LLMs Already Know What Not to Share: Span-Level Contextual Integrity via Probing and Attention Masking

Anonymous authors
Paper under double-blind review

Abstract

As large language model (LLM) agents act on users' behalf, they are increasingly given access to personal information. Yet access does not imply permission to disclose: under Contextual Integrity (CI), the appropriateness of each information span depends on the current task. Given their large-scale training, we hypothesize that LLMs possess an implicit capacity for CI, reflected in contextual privacy signals encoded during prompt processing. To formalize this capacity, we propose the *Contextual Linear Privacy Hypothesis*, which models privacy-relevant information as a linear component of the span representation whose strength varies with context, and characterizes how changing context shifts the representation of a fixed span. We operationalize this hypothesis with layer-specific linear probes and introduce probe-guided attention masking, which prevents generated tokens from attending to cached positions classified as restricted without modifying the base model or generating additional privacy reasoning. Across five models, probes average 90.5% accuracy and move in the expected direction on 96.2% of 226 context-flip pairs. Our masking achieves the highest Completeness on every model (32.3% versus 24.7% for CI-CoT), while averaging 178 generated tokens compared with 475 for CI-CoT. These results show that contextual privacy signals in LLM representations can support targeted and efficient disclosure control.

1 Introduction

Large language models (LLMs) are evolving from conversational assistants into agents that plan, use tools, and act on behalf of users (Schick et al., 2023; Wang et al., 2024; Xi et al., 2025). To draft emails, fill out forms, or contact third parties, these agents are routinely given access to users' private information, yet they may inadvertently disclose information that the task does not require (Mireshghallah et al., 2024; Shao et al., 2024). We focus on this contextual privacy risk, formalized by Nissenbaum (2004) as Contextual Integrity (CI): an information flow is appropriate only if it conforms to the norms of its context, such as its recipient and purpose. An employee's salary, for instance, should be shared when updating payroll records but withheld when reporting department-level statistics.

Existing approaches to enforcing CI fall into three groups, each with its own limitation. Reasoning-based methods prompt the model to deliberate about each information flow before responding (Ghalebikesabi et al., 2025; Lan et al., 2025), which substantially increases inference cost. Training-based methods instill CI through fine-tuning or reinforcement learning (Fan et al., 2024; Lan et al., 2025), but require curated data and modify the model itself. Representation-based methods intervene on hidden states without training (Zhang et al., 2026; Wang et al., 2026; Hu et al., 2026), yet they either steer all tokens along a global direction or make a single decision for the entire task. Across these approaches, CI is supplied to the model from the outside, and even methods that read its hidden states act on the task or the model as a whole rather than on the specific information that should be withheld.

So our goal is different: we examine whether the model can internally identify which specific spans should not be shared in a given context, and use this judgment to decide what to withhold. Since LLMs acquire broad knowledge of social norms through pretraining, a model may already judge

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

which information is appropriate to disclose while it reads a task. We therefore propose the Contextual Linear Privacy Hypothesis, which states that the privacy status of a span in its context is linearly accessible from its representation. Based on this hypothesis, we train a linear probe on the representation of each span and examine whether it separates spans that may be shared from those that should be withheld. To make use of this prefill signal, we further design an attention masking method: spans classified as private are masked from attention during generation, making the model less likely to reproduce them in its output.

Our contributions are summarized as follows:

- We present the first analysis of how context shapes the representation of individual information spans. We formulate the *Contextual Linear Privacy Hypothesis*, representing privacy as a context-dependent linear component and context changes as directional shifts.
- We introduce probe-guided attention masking, the first application of attention steering to inference-time contextual disclosure control. It masks probe-identified KV-cache positions without updating the base model or generating privacy reasoning.
- Our proposed linear probes achieve 90.5% average accuracy across five models, with 96.2% of 226 context-flip pairs shifting in the predicted direction. Probe-guided masking achieves the highest Completeness on every model, averaging 32.3% versus 14.5% for Base and 24.7% for CI-CoT, while using substantially fewer tokens than CI-CoT. Ablations further support the importance of context and hard span-selective masking.

2 Related Work

2.1 Contextual Integrity in LLMs

Contextual integrity (CI) (Nissenbaum, 2004) considers an information flow appropriate when it conforms to contextual norms. Existing benchmarks identify frequent CI violations in conversations (Mireshghallah et al., 2024; Cheng et al., 2024) and agent trajectories (Shao et al., 2024); we adopt the allowed–restricted attribute setting of Lan et al. (2025). Existing defenses either elicit explicit or multi-agent reasoning (Ghalebikesabi et al., 2025; Lan et al., 2025; Li et al., 2025), filter inputs with an external minimizer (Bagdasarian et al., 2024), or adapt the model through fine-tuning or reinforcement learning (Fan et al., 2024; Lan et al., 2025). The first two introduce additional inference cost or components, whereas training-based methods require curated data and parameter updates. Representation-based methods operate on internal model states. Activation-steering methods identify directions associated with privacy-compliant outputs and shift hidden states along those directions (Zhang et al., 2026; Wang et al., 2026; Batra et al., 2025). However, this global intervention does not model how context changes the privacy status of specific input information and may also degrade utility. PrivGate (Hu et al., 2026) probes whether the current task requires CI reasoning and invokes that reasoning selectively, but its prediction serves as a task-level routing signal rather than a direct privacy intervention.

Our work instead investigates whether a frozen LLM already encodes a context-dependent judgment of whether each attribute is allowed or restricted. We recover this signal from span representations with a linear probe and use it to guide targeted attention masking during generation, connecting the model's latent CI capability to an effective intervention.

2.2 Probing and Attention Steering

Probing classifiers are widely used to test whether particular concepts can be decoded from a model's internal representations (Alain & Bengio, 2016; Belinkov, 2022). Motivated by the linear representation hypothesis (Park et al., 2024), prior work finds that LLMs encode properties such as truthfulness along linear directions, even when their outputs do not reflect them (Burns et al., 2022; Azaria & Mitchell, 2023; Marks & Tegmark, 2023). This view motivates activation steering, which modifies hidden states along learned directions at inference time (Turner et al., 2024; Zou et al., 2023; Rimsky et al., 2024). ITI (Li et al., 2023) uses probes to select and steer attention heads, while Arditi et al. (2024) identify a direction mediating refusal. Attention steering intervenes on information routing rather than directly modifying hidden states. PASTA (Zhang et al., 2024a) reweights selected attention heads toward user-specified spans, AutoPASTA (Zhang et al., 2024b) lets the model identify

these spans to improve contextual faithfulness dynamically steer attention toward instructions; attention boosting (Guardieiro et al., 2026) similarly adds biases to selected attention logits. These methods amplify attention to improve faithfulness or instruction following.

We are the first to apply attention steering as an inference-time intervention against the disclosure of restricted information. Prior attention-steering methods amplify selected spans to improve faithfulness or instruction following (Zhang et al., 2024a;b; Guardieiro et al., 2026); in contrast, we use contextual privacy signals recovered by a linear probe to suppress attention to spans that should be withheld. This redistributes the model's attention over the remaining context, reducing its reliance on restricted spans and the likelihood that their content appears in the output.

3 Problem Setup

We study contextual privacy as a span-level disclosure decision. The same span may be *allowed* in one context and *restricted* in another; the relevant unit is therefore a span together with the context needed to interpret it.

