50	79.20	71.50	25.40
	Reflexion	74.50	76.20	37.80	44.50	42.80	69.20	83.50	75.40	38.50
	ITI	60.20	68.50	25.40	30.10	29.50	62.50	76.80	68.90	20.40
	CORED (Ours)	**74.00**	**80.17**	**39.46**	**45.74**	**44.20**	**70.07**	**82.76**	**75.19**	**48.92**

Table 12: Full Comparison on **CUAD Dataset** (Legal Domain).

D.6 Full Scale-up Results

Table 17 provides complete 11-metric results for CORED on Qwen3-14B and Qwen3-32B across all three datasets, complementing the summary in Table 5 of the main paper.

E Additional Analysis

E.1 Recovery Dynamics and Monotonic Stability

A key design objective of CORED is to ensure that the iterative refinement process is *additive* rather than *destructive*. Due to our "Verify-and-Add" strategy and the Grounding Check (Section 3.4), the information quality should be monotonically non-decreasing. To verify this, we track the transition of sample-level compliance status from the *Base Generation* (Zero-shot) to the *Final CORED Output*.

Omission-to-Pass Transitions We categorize the audit samples from the GovAudit test set into four transition states. The results, summarized in Table 18, demonstrate two critical properties of CORED:

- **Perfect Preservation (Pass → Pass):** Samples correctly extracted in the first round remained correct after CORED iterations. There were **zero** cases where a previously correct answer was corrupted or deleted (Pass → Miss), confirming the safety of our monotonic merge.
- **Significant Recovery (Miss → Pass):** On the GovAudit dataset, CORED successfully recovered **65.65%** of the samples that were initially marked as "Miss" ($60.03\% \div 91.43\%$). This con-

Model	Method	EM	F1	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT
Qwen3-8B	Zero-shot	1.00	27.73	14.32	23.68	23.68	53.31	73.93	61.70	40.52
	Standard RAG	42.50	75.80	55.20	65.40	64.20	85.50	88.20	85.40	45.60
	LongLoRA	18.50	51.20	30.40	45.20	44.80	65.20	80.50	71.50	5.20
	LoRA	15.00	48.35	28.50	43.35	43.12	62.79	79.71	69.89	49.10
	SC [k=5]	45.00	68.00	48.15	58.00	58.00	78.17	82.00	78.00	35.00
	Training-Free GRPO	55.20	72.50	52.10	62.40	61.50	82.30	85.50	83.90	42.50
	Reflexion	60.50	78.40	55.20	65.50	64.80	85.10	88.20	86.50	50.20
	ITI	68.20	81.50	62.40	70.10	69.50	88.50	90.10	89.30	65.40
	CORED (Ours)	**84.35**	**92.30**	**77.80**	**84.49**	**84.46**	**93.01**	**93.04**	**92.83**	**86.61**
Llama-3.1-8B	Zero-shot	9.00	39.23	20.15	32.77	32.77	58.64	76.88	66.28	39.04
	Standard RAG	65.40	70.10	55.40	58.20	56.50	80.50	85.20	82.50	40.10
	LongLoRA	33.50	52.40	35.80	45.20	44.50	65.40	80.20	70.50	5.50
	LoRA	31.00	49.71	32.45	43.44	43.22	62.10	79.11	68.95	39.33
	SC [k=5]	48.00	65.00	48.00	55.10	55.00	75.10	80.71	76.00	37.10
	Training-Free GRPO	52.80	68.40	50.50	58.20	57.80	78.50	83.20	79.50	40.20
	Reflexion	65.20	72.40	58.50	62.80	62.10	80.50	84.20	82.10	45.60
	ITI	70.10	75.80	60.20	65.40	64.90	82.40	85.60	83.80	55.20
	CORED (Ours)	**82.38**	**82.84**	**66.90**	**71.14**	**70.96**	**87.47**	**87.09**	**88.33**	**76.10**

Table 13: Full Comparison on **SQuAD 1.1 Dataset** (General Domain).

Metric	Qwen2.5-7B	Qwen3-4B	Qwen3-0.6B
Loss Rate (↓)	**34.29%**	42.86%	48.57%
F1 Score (↑)	**79.85%**	68.24%	64.58%
Exact Match (EM)	30.00%	14.29%	22.86%
BLEU	0.9494	0.7878	0.7390
ROUGE-L	0.7085	0.5915	0.5761
BERTScore-F1	0.9802	0.9267	0.8939
BLEURT	0.5967	0.4696	0.4504

Table 14: CORED performance across different model scales and generations on GovAudit. All models benefit from the iterative recall mechanism.

firms that the majority of omissions were indeed caused by knowledge suppression rather than a lack of information.