Let $\mathbf{x} = (x_1, \ldots, x_T)$ be a tokenized prompt, and let $\mathcal{S}(\mathbf{x}) = \{S_1, \ldots, S_m\}$ denote its candidate spans. Each span S_i occupies positions a_i through b_i, with $C_i = \mathbf{x}_{<a_i}$ denoting its preceding context. We assign each span–context pair a task-specific label $Y(S_i, C_i) \in \{-1, +1\}$, where $+1$ denotes *restricted* and -1 denotes *allowed*.

Given a frozen model, we investigate whether it can judge the contextual integrity of each span and whether this judgment is already encoded in its prefill activations. If so, we further examine whether we can make use of this signal during generation.

4 Contextual Linear Privacy Hypothesis

Privacy is not an intrinsic property of a span but depends on its surrounding context. Since causal self-attention incorporates preceding context into the span representation during prefill, we hypothesize that this contextual dependence is reflected in the model's latent space.

Contextual Privacy Representation. We first formalize how preceding context affects the representation of a span. For a span–context pair (S, C) defined in Section 3, causal self-attention incorporates C into the representation of S during prefill. Its representation after the first l Transformer layers is therefore

$$\mathbf{h}_S^{(l)}(C) = F_l(S, C), \tag{1}$$

allowing the same span to acquire different representations under different preceding contexts.

Motivated by the linear representation hypothesis, we hypothesize that privacy-relevant information contributes to the contextualized representation along a direction $\mathbf{v}_P^{(l)}$:

$$\mathbf{h}_S^{(l)}(C) = \mathbf{h}_{\text{base},S}^{(l)}(C) + \alpha_S^{(l)}(C)\mathbf{v}_P^{(l)} + \boldsymbol{\epsilon}_S^{(l)}(C), \qquad \|\mathbf{v}_P^{(l)}\|_2 = 1. \tag{2}$$

Here, $\mathbf{h}_{\text{base},S}^{(l)}(C)$ captures other content- and context-dependent features, $\alpha_S^{(l)}(C)$ denotes the strength of the proposed privacy-related component, and $\boldsymbol{\epsilon}_S^{(l)}(C)$ contains residual variation. In particular, the strength of the component can change even when the span content remains fixed.

Contextual Shift. The decomposition above describes a span under a single context, whereas our central claim concerns how its privacy status changes when only the context changes. To characterize this, we introduce a reference context $\widetilde{C}$ and define

$$\Delta\mathbf{h}_S^{(l)}(C; \widetilde{C}) = \mathbf{h}_S^{(l)}(C) - \mathbf{h}_S^{(l)}(\widetilde{C}). \tag{3}$$

This difference holds span content fixed and isolates the representational change associated with replacing $\widetilde{C}$ by C. Its projection onto the proposed privacy direction is

$$\mathrm{CI}_{P,S}^{(l)}(C; \widetilde{C}) = \left(\mathbf{v}_P^{(l)}\right)^\top \Delta\mathbf{h}_S^{(l)}(C; \widetilde{C}). \tag{4}$$

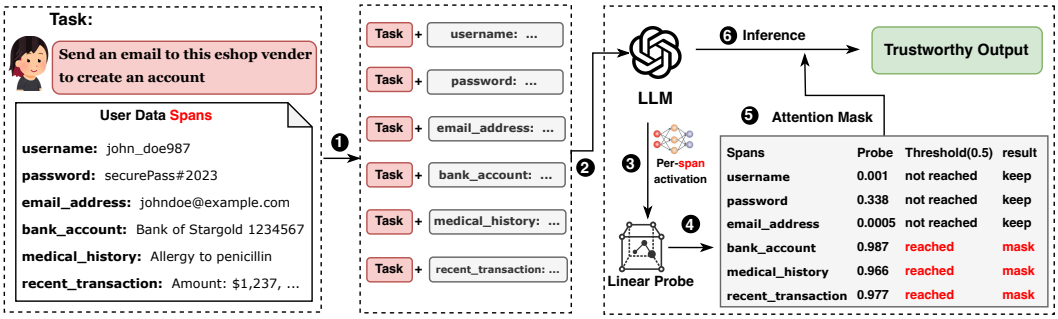


Figure 1: Overview of probe-guided attention masking. First, each attribute is paired with the task to form an isolated input to the frozen LLM. Then a linear probe scores its mean-pooled span activation against a threshold τ. Finally, spans reaching τ are masked from attention during generation.

Thus, $\alpha_S^{(l)}(C)$ describes the privacy-related component under a particular context, whereas $\mathrm{CI}_{P,S}^{(l)}(C;\widetilde{C})$ measures the relative displacement between two contexts along the privacy direction.

Scope of the Hypothesis. The full span representation also contains content-, context-, and residual components whose projections may affect both linear decoding and contextual shift. To delineate the scope of our hypothesis, Appendix A derives sufficient conditions under which linear accessibility and the interpretation of $\mathrm{CI}_{P,S}^{(l)}$ remain reliable despite this projected nuisance.

5 Methodology

Figure 1 illustrates our method, which consists of two stages: a linear probe first classifies each span from its prefill activations (steps 1–4), and spans classified as restricted are then masked from attention during generation (steps 5–6).

Linear Probe. Building on the hypothesis introduced in Section 4, we operationalize linear privacy accessibility by learning a privacy direction from contextualized span representations.

Specifically, we freeze the language model and extract layer-l activations during prefill. We instantiate the span representation $\mathbf{h}_{S_i}^{(l)}(C_i)$ in eq. (1) through mean pooling:

$$\mathbf{h}_{S_i}^{(l)}(C_i) \equiv \mathbf{r}_i^{(l)} = \frac{1}{|\mathcal{I}(S_i)|} \sum_{t \in \mathcal{I}(S_i)} \mathbf{h}_t^{(l)}, \qquad p_i^{(l)} = \sigma\left(\mathbf{w}_l^\top \mathbf{r}_i^{(l)} + b_l\right). \tag{5}$$

The normalized probe weight $\widehat{\mathbf{v}}_P^{(l)} = \mathbf{w}_l / \|\mathbf{w}_l\|_2$ serves as an empirical estimate of the hypothesized privacy direction $\mathbf{v}_P^{(l)}$. When the two directions align, the probe projection reads out $\alpha_{S_i}^{(l)}(C_i)$ up to other representational variation projected onto the same direction, as characterized in Appendix A.

We train $\mathbf{w}_l$ and b_l with binary cross-entropy, using $y_i = 1$ for *restricted* and $y_i = 0$ for *allowed*. During inference, the predictions determine the token positions to be masked:

$$\widehat{y}_i = \mathbb{I}\left[p_i^{(l)} \geq \tau\right], \qquad \mathcal{M} = \bigcup_{i:\widehat{y}_i=1} \mathcal{I}(S_i), \tag{6}$$

where τ is the decision threshold and $\mathcal{M}$ contains the positions of spans classified as *restricted*.

Inputs Template. Because causal representations incorporate only preceding tokens, we place the user task and its descriptions before the information list, ensuring that every target span follows its full task context. We then encode each span independently using this shared context followed by the target span alone. This construction retains the context needed to interpret the span while excluding variation introduced by unrelated attributes and their ordering. It is therefore designed to reduce the projected nuisance characterized in Appendix A, making $\mathbf{h}_S^{(l)}(C)$ more directly reflect the span's context-dependent privacy component.