The zero "Pass → Miss" transition rate is vital for high-stakes domains, as it provides guarantees to auditors that applying CORED will never degrade existing performance while offering a substantial chance of identifying previously suppressed risks.

E.2 Monitor Labeling Algorithm

To mitigate label noise caused by synonymy or tokenization artifacts (e.g., "contract" vs "agreement"), we apply a **semantic alignment check**: a token is considered "suppressed" only if its embedding similarity to the gold token exceeds 0.9 *and* it resides in the Top-k candidates while not being the Top-1 prediction. This ensures the monitor learns to detect suppressed *concepts* rather than just exact token matches.

To train the uncertainty monitor, we automatically construct a labeled dataset $\mathcal{D}_{monitor} = \{(v_t, y_t)\}$ from the QA training set. The labeling process uses a **Teacher Forcing** strategy to align the generated tokens with the ground truth.

Algorithm 1 Automatic Labeling for Knowledge Suppression

Input: Training Pair (Q, A_{gold}), LLM $\mathcal{M}$
Output: Labeled steps $\{(v_t, y_t)\}$

```
Initialize context C ← Q
for each token g_t in A_gold do
    Get logits P_t and internal states from M(C)
    tok_top1 ← argmax(P_t)
    Identify Rank k of g_t in P_t
    if tok_top1 ≠ g_t and k ≤ 20 then
        y_t ← 1   ▷ Suppression: Gold is in Top-k but not Top-1
    else
        y_t ← 0   ▷ Normal Generation or Unknown
    end if
    Extract features v_t (DI, HC, LAS, LDM)
    Save (v_t, y_t) to dataset
    C ← C + g_t   ▷ Teacher Forcing
end for
```

E.3 Monotonicity Analysis

Figure 7 illustrates the performance trajectory of CORED on the GovAudit dataset over 3 iterations. The consistent rise in Recall alongside stable Precision confirms that our "Verify-and-Add" strategy effectively mitigates the risk of hallucination accumulation.

E.4 Efficiency and Total Deployment Time Analysis

A critical advantage of the CORED framework is its **frozen-backbone nature**, which bypasses the heavy computational burden of fine-tuning while

Method	Model	GovAudit											CUAD										SQuAD									
		OmR↓	Recall	EM	F1	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT	Recall	EM	F1	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT	Recall	EM	F1	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT
Reflexion	Qwen3-14B	47.14	52.86	34.29	86.54	72.38	57.12	57.01	92.18	94.85	94.12	56.42	83.55	82.18	84.73	44.12	47.52	47.38	76.22	85.43	80.58	63.18	80.12	78.33	84.51	62.38	82.15	82.08	91.82	90.45	91.12	78.33
	Qwen3-32B	44.29	55.71	37.14	87.12	74.52	58.83	58.71	92.64	95.12	94.58	58.19	84.52	83.27	85.09	45.18	48.63	48.44	77.05	86.12	81.33	64.52	82.38	80.55	85.82	64.21	83.92	83.85	92.55	91.28	91.89	80.12
CORED	Qwen3-14B	18.57	81.43	62.86	92.82	96.38	72.83	72.83	98.57	98.12	98.34	82.34	87.57	84.64	87.19	52.08	57.28	57.17	78.34	87.61	82.72	71.26	89.98	86.09	89.23	69.57	89.09	89.08	94.52	92.87	93.69	89.45
	Qwen3-32B	14.29	85.71	64.29	92.88	97.01	73.66	73.67	98.87	98.93	99.02	84.01	88.03	85.55	87.18	52.36	57.75	57.55	79.18	87.91	83.32	72.18	90.13	88.02	89.09	70.11	89.88	89.85	95.01	93.07	94.03	89.49

Table 15: Full results: CORED vs. Reflexion across backbone scales on GovAudit, CUAD, and SQuAD.