Probe-Guided Attention Masking. To utilize the span-level detections produced by the probe, we convert the predicted set $\mathcal{M}$ into an attention mask. Our intervention is inspired by attention boosting (Guardieiro et al., 2026), which modifies selected attention logits through additive biases. Rather than boosting the selected positions, we assign a negative-infinite bias to cached positions belonging to spans classified as *restricted*.

Specifically, for query position t, cached position j, layer r, and attention head h, we compute

$$\widetilde{a}_{t,j}^{(r,h)} = \frac{(\mathbf{q}_t^{(r,h)})^\top \mathbf{k}_j^{(r,h)}}{\sqrt{d_h}} + \begin{cases} -\infty, & j \in \mathcal{M}, \\ 0, & j \notin \mathcal{M}. \end{cases} \tag{7}$$

The prompt is prefilled without any mask. During generation, we then apply the mask at every decoding step across all layers and heads, so the selected cached positions receive zero attention weight while the remaining prompt and generated tokens remain accessible.

Since modifying attention naively at every step slows down inference, we implement the mask as a static source-token vector computed once before generation, which adds only 10–20% overhead to generation time; details and pseudocode are given in Appendix C.

6 Experiments

Models and Data. We evaluate five instruction-tuned models from four families: Qwen2.5-7B, Qwen2.5-14B, Llama-3.1-8B, Mistral-7B, and Gemma-2-9B. Experiments use the synthetic contextual integrity benchmark of Lan et al. (2025), where each example contains a task context, a user request, and a list of user attributes, each labeled as allowed or restricted in that context. We keep the original split of 590, 66, and 73 examples (3,265, 361, and 398 attribute spans) for training, validation, and test. To evaluate how probes respond to changes in context, we further construct a context-flip set of 226 pairs with GPT-4 Turbo, in which the same span is allowed under one user task and restricted under another (Section 6.1; Appendix D). This set is used only for evaluation.

Probing. We construct one linear probe input per attribute as described in Section 5 and train the probe on mean-pooled span activations of the frozen model. A span is classified as restricted, and thus masked during generation, when the predicted probability of the probe is at least a decision threshold τ. We select the layer, the ℓ_2 regularization strength, and $\tau \in \{0.5, 0.6, \ldots, 0.9\}$ jointly by validation accuracy. The selected τ is used for both probe evaluation and masking and the test set is used only for final evaluation.

Masking and Baselines. At generation time, the model receives the full attribute list, and each attribute flagged by its probe is masked from its field key to the end of its value in all layers and heads. BASE uses the same privacy instruction as our method but applies no masking, so the two differ only in the intervention. CI-CoT (Lan et al., 2025) instead prompts the model to reason about the recipient, purpose, and appropriateness of each attribute before answering.

Evaluation. Following Lan et al. (2025), we evaluate only the final answer. Integrity (I) holds when no restricted value appears, Utility (U) when all required values appear, and Completeness (C) when both hold. We report the percentage of test examples satisfying each metric, and Cost as the average number of generated tokens. Hyperparameters, prompts, and decoding settings are listed in Appendix B.

6.1 Main Results

Linear Probe Results. We first evaluate whether contextual privacy labels can be recovered from span representations. Table 1 reports the test performance of the linear probes, with the layer and threshold chosen on the validation set. Across all five models, the probes achieve 89.9–91.7% accuracy and 89.4–91.6% F1, with precision and recall both close to 90% on average. These consistent results across model families support the hypothesis that contextual privacy is linearly accessible from span representations.

Table 1: Test performance of linear probes on span-level contextual privacy classification. The probing layer (selected / total) and decision threshold τ are chosen on the validation set.

Model	Layer	τ	Accuracy	Precision	Recall	F1
Qwen2.5-7B	22/28	0.7	89.9	88.8	90.7	89.7
Qwen2.5-14B	30/48	0.5	91.7	90.0	93.3	91.6
Llama-3.1-8B	17/32	0.5	89.9	89.6	89.6	89.6
Mistral-7B	18/32	0.6	89.9	91.8	87.0	89.4
Gemma-2-9B	22/42	0.5	90.7	91.5	89.1	90.3
Average	–	–	90.5	90.4	89.9	90.1

Table 2: Probe behavior on paired contexts that differ only in the user task. Endpoint accuracy (%) and AUROC are computed over all 452 span–context instances; Both Correct (%) and Direction (%) are computed over the 226 pairs.

Model	Layer	τ	Endpoint Acc.	AUROC	Both Correct	Direction
Qwen2.5-7B	22/28	0.7	72.6	0.812	45.6	96.5
Qwen2.5-14B	30/48	0.5	74.3	0.830	48.7	96.5
Llama-3.1-8B	17/32	0.5	71.9	0.791	43.8	95.6
Mistral-7B	18/32	0.6	75.2	0.827	50.4	95.1
Gemma-2-9B	22/42	0.5	76.3	0.837	54.0	97.3
Average	–	–	74.1	0.819	48.5	96.2

Contextual Shift on Paired Contexts. To test the contextual shift in Section 4, we use GPT-4 Turbo to build 226 context-flip pairs in the format of Lan et al. (2025). Each pair places the same target span S_i in two contexts that share the roles and information list but differ in the user task, so that the span is allowed under C_i^A but restricted under C_i^R (details in Appendix D). Applying the probes of Table 1 without retraining, we measure how often the shift points toward the restricted side:

$$\text{Direction} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\Big[\mathbf{w}_l^\top \Delta \mathbf{h}_{S_i}^{(l)}\big(C_i^R; C_i^A\big) > 0\Big]. \tag{8}$$

As shown in Table 2, Direction exceeds 95% on all five models: context moves the span representation toward the privacy direction in nearly every pair. The shift, however, does not always cross the decision boundary. Average endpoint accuracy is 74.1%, and only 48.5% of pairs are classified correctly at both ends on average. Although the input template is fixed, this independently constructed, out-of-distribution set may introduce additional projected nuisance, as characterized in Appendix A. The contextual shift thus holds consistently, while a fixed threshold calibrated on the original data does not always separate the two contexts.

Probe-Guided Attention Masking. We next evaluate whether the detected span-level signal can support an effective inference-time intervention. Table 3 compares our probe-guided masking method with CI-CoT and generation without masking. We report Integrity (I), Utility (U), Completeness (C), and inference cost measured by the average number of generated tokens.

Completeness is our primary evaluation criterion. Our method obtains the highest C score on every model, improving the macro-average from 24.66% with CI-CoT and 14.52% without masking to 32.33%. The stronger privacy protection incurs a reduction in Utility. Nevertheless, the consistent improvement in the primary Completeness metric indicates a more favorable overall outcome under the contextual privacy objective. Moreover, our average cost is comparable to generation without masking (177.6 versus 174.6 tokens) and substantially lower than CI-CoT (475.3 tokens).

Compatibility with CI-CoT. Our masking intervention can also be composed with CI-CoT. As detailed in Appendix E.3, this combination increases average C from 32.33% to 36.99%, but also

Table 3: Main results of probe-guided attention masking. I, U, and C denote Integrity, Utility, and Completeness (%) following synthetic dataset in Lan et al. (2025); Cost is the average number of generated tokens. **Base**: Same instruction as our method without attention masking.