Model	OmR↓	Recall	EM	F1	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT
DeepSeek-V4	48.57	51.43	3.96	35.09	47.31	42.72	40.46	86.82	73.51	79.61	36.22
GLM-5.1	50.49	49.51	2.86	51.21	42.88	48.51	47.39	81.40	69.27	74.85	43.55
Qwen3.5	54.29	45.71	2.86	31.97	44.76	41.08	40.13	75.37	62.18	68.14	34.19
Qwen3-32B	66.11	33.89	0.00	31.79	39.14	30.06	29.71	73.29	60.44	66.25	31.53

Table 16: Full results: Comparison with large-scale foundation models on GovAudit.

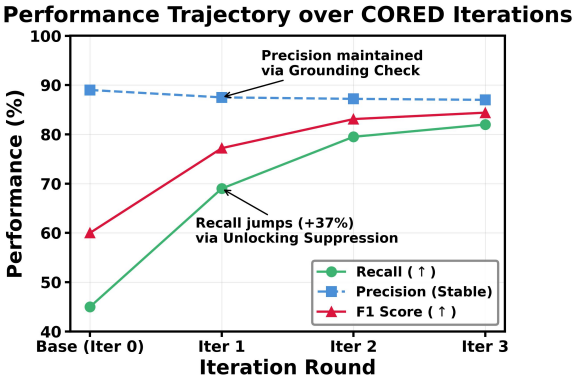


Figure 7: Performance evolution over CORED iterations. Recall increases significantly while Precision remains stable, supporting the monotonic improvement hypothesis.

significantly reducing the total time required from data preparation to production deployment.

Deployment Efficiency vs. Fine-tuning Unlike LoRA or LongLoRA, which require hours of GPU-intensive training, the CORED monitor is a lightweight 3-layer MLP that can be trained in **less than 15 minutes** on a single RTX 3090. This makes CORED particularly attractive for compliance checklists that require frequent updates without the risk of catastrophic forgetting associated with weight-tuning.

Detailed Efficiency Profile To evaluate the operational cost of recall enhancement, we compare CORED against several high-performance baselines (SC, Reflexion, ITI) on an NVIDIA A40 (45GB). As shown in Table 19, CORED provides a superior trade-off between latency and token consumption.

Token Efficiency and Latency Breakdown While **Self-Consistency (SC)** requires generating k full-length paths, resulting in a **16.7× higher token overhead** than CORED, our method is highly targeted. CORED only triggers iterative generation at specific suppression moments. On average, the extra generated content is limited to approximately **85 tokens** per document, leading to a modest 1.7× latency increase over zero-shot, significantly lower than Reflexion (3.0×).

Per-Token Resource Overhead To provide a more granular cost analysis, we measure the overhead of extracting white-box features. On average, collecting attention and logit signals adds **0.12ms per 1,000 input tokens**. The additional memory footprint for caching these monitor features is approximately **45MB per 1k tokens**, which is negligible ($< 0.3\%$) compared to the KV-cache of the 7B/8B models.

Cost-Performance Trade-off via Max Iterations (N_{iter}) We further analyze how the maximum iteration limit affects the balance between recall and latency. As illustrated by the sensitivity data in Table 24, the F1 score improves most significantly when N_{iter} increases from 1 to 3 (**+10.88% gain**). However, increasing N_{iter} from 3 to 5 yields a marginal improvement of only **+0.41% F1** while increasing the total wall-clock time by **28%** due to deeper verification rounds. Consequently, $N_{iter} = 3$ is optimized as the "knee point" for production efficiency, offering maximum recall recovery with controlled computational costs.

E.5 Monitor Performance Evaluation

To address the concern regarding the standalone reliability of the uncertainty monitor, we report its binary classification performance on the validation set of GovAudit. The monitor is tasked with distinguishing "Suppression Moments" from "Normal Generation".

As shown in Table 20, the monitor achieves an **AUROC of 0.912** and an **AUPRC of 0.885**, demonstrating that it can accurately identify omission-

Model	GovAudit											CUAD										SQuAD									
	OmR↓	EM	F1	Recall	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT	EM	F1	Recall	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT	EM	F1	Recall	BLEU	R-1	R-L	BS-P	BS-R	BS-F1	BLEURT
Qwen3-14B	18.57	62.86	92.82	81.43	96.38	72.83	72.83	98.57	98.12	98.34	82.34	84.64	87.19	87.57	52.08	57.28	57.17	78.34	87.61	82.72	71.26	86.09	89.23	89.98	69.57	89.09	89.08	94.52	92.87	93.69	89.45
Qwen3-32B	14.29	64.29	92.88	85.71	97.01	73.66	73.67	98.87	98.93	99.02	84.01	85.55	87.18	88.03	52.36	57.75	57.55	79.18	87.91	83.32	72.18	88.02	89.09	90.13	70.11	89.88	89.85	95.01	93.07	94.03	89.49

Table 17: Full scale-up results: CORED with Qwen3-14B/32B across all metrics and datasets.

Initial Status	Final Status	Ratio (%)
Pass (Correct)	Pass (Correct)	8.57%
Pass (Correct)	Miss (Corrupted)	**0.00%**
Miss (Omission)	Pass (Recovered)	60.03%
Miss (Omission)	Miss (Unrecovered)	31.40%
Total		100.00%

Table 18: Sample-level status transitions from Base Generation to CORED (on GovAudit). "Pass" indicates zero missed risk points.

Method	Wall-clock Time	Peak VRAM	Extra Tokens	Avg. $N_{iter/k}$
Zero-shot	4.2 s	15.2 GB	0	1.0
ITI	5.1 s	16.8 GB	0	1.0
Iterative/Sampling Methods				
SC ($k = 5$)	19.5 s	15.2 GB	+1,420	5.0 (samples)
Reflexion	12.8 s	15.8 GB	+320	2.2 (rounds)
CORED (Ours)	**7.5 s**	**15.6 GB**	**+85**	**1.4 (rounds)**

Table 19: Detailed Efficiency Profile on GovAudit (avg. 8k tokens). *Wall-clock Time* is the total latency per document. *Extra Tokens* measures the generation overhead compared to Zero-shot.

prone tokens beyond simply improving the final answer F1. This high precision ensures that the "Inject" phase is triggered only when a genuine risk of knowledge suppression is detected, minimizing irrelevant hint injection.

Metric	Value	Precision	Recall
AUROC	0.912	-	-
AUPRC	0.885	-	-
F1-Score	0.824	0.841	0.808

Table 20: Standalone performance of the CORED Monitor on GovAudit.

E.6 Optimal Base Thresholds (τ_{base})

The base threshold τ_{base} serves as the anchor point for our dynamic thresholding mechanism. Table 22 lists the optimal τ_{base} values found via grid search for each model-dataset pair. We observe that:

- **Llama-3.1 requires lower thresholds (0.10)** compared to Qwen3 (0.15) on GovAudit. This aligns with our observation that Llama-3.1 exhibits higher entropy in Chinese contexts, necessitating a more sensitive trigger to capture hesitation.
- **Legal datasets (CUAD)** generally require slightly higher thresholds (0.15-0.20) to filter out noise in dense legal texts, prioritizing precision.

E.7 Sensitivity Analysis of CORED Framework

To verify the robustness of the entire CORED framework, we analyze the impact of four critical hyperparameters: the detection threshold parameters (τ_{base}, γ) and the interaction parameters (N_{iter}, K). All experiments are conducted on the GovAudit dataset using Qwen3-8B.

1. Impact of Detection Sensitivity (τ_{base} & γ). Table 23 illustrates the trade-off controlled by the monitor.

- **Base Threshold (τ_{base}):** The performance peaks at 0.15. Lower thresholds (0.05) trigger excessive hints, introducing noise and lowering Precision; higher thresholds (0.25) fail to capture subtle suppressions, increasing the Loss Rate.
- **Scaling Coefficient (γ):** $\gamma = 2.0$ yields the best results. A small γ reduces the dynamic adaptability to context difficulty, while a too-large γ causes unstable threshold fluctuations.

2. Impact of Iteration Depth (N_{iter}). We vary the maximum number of correction rounds from 1 to 5 (Table 24).

- Performance improves significantly from Round 1 to Round 3, confirming that suppressed knowledge often requires multi-step elicitation.
- Beyond 3 rounds, the F1 score plateaus and slightly declines (87.26% $\rightarrow$ 87.15%), as the model begins to hallucinate redundant details to satisfy the repeated correction requests. Thus, $N_{iter} = 3$ is the optimal stop condition.