Model	Base				CI-CoT				Masking (Ours)			
	$I\uparrow$	$U\uparrow$	$C\uparrow$	Cost$\downarrow$	$I\uparrow$	$U\uparrow$	$C\uparrow$	Cost$\downarrow$	$I\uparrow$	$U\uparrow$	$C\uparrow$	Cost$\downarrow$
Qwen2.5-7B	37.0	64.4	20.6	160	35.6	61.6	21.9	472	72.6	53.4	**35.6**	151
Qwen2.5-14B	27.4	60.3	17.8	168	46.6	56.2	21.9	559	72.6	48.0	**34.2**	161
Llama-3.1-8B	24.7	68.5	12.3	189	61.6	57.5	30.1	543	65.8	49.3	**31.5**	217
Mistral-7B	8.2	71.2	4.1	233	38.4	67.1	26.0	453	50.7	61.6	**30.1**	241
Gemma-2-9B	54.8	42.5	17.8	123	58.9	48.0	23.3	349	78.1	39.7	**30.1**	117
Average	30.4	**61.4**	14.5	**175**	48.2	58.1	24.7	475	**67.9**	50.4	**32.3**	178

Table 4: Probe accuracy (%) with full contextualized inputs and isolated key–value pairs. The same trained probes and thresholds are used without retraining. Δ Acc. denotes the change from full-context to isolated accuracy.

Model	Layer	τ	Full Acc.	Isolated Acc.	Δ Acc.	Isolated F1
Qwen2.5-7B	22/28	0.7	89.9	63.8	-26.1	68.8
Qwen2.5-14B	30/48	0.5	91.7	67.1	-24.6	59.2
Llama-3.1-8B	17/32	0.5	89.9	76.9	-13.1	76.5
Mistral-7B	18/32	0.6	89.9	74.4	-15.6	71.0
Gemma-2-9B	22/42	0.5	90.7	73.1	-17.6	67.1
Average	–	–	90.5	71.1	-19.4	68.5

raises the average generation cost from 178.0 to 523.4 tokens. We therefore report it as a compatibility analysis while retaining direct-prompt masking as our primary efficiency-oriented setting.

6.2 Ablations

6.2.1 Probe

Context Removal. We first examine whether the probe relies on contextualized span representations rather than on span content alone. To this end, we set $\widetilde{C} = \varnothing$ in eq. (3): using the same trained probes and decision thresholds, we remove all preceding context at test time and keep only the key–value pair of each span. Table 4 compares performance on the original contextualized inputs and on these context-free inputs.

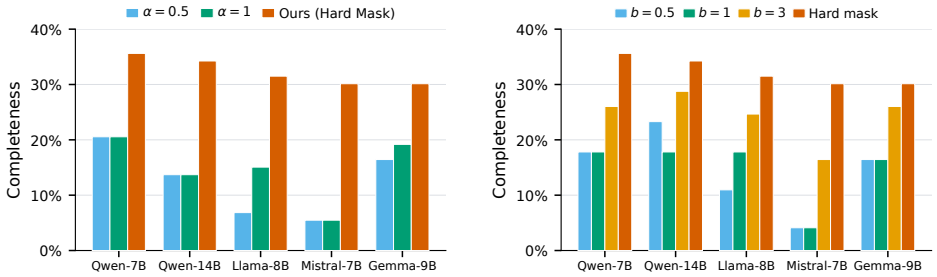
Removing context decreases accuracy for every model, with drops ranging from 13.1 to 26.1 percentage points and an average drop of 19.4 points. The remaining accuracy is expected because some fields carry privacy-relevant cues even in isolation, allowing their labels to be partially inferred from span content alone. Importantly, the isolated inputs retain both the field key and value; the substantial degradation therefore shows that full-context probe performance cannot be explained solely by shortcuts based on field identity or lexical content. Instead, privacy labels become considerably less linearly accessible without the preceding context, consistent with the contextual dependence in Section 4. This result characterizes a loss of linear accessibility rather than the complete absence of privacy-related information.

Probe Architecture. We next test whether the linear probe underfits the privacy signal by comparing it with two higher-capacity alternatives: an MLP with a 128-dimensional hidden layer and a rank-16 adapter. To isolate the effect of probe architecture, all alternatives are evaluated at a fixed layer for each model. Table 5 reports test accuracy, with Δ measured relative to the linear probe.

The MLP improves average accuracy by only 0.30 percentage points, while the adapter decreases it by 0.40 points, and neither alternative improves consistently across models. The linear probe

Table 5: Test accuracy (%) of different probe architectures at a fixed layer for each model. Δ denotes the difference from the corresponding linear probe.

Model / Layer	Linear	MLP-128	Δ	Adapter-r16	Δ
Qwen2.5-7B / L21	90.45	**91.96**	+1.51	89.95	−0.50
Qwen2.5-14B / L30	91.71	91.71	0.00	**91.96**	+0.25
Llama-3.1-8B / L25	**87.94**	87.94	0.00	86.68	−1.26
Mistral-7B / L18	**89.95**	89.45	−0.50	89.45	−0.50
Gemma-2-9B / L22	90.70	**91.21**	+0.51	90.70	0.00
Average	90.15	**90.45**	+0.30	89.75	−0.40



(a) Activation steering (b) Attention suppression

Figure 2: Ablations on the form of intervention. (a) Completeness (C) under activation steering with strength α, compared with our hard masking. (b) Completeness under soft attention steering with bias b and under hard masking, using the same probe decisions.

therefore captures most of the accessible signal while remaining simpler and directly aligned with our linear privacy hypothesis.

6.2.2 Attention Masking

Attention Masking versus Activation Steering. We compare our masking with activation steering (Zhang et al., 2026), which adds $\gamma \mathbf{v}_l$ to the residual stream of one layer, where $\mathbf{v}_l$ is the difference between the mean representations of allowed and restricted spans (Appendix E.1). As shown in Figure 2a, averaged over the five models, steering keeps C near the BASE level (12.6 and 14.8 for $\gamma = 0.5$ and 1), far below the 32.3 of our masking. The direction is estimated from prefill span representations and likely does not transfer to generation, where hidden states follow a different distribution; moreover, a global direction cannot single out the spans that should be withheld.

To ensure that this comparison is not limited by estimating steering directions from our training set, we additionally reproduce CI-parametric activation steering (Wang et al., 2026) using its original training data and evaluate the learned directions on our test set under identical settings. As detailed in Appendix E.2, steering remains below masking on Qwen and produces degenerate generations on Llama and Mistral, showing that it is less effective in this setting and can destabilize model outputs on some models.

Hard Masking versus Soft Attention Steering. Our intervention removes the selected positions from attention entirely. To test whether weaker suppression suffices, we instead subtract a finite bias β from their attention logits, scaling their unnormalized attention weights by $e^{-\beta}$. As shown in Figure 2b, averaged over the five models, $\beta \in \{0.5, 1\}$ leaves C at the BASE level (14.5 and 14.8), and even $\beta = 3$ reaches only 24.4, below the 32.3 of hard masking, which is best on every model. Weakening attention is thus insufficient and remains less effective than masking the spans entirely.