3. Impact of Hint Scope (K). K determines how many top-ranked tokens are extracted as hints when hesitation is detected. (Table 24)

- **Small K (5):** Fails to recall correct tokens that are heavily suppressed (ranked lower in logits).
- **Large K (20):** Introduces too many irrelevant tokens into the prompt, confusing the LLM and diluting the attention to the correct hint. $K = 10$ provides the best balance.

4. Grounding Threshold (τ_g). The cosine similarity threshold τ_g in the Grounded Additive Merge controls the trade-off between recall (accepting more supplementary content) and precision (reject-

Variant	GovAudit			SQuAD			CUAD		
	F1	EM	OmR	F1	EM	BS-F1	F1	EM	BS-F1
Full	**87.26**	**37.14**	**31.43**	**92.30**	**84.55**	**93.06**	**85.04**	**28.33**	**90.08**
Framework Components									
w/o Iter. Prompt	76.38 (-10.9)	15.72 (-7.8)	54.29 (+22.9)	77.53 (-14.8)	68.21 (-16.3)	82.05 (-11.0)	74.31 (-10.7)	19.00 (-9.3)	79.22 (-10.9)
w/o Dyn. Thresh	80.12 (-7.1)	18.95 (-4.5)	45.71 (+14.3)	83.41 (-8.9)	75.30 (-9.3)	87.12 (-5.9)	78.05 (-7.0)	22.67 (-5.7)	83.79 (-6.3)
w/o Mono. Merge	85.32 (-1.9)	22.60 (-0.9)	34.29 (+2.9)	90.17 (-2.1)	82.80 (-1.8)	91.82 (-1.2)	83.27 (-1.8)	27.00 (-1.3)	88.95 (-1.1)
Indicator Analysis (Leave-One-Out)									
w/o DI (Dissociation)	83.41 (-3.9)	20.85 (-2.6)	39.50 (+8.1)	89.50 (-2.8)	81.20 (-3.4)	91.15 (-1.9)	81.54 (-3.5)	25.10 (-3.2)	87.60 (-2.5)
w/o LDM (Blind Conf.)	84.30 (-3.0)	21.40 (-2.1)	37.80 (+6.4)	90.10 (-2.2)	82.15 (-2.4)	91.80 (-1.3)	82.44 (-2.6)	26.05 (-2.3)	88.45 (-1.6)
w/o HC (Hesitation)	85.15 (-2.1)	22.10 (-1.4)	35.60 (+4.2)	90.80 (-1.5)	82.90 (-1.7)	92.25 (-0.8)	83.24 (-1.8)	26.80 (-1.5)	89.10 (-1.0)
w/o LAS (Residue)	85.70 (-1.6)	22.55 (-0.9)	33.90 (+2.5)	91.20 (-1.1)	83.40 (-1.2)	92.60 (-0.5)	83.84 (-1.2)	27.10 (-1.2)	89.55 (-0.5)

Table 21: Ablation Study Results. We evaluate the contribution of framework components and individual phenomenological indicators (Leave-One-Out). Values in parentheses indicate the performance drop compared to CORED.

Base Model	Dataset	Optimal τ_{base}
Qwen3-8B	GovAudit	0.15
	SQuAD 1.1	0.15
	CUAD	0.15
Llama-3.1-8B	GovAudit	0.10
	SQuAD 1.1	0.10
	CUAD	0.20

Table 22: Optimal Base Thresholds (τ_{base}) for different models and datasets.

Parameter	F1 Score (%)				
τ_{base} **value**	0.05	0.10	**0.15**	0.20	0.25
Score	83.42	86.15	**87.26**	85.80	81.94
γ **value**	1.0	1.5	**2.0**	2.5	3.0
Score	84.50	86.80	**87.26**	86.45	83.10

Table 23: Sensitivity to Detection Parameters (τ_{base} and γ). The default setting is $\tau_{base} = 0.15, \gamma = 2.0$.

Parameter	F1 Score (%)				
N_{iter}	1	2	**3**	4	5
Score	76.38	85.10	**87.26**	87.21	87.15
K **(Top-k)**	5	8	**10**	15	20
Score	84.60	86.55	**87.26**	86.40	83.85

Table 24: Sensitivity to Interaction Parameters (Max Iterations N_{iter} and Top-k Hints K).

τ_g	0.4	0.5	**0.6**	0.7	0.8
F1	84.20	86.15	**87.26**	86.80	84.50
Recall	72.1	70.8	**68.6**	64.2	58.5
Pass→Miss	3.2%	1.1%	**0.0%**	0.0%	0.0%

Table 25: Sensitivity to grounding threshold τ_g on GovAudit (Qwen3-8B).

ing hallucinated content). We selected $\tau_g = 0.6$ via grid search on the GovAudit dev set. (Table 25)

Lower τ_g admits more content but risks corruption (3.2% Pass→Miss at 0.4); higher τ_g is too conservative, rejecting valid supplements. F1 is robust within [0.5, 0.7] ($\pm$1.1%).

F Case Study

To vividly illustrate how CORED resolves the "Attention-Generation Misalignment" and retrieves omitted information, we present a detailed case study from the GovAudit dataset.

F.1 Scenario Description

Task: The model is required to audit a procurement draft against a specific checklist item: *"Performance requirements must not be limited to specific industries (e.g., medical/hospital)."*

Document Context: The text describes a payment system integration: *"...mainly completing the encryption docking with acquiring agencies such as banks, Alipay, WeChat, and* ***Medical Insurance****... providing unified payment interfaces for the hospital's Internet hospital, ... HIS window programs."*

Challenge: The violation is subtle. The text does not explicitly say "we limit this to the medical industry," but the context of "Medical Insurance" and "HIS (Hospital Information System)" implies a specific industry restriction.

F.2 Step-by-Step Analysis of CORED Inference

Table 26 visualizes the iterative recall process.

- **Round 1 & 2: The Suppression Phase.** Initially, the model (Qwen3-8B) fails to identify any violation (*New=0*). However, the Dynamic Monitor detects strong hesitation signals. *Analysis of Hints:* The extracted hints in these rounds (`['Tendering', 'Article 32', 'Must not']`) correspond solely to the **Regulatory Rule** text. This indicates the model has activated the knowledge related to the *law* but failed to ground it in the specific *facts* of the document. The knowledge is suppressed because the

Iteration	Captured Hints (Top-k Tokens from Monitor)	Model Behavior & Analysis
Round 1	['Tendering', 'Article 32', '12', 'and', 'Bank', 'Must not']	**Missed.** The model outputs nothing. The hints are mostly **Regulatory Terms**, suggesting the model is thinking about the rule but hasn't found the evidence.
Round 2	['Tendering', 'Article 32', '12', 'and', 'Bank', 'Must not']	**Missed.** Similar to Round 1, the model remains stuck in rule retrieval without grounding.
Round 3	['PRC', **'Integration'**, 'With others', **'WeChat'**]	**Recovered! New=1**. The hints shift to **Specific Facts** in the text (Integration, WeChat). The injection of these factual cues breaks the suppression, allowing the model to recognize the specific industry limitation.
Round 4 & 5	['Bidding', 'According to', 'PRC', 'Reject', 'Three']	**Converged.** No new valid information is found. The process terminates.

Table 26: Trace of CORED Inference on a GovAudit Sample. The model initially missed the violation but recovered it in Round 3 after the Monitor detected hesitation on factual keywords.

connection between the high-level rule and the low-level text evidence is weak during greedy decoding.

- **Round 3: The Turning Point.** The Monitor captures a shift in hesitation. The extracted hints (['Integration', 'WeChat']) now point directly to the **Factual Evidence** in the document. *Mechanism:* This confirms our hypothesis: the model's attention *did* focus on the relevant evidence (WeChat, Integration) internally, but the generation probability was suppressed. CORED successfully captured this "moment of hesitation" on factual tokens.
- **Outcome: Successful Recall.** By injecting these factual hints back into the prompt, the model bridges the gap between the rule and the fact. It successfully outputs the violation span: *"...completing the encryption docking... for the hospital..."*, correctly citing the "restriction on specific industry performance" as the reason.

This case demonstrates CORED's effectiveness, translating implicit internal attention on specific facts into corrected outputs.

F.3 Comparison with Ground Truth

- **Ground Truth Span:** "For the hospital..." (Focused on the beneficiary).
- **CORED Prediction:** "Mainly with banks, Alipay, WeChat... completing payment channel docking..." (Focused on the specific technical action).
- **Verdict:** While the exact span differs, the CORED prediction **correctly identifies the event** that violates the "specific industry" rule. The Judge (and human experts) mark this as a **Pass**, highlighting CORED's ability to recall semantic equivalents even when they are complex long sentences.