Oracle Attention Masking. To isolate the effect of span-selection errors and estimate the headroom of our masking mechanism, we replace probe predictions with ground-truth labels and mask all and only the restricted spans. As shown in Figure 3, averaged across the five models, oracle masking increases C from 14.5% without masking to 43.6%. Our probe-guided method reaches 32.3%,

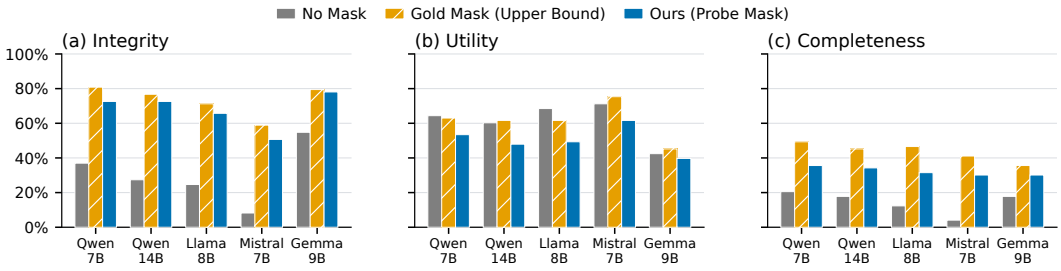


Figure 3: Effect of the masking source. Gold Mask uses ground-truth labels and gives an upper bound on the gain of attention masking; Ours uses probe predictions.

recovering approximately 61% of the oracle improvement. Meanwhile, oracle masking leaves U essentially unchanged at 61.4%, suggesting that correctly masking restricted spans incurs little average utility loss. The utility reduction under probe-guided masking is therefore likely driven primarily by span-selection errors, particularly false-positive masking of allowed spans.

Layer-wise Masking Coverage. We further examine whether masking must be applied at every layer by comparing all-layer masking with a partial variant that applies the mask only from the selected probe layer through the final layer. All-layer masking increases average C from 30.4% to 32.3% at essentially identical generation cost, supporting its use as our default. Per-model results are provided in Appendix E.4.

7 Discussions

7.1 Limitations and Future Works

Limitations. Our implementation is optimized for a structured template that places task context before each candidate span and probes spans independently, thereby reducing irrelevant representational variation. Substantially different or unstructured inputs may increase the projected nuisance described in Appendix A and reduce probe reliability; schema-aware parsing or input canonicalization provide natural extensions to such settings. Since the method is modular, it can also be activated selectively for privacy-sensitive tasks. When no restricted spans are present, the intervention offers limited benefit and occasional false positives may mask useful context. A task-level router such as PrivGate (Hu et al., 2026) could therefore invoke probing and masking only when relevant.

Future Work. Future work will extend the evaluation to domain-specific settings such as healthcare, legal, and enterprise applications, where contextual privacy norms require specialized knowledge. Domain-adapted models and probes trained on such data may develop stronger CI capabilities and improve span-level classification accuracy. We also plan to expand the dataset with more diverse input formats and template variations, enabling the probe and masking pipeline to transfer more reliably across heterogeneous and less structured inputs.

7.2 Conclusion

In this work, we investigated how LLMs represent the contextual privacy of individual spans and how this signal can guide generation. The Contextual Linear Privacy Hypothesis formalizes privacy relevance as a context-dependent component of span representations, which experiments across five models show to be linearly accessible during prefill. Probe-guided attention masking leverages this signal to suppress access to restricted spans, improving integrity and completeness at substantially lower generation cost than reasoning-based baselines. Overall, latent contextual privacy signals provide a practical basis for lightweight and targeted inference-time control.

AI USE STATEMENT

In this work, we used generative AI tools to generate a synthetic dataset, to implement methods, and to assist with translation. We have not used generative AI tools to develop the research idea, propose the hypothesis, or design the proposed method, and the remaining tasks with required disclosure, such as formulating mathematical claims or proofs, cleaning or reformatting datasets, and qualitative data analysis, are not applicable to this work. Additionally, we used generative AI tools to edit the paper for readability and grammar, to draft and restructure parts of the text, to identify relevant literature and summarize the limitations of existing methods, to format references, and to create or edit software code. We have reviewed all AI-assisted work. The context-flip set was generated with GPT-4 Turbo, and its labels were manually verified by the authors (Appendix D); code written with the assistance of Codex was carefully reviewed by the authors to ensure its correctness; and all literature identified with AI assistance was checked against the original papers. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, and all experiments are conducted on synthetic data. The benchmark of Lan et al. (2025) and our context-flip set consist of generated scenarios whose names and attribute values are fictional, so no real personal information is collected, used, or released. Our goal is to reduce the inappropriate disclosure of private information by LLM agents. Since our method lowers but does not eliminate such disclosure, it should complement rather than replace other privacy safeguards in deployed systems. We are not aware of any other ethical concerns or conflicts of interest related to this work.

REPRODUCIBILITY STATEMENT

We have made several efforts to ensure the reproducibility of our work. All experiments use publicly available open-weight models and the public contextual integrity benchmark of Lan et al. (2025) with its original data splits (Section 6). The construction of span-level probe inputs, probe training, and the search over layers, regularization strengths, and decision thresholds are described in Section 5 and Appendix B, together with the prompts and decoding settings, including fixed random seeds, used for all methods. The attention masking implementation is detailed in Appendix C, with pseudocode in Algorithm 1, and the activation steering baseline is described in Appendix E.1. The generation procedure, filtering rules, and generation prompt of the context-flip set are given in Appendix D, and the conditions underlying our hypothesis are stated in Appendix A. The supplementary material also includes the experiment logs underlying the main-results table, the model outputs for each evaluated method, and our span-level probe predictions. We will publicly release the anonymized source code upon publication.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

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

Eugene Bagdasarian, Ren Yi, Sahra Ghalebikesabi, Peter Kairouz, Marco Gruteser, Sewoong Oh, Borja Balle, and Daniel Ramage. Airgapagent: Protecting privacy-conscious conversational agents. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pp. 3868–3882, 2024.

Shourya Batra, Pierce Tillman, Samarth Gaggar, Shashank Kesineni, Kevin Zhu, Sunishchal Dev, Ashwinee Panda, Vasu Sharma, and Maheep Chaudhary. Salt: Steering activations towards leakage-free thinking in chain of thought. *arXiv preprint arXiv:2511.07772*, 2025.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*, 2022.

Zhao Cheng, Diane Wan, Matthew Abueg, Sahra Ghalebikesabi, Ren Yi, Eugene Bagdasarian, Borja Balle, Stefan Mellem, and Shawn O'Banion. Ci-bench: Benchmarking contextual integrity of ai assistants on synthetic data. *arXiv preprint arXiv:2409.13903*, 2024.

Wei Fan, Haoran Li, Zheye Deng, Weiqi Wang, and Yangqiu Song. Goldcoin: Grounding large language models in privacy laws via contextual integrity theory. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 3321–3343, 2024.

Sahra Ghalebikesabi, Eugene Bagdasarian, Ren Yi, Itay Yona, Ilia Shumailov, Aneesh Pappu, Chongyang Shi, Laura Weidinger, Robert Stanforth, Leonard Berrada, et al. Privacy awareness for information-sharing assistants: A case-study on form-filling with contextual integrity. *Transactions on Machine Learning Research*, 2025.

Vitoria Guardieiro, Avishree Khare, Adam Stein, and Eric Wong. Instruction following by principled boosting attention of large language models, 2026. URL https://arxiv.org/abs/2506.13734.

Runshan Hu, Yukun Dong, Yingying Huangfu, Ruohan Zhao, Yi Xie, and Tieyan Li. Privgate: Steering contextual integrity in LLMs via latent space geometry. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=oKizUOP7E4.

Guangchen Lan, Huseyin A Inan, Sahar Abdelnabi, Janardhan Kulkarni, Lukas Wutschitz, Reza Shokri, Christopher Brinton, and Robert Sim. Contextual integrity in LLMs via reasoning and reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=Xm57IXqU0n.

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

Wenkai Li, Liwen Sun, Zhenxiang Guan, Xuhui Zhou, and Maarten Sap. 1-2-3 check: Enhancing contextual privacy in LLM via multi-agent reasoning. In Leon Derczynski, Jekaterina Novikova, and Muhao Chen (eds.), *Proceedings of the The First Workshop on LLM Security (LLMSEC)*, pp. 115–128, Vienna, Austria, August 2025. Association for Computational Linguistics. ISBN 979-8-89176-279-4. URL https://aclanthology.org/2025.llmsec-1.9/.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, 2023.

Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. Can llms keep a secret? testing privacy implications of language models via contextual integrity theory. In *International Conference on Learning Representations*, volume 2024, pp. 1892–1915, 2024.

Helen Nissenbaum. Privacy as contextual integrity. *Wash. L. Rev.*, 79:119, 2004.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models, 2024. URL https://arxiv.org/abs/2311.03658.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, 2024.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

Yijia Shao, Tianshi Li, Weiyan Shi, Yanchen Liu, and Diyi Yang. Privacylens: Evaluating privacy norm awareness of language models in action. *Advances in Neural Information Processing Systems*, 37:89373–89407, 2024.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248.

Haoran Wang, Li Xiong, and Kai Shu. Do llms know what is private internally? probing and steering contextual privacy norms in large language model representations. *arXiv preprint arXiv:2604.00209*, 2026.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. A survey on large language model based autonomous agents. *Frontiers of computer science*, 18(6):186345, 2024.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, et al. The rise and potential of large language model based agents: A survey. *Science China information sciences*, 68(2):121101, 2025.

Jie Zhang, Chenxu Niu, Zhefeng Nan, Yangyan Xu, and Jinta Weng. Steering representations, safeguarding privacy: A cross-modal privacy protection method for generative ai. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 34719–34727, 2026.

Qingru Zhang, Chandan Singh, Liyuan Liu, Xiaodong Liu, Bin Yu, Jianfeng Gao, and Tuo Zhao. Tell your model where to attend: Post-hoc attention steering for llms. In *International Conference on Learning Representations*, volume 2024, pp. 42411–42430, 2024a.

Qingru Zhang, Xiaodong Yu, Chandan Singh, Xiaodong Liu, Liyuan Liu, Jianfeng Gao, Tuo Zhao, Dan Roth, and Hao Cheng. Model tells itself where to attend: Faithfulness meets automatic attention steering, 2024b. URL https://arxiv.org/abs/2409.10790.

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

A Conditions for Linear Accessibility

Beyond the privacy-related component, the representation in eq. (2) contains other content-, context-, and residual variation. We characterize how the projection of these components onto the proposed privacy direction affects linear accessibility and the interpretation of contextual shift.

Projected Nuisance. Define the projected nuisance term and projection score as

$$\begin{aligned} n_S^{(l)}(C) &= \left(\mathbf{v}_P^{(l)}\right)^\top \left[\mathbf{h}_{\text{base},S}^{(l)}(C) + \boldsymbol{\epsilon}_S^{(l)}(C)\right], \\ s_S^{(l)}(C) &= \left(\mathbf{v}_P^{(l)}\right)^\top \mathbf{h}_S^{(l)}(C) = \alpha_S^{(l)}(C) + n_S^{(l)}(C). \end{aligned} \tag{9}$$

Suppose that, for the span–context pairs under consideration, there exist τ_l, c_l, $\gamma_l > 0$, and $\eta_l \geq 0$ such that

$$Y(S,C)\big(\alpha_S^{(l)}(C) - \tau_l\big) \geq \gamma_l, \qquad \Big|n_S^{(l)}(C) - c_l\Big| \leq \eta_l. \tag{10}$$

Here, γ_l denotes the separation of the privacy-related coefficient, while η_l bounds nuisance variation projected onto the same direction.

Effect on Linear Accessibility. Using the threshold $t_l = \tau_l + c_l$, we obtain

$$\begin{aligned} Y(S,C)\big[s_S^{(l)}(C) - t_l\big] &= Y(S,C)\big[\alpha_S^{(l)}(C) - \tau_l\big] + Y(S,C)\big[n_S^{(l)}(C) - c_l\big] \\ &\geq \gamma_l - \Big|n_S^{(l)}(C) - c_l\Big| \geq \gamma_l - \eta_l. \end{aligned} \tag{11}$$

Therefore, $\eta_l < \gamma_l$ guarantees a positive linear classification margin. When $\eta_l \geq \gamma_l$, the bound no longer guarantees linear separability, although it does not imply that the privacy-related component is absent.

Effect on Contextual Shift. Let

$$\Delta\alpha_S^{(l)}(C;\widetilde{C}) = \alpha_S^{(l)}(C) - \alpha_S^{(l)}(\widetilde{C}). \tag{12}$$

The contextual projection then satisfies

$$\mathrm{CI}_{P,S}^{(l)}(C;\widetilde{C}) = \Delta\alpha_S^{(l)}(C;\widetilde{C}) + n_S^{(l)}(C) - n_S^{(l)}(\widetilde{C}). \tag{13}$$

If both contexts satisfy eq. (10), then

$$\begin{aligned} \Big|\mathrm{CI}_{P,S}^{(l)}(C;\widetilde{C}) - \Delta\alpha_S^{(l)}(C;\widetilde{C})\Big| &= \Big|n_S^{(l)}(C) - n_S^{(l)}(\widetilde{C})\Big| \\ &\leq \Big|n_S^{(l)}(C) - c_l\Big| + \Big|n_S^{(l)}(\widetilde{C}) - c_l\Big| \leq 2\eta_l. \end{aligned} \tag{14}$$

Consequently, the sign of the contextual projection agrees with the sign of the privacy-coefficient change whenever

$$\Big|\Delta\alpha_S^{(l)}(C;\widetilde{C})\Big| > 2\eta_l. \tag{15}$$

When this condition is not satisfied, projected nuisance may obscure or reverse the observed direction of the contextual shift.

For a context change from *allowed* to *restricted*, the separation condition gives

$$Y(S,C) = +1, \quad Y(S,\widetilde{C}) = -1 \quad \Longrightarrow \quad \Delta\alpha_S^{(l)}(C;\widetilde{C}) \geq 2\gamma_l. \tag{16}$$

It follows that

$$\mathrm{CI}_{P,S}^{(l)}(C;\widetilde{C}) \geq 2(\gamma_l - \eta_l) > 0 \qquad \text{whenever} \qquad \eta_l < \gamma_l. \tag{17}$$

Scope. The quantities γ_l and η_l are analytical descriptors of a fixed hypothesized decomposition rather than separately identifiable empirical effects. The conditions above are sufficient but not necessary and apply only to the representation family over which the corresponding bounds hold.

Thus, projected nuisance reduces the guaranteed linear margin by η_l and perturbs the contextual shift by at most $2\eta_l$.

B Experiments Details

Experimental Setup. We evaluate our method on five instruction-tuned language models: Qwen2.5-7B-Instruct, Qwen2.5-14B-Instruct, Llama-3.1-8B-Instruct, Mistral-7B-Instruct-v0.2, and Gemma-2-9B-it. Experiments are conducted on the Contextual Integrity Synthetic Dataset, using a fixed context-level split of 590/66/73 contexts for training, validation, and testing, corresponding to 3,265/361/398 annotated spans. There is no context overlap across the three partitions, and the test set is accessed only after model selection is frozen.

All experiments are run on a single NVIDIA RTX PRO 6000 Blackwell Max-Q GPU with 96 GB of memory. Models are loaded in BF16 without quantization, using PyTorch 2.9.1, Transformers 5.4.0, CUDA 12.8, and SDPA attention. Activation extraction and generation both use a batch size of 16.

Generation uses stochastic decoding with temperature 0.7, top-$p = 1.0$, and a maximum of 2,048 newly generated tokens. We use seed 42, set once at the beginning of each independent experimental condition without reseeding individual batches. Native model-specific EOS tokens are preserved. Integrity, Utility, and Completeness are evaluated only over the text inside the answer phase, and generation cost is measured by the average number of model-emitted output tokens.

Probe Training. Span activations are mean-pooled without centering or normalization. Each probe minimizes

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{BCE}(z_i, y_i) + \frac{\lambda}{d}\,\|\mathbf{w}\|_2^2, \tag{18}$$

where z_i is the probe logit and d is the hidden dimension; the bias is not regularized. We train for 80 epochs with full-batch AdamW (learning rate 0.05, zero initialization, no weight decay), searching all layers, $\lambda \in \{10^{-4}, 10^{-3}, 10^{-2}\}$, and $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$. All five selected configurations use $\lambda = 10^{-2}$. F1 treats restricted spans as the positive class.

Attention Masking. Probe decisions on the isolated inputs are mapped to the matching spans in the full generation prompt. Masking adds -10^4 to the attention logits of the masked source positions, both at the final prefill query and at every decoding step. Model parameters, hidden states, and output logits are left unchanged. Cost excludes the prefill passes used for probing.

Prompts. All prompts are inserted immediately before `User Input`. BASE and our method share the privacy instruction below across all models. For CI-CoT, we use the prompt of Lan et al. (2025) without modification.

Privacy Instruction

Share all and only the information necessary to complete the task, and do not disclose information that is inappropriate for the recipient, purpose, or context.

For a set of masked source positions $\mathcal{M} \subseteq \{1, \ldots, L\}$, we define the masked attention logit for the active query q as

$$\widetilde{a}_{q,k}^{(l,h)} = a_{q,k}^{(l,h)} - \beta\,\mathbb{I}[k \in \mathcal{M}], \tag{19}$$

where β is the masking bias; hard masking corresponds to $\beta \to \infty$. During prefill, the mask is applied only to the final prompt query, whereas each decoding step contains only one active query. Because $\mathcal{M}$ contains source positions only, generated tokens remain unmasked.

Both MASKEDPREFILL and MASKEDDECODESTEP apply eq. (19) at every layer and attention head. The former masks only the final prompt query, while the latter masks the current single-token query using the existing KV cache.

C EFFICIENT IMPLEMENTATION AND RUNTIME ANALYSIS

A direct implementation of eq. (7) may repeatedly construct dense masks over all attention heads and token positions. We reduce this overhead through batched one-time probing, a static source-token mask, and standard KV-cache reuse.

One-Time Batched Probing. For each candidate attribute A_i, we construct an isolated probe input $\mathrm{Serialize}(C, T, \{A_i\})$ containing the context, user task, and that attribute alone. All isolated inputs are evaluated in a single batch before generation. If A_i is classified as *restricted*, its corresponding interval R_i, extending from the field key to the end of its value, is added to the masked set $\mathcal{M}$. The resulting source-token mask is fixed and reused throughout generation, without repeated probe evaluation.

Efficient Attention Masking. All prompt-to-prompt attention remains unchanged except at the final prefill query, whose output produces the first generated token. The same mask is subsequently applied to each decoding query while reusing the standard KV cache. It is stored as a one-dimensional vector over source positions and broadcast across attention heads; newly generated positions are assigned zero mask values and therefore remain accessible.

In practice, we implement the $-\infty$ bias in eq. (7) using a finite bias $-\beta$, with $\beta = 10^4$. This makes attention to masked positions numerically negligible and corresponds to the hard-mask limit of the soft-steering ablation in Section 6.2.2.

Algorithm 1 Accelerated Probe-Guided Attention Masking

Require: Frozen model F; prompt $X = (C, T, \mathcal{A})$; candidates $\{(A_i, S_i, R_i)\}_{i=1}^{K}$; probe $(\mathbf{w}, b)$ at layer $l^\star$; threshold τ; masking bias β; maximum generation length $T_{\max}$
Ensure: Generated response Y

$\mathcal{B} \leftarrow [\,]$
for $i = 1, \ldots, K$ **do**
  $X_i^{\text{iso}} \leftarrow \text{Serialize}(C, T, \{A_i\})$
  $\mathcal{B} \leftarrow \mathcal{B} \parallel [X_i^{\text{iso}}]$
end for
$\{\mathbf{H}_i^{(l^\star)}\}_{i=1}^{K} \leftarrow \textsc{BatchedPrefill}(F, \mathcal{B})$
$\mathcal{M} \leftarrow \varnothing$
for $i = 1, \ldots, K$ **do**
  $\mathbf{h}_i \leftarrow \text{MeanPool}(\mathbf{H}_i^{(l^\star)}, S_i)$
  if $\sigma(\mathbf{w}^\top \mathbf{h}_i + b) \geq \tau$ **then**
    $\mathcal{M} \leftarrow \mathcal{M} \cup R_i$
  end if
end for
$(\mathbf{z}, \mathcal{KV}) \leftarrow \textsc{PrefillFinalQueryMask}(F, X, \mathcal{M}, \beta)$
$Y \leftarrow [\,]$
for $t = 1, \ldots, T_{\max}$ **do**
  $y_t \sim \text{Decode}(\mathbf{z})$
  if $y_t = \text{EOS}$ **then**
    break
  end if
  $Y \leftarrow Y \parallel [y_t]$
  $(\mathbf{z}, \mathcal{KV}) \leftarrow \textsc{MaskedDecodeStep}(F, y_t, \mathcal{KV}, \mathcal{M}, \beta)$
end for
return Y

Computational Complexity. Let K denote the number of candidate spans per context ($K = 5.45$ on average), P_i the length of the i-th isolated probe input, N the full generation-prompt length, and M the number of generated tokens. Let L, H, and d denote the number of layers, attention heads, and hidden dimension, respectively.

The additional computation for the isolated probe prefills is

$$\Delta\mathcal{C}_{\text{probe}} = O\left(L \sum_{i=1}^{K} \left(P_i d^2 + P_i^2 d\right)\right). \tag{20}$$

The linear classifier itself costs only $O(Kd)$ and is negligible relative to the Transformer forward passes.

Applying the mask to the final prefill query costs $O(LHN)$. Across decoding steps, its additional cost is

$$\Delta\mathcal{C}_{\text{mask}} = O(LHN) + O\left(LH \sum_{t=1}^{M} (N + t)\right) = O\left(LH(N + MN + M^2)\right). \tag{21}$$

Therefore, relative to standard inference, the total additional computation is

$$\boxed{\Delta\mathcal{C} = O\left(L \sum_{i=1}^{K} \left(P_i d^2 + P_i^2 d\right)\right) + O\left(LH(N + MN + M^2)\right)}. \tag{22}$$

The masking term consists only of applying an additive bias to existing attention logits and does not require rerunning the Transformer or recomputing cached keys and values.

Wall-Clock Overhead. Table 6 reports the measured runtime under the same inference setting. Probe Prefill includes the batched isolated inputs, Total is the sum of Probe Prefill and Masked Generation, and Relative Overhead compares Total against Base Generation.

Table 6: Wall-clock runtime of probe-guided attention masking. Times are reported in seconds and rounded to one decimal.

Model	Probe Prefill	Base Gen.	Masked Gen.	Total	Overhead
Qwen2.5-7B	7.0	55.0	53.4	60.4	1.10×
Qwen2.5-14B	15.0	103.4	100.4	115.4	1.12×
Llama-3.1-8B	8.0	64.6	71.5	79.5	1.23×
Mistral-7B	8.0	67.9	73.0	81.0	1.19×
Gemma-2-9B	9.0	72.0	70.6	79.6	1.11×
Average	**9.4**	**72.6**	**73.8**	**83.2**	**1.15×**

Although probing processes K isolated inputs, it does not incur K full generations: these inputs are short ($P_i \ll N$), evaluated in parallel as a batch, and require no autoregressive decoding. Mask application is also inexpensive relative to attention projection, value aggregation, and MLP computation. Consequently, the measured average overhead is

$$R = \frac{\overline{T}_{\text{probe}} + \overline{T}_{\text{masked}}}{\overline{T}_{\text{base}}} = \frac{9.4 + 73.8}{72.6} \approx 1.15 \times . \tag{23}$$

Model-level variation mainly reflects differences in generation length, including the longer-tailed generations observed for Llama. More generally, wall-clock latency also depends on GPU parallelism, kernel fusion, memory bandwidth, padding, and other constant factors not captured by asymptotic complexity.

D Context-Flip Set

The context-flip set is designed to isolate the effect of context on the representation of a fixed span. Each pair places the same target span in two contexts that differ only in the user task, such that the span is allowed in one context and restricted in the other.

D.1 Construction

Formulation. The set consists of 226 pairs $\mathcal{D}_{\text{flip}} = \{(F_i, S_i, C_i^A, C_i^R)\}_{i=1}^{226}$, where F_i is the information list shared by both contexts and S_i is the target span within F_i. Within each pair, the roles, information list, field order, target field, field values, and target span are identical; only the user task differs between the allowed context C_i^A and the restricted context C_i^R.

Generation. Each pair is produced in a single GPT-4 Turbo call, which (1) writes a neutral information list of 5–8 attributes; (2) designates a target field and the exact target span; (3) fixes the sender, recipient, data subject, domain, and scenario; (4) writes two user tasks around the same information list, one for which the target is necessary and appropriate to share and one for which its disclosure is inappropriate; and (5) labels the target as allowed and restricted under the two tasks, respectively. The prompt forbids the tasks from containing the target value, the label, or explicit instructions on whether to share. The data cover six domains (Finance, Healthcare, Education, Work, Hospitality, and eCommerce) and include both email and chat tasks. The second and third batches additionally require coverage of different information families, reducing templated pairs that differ only in names or numbers.

Filtering. We keep a pair only if (1) both contexts share exactly the same information list and target span; (2) the two user tasks differ; (3) one context is labeled allowed and the other restricted; (4) the target span can be located exactly within its field value; (5) all roles and required fields are present; and (6) its information list is neither an exact nor a high-overlap lexical duplicate of the original benchmark, of earlier synthetic data, or of other pairs in the same batch.

D.2 EXAMPLE

We show a pair from the set below. Both contexts share the same roles and information list, and the target field (highlighted) is identical; only the user task differs. The salary is necessary for updating payroll records, but is not needed for reporting department-level statistics. Since the span is identical in both contexts, a correct contextual shift should increase its probe score from the allowed to the restricted context, i.e., $\mathbf{w}_l^\top \big(\mathbf{h}_S(C^R) - \mathbf{h}_S(C^A)\big) > 0$, as measured by eq. (8).

Shared Elements

Domain: Work **Scenario:** chat **Sender:** Alexa Thompson, HR Manager
Recipient: Payroll Department **Data subject:** John Smith

Information list:

```
employee_id: JS-7482
position_title: Senior Developer
department: Software Development
hire_date: March 15, 2018
current_annual_salary: $120,000
```

Allowed Context (C^A)

Please update the payroll records to reflect John Smith's new salary adjustment.

Label: allowed. The salary is required to update the payroll records.

Restricted Context (C^R)

Could you send me the latest financial statistics for our department?

Label: restricted. Department-level statistics do not require John Smith's individual salary.

E ADDITIONAL ABLATIONS

E.1 ACTIVATION STEERING BASELINE

We adapt the centroid-difference steering of Zhang et al. (2026) to our span-level labels. This baseline tests whether a single privacy direction, applied uniformly to all tokens, can match the effect of span-specific attention masking.

Steering direction. For each layer l, the steering direction is the difference between the mean representations of allowed and restricted spans:

$$\mathbf{v}_l = \text{mean}\big\{\mathbf{h}_S^{(l)} : S \text{ allowed}\big\} - \text{mean}\big\{\mathbf{h}_S^{(l)} : S \text{ restricted}\big\}. \tag{24}$$

The representations are the mean-pooled span activations of the training split, i.e., the same features used to train our probe. Whereas Zhang et al. (2026) compute class centroids over whole texts, we compute them over spans, matching the granularity of our labels. Following their formulation, the direction points from restricted toward allowed representations and is not normalized, so that it retains the scale of the hidden states.

Intervention. During generation, we modify the residual stream of a single layer l as

$$\mathbf{h}_l' = \mathbf{h}_l + \gamma\, \mathbf{v}_l, \qquad \gamma \in \{0.5, 1\}. \tag{25}$$

The intervention is applied at all prompt positions during prefill and at every decoding step. Unlike our method, it does not depend on probe decisions and treats all tokens identically.

Configuration selection. Candidate layers are the five layers with the highest probe validation accuracy. For each γ, we select the candidate layer with the highest Completeness on the validation set, freeze this configuration, and evaluate it once on the test set.

Evaluation. The prompt, decoding settings, and evaluation protocol are identical to those of the main experiments, so that the steering baseline and our method differ only in the form of intervention.

Table 7: Comparison with CI-parametric activation steering trained on the original data of Wang et al. (2026). I, U, and C are reported in percent, and Cost is the average number of generated tokens. † marks degenerate Integrity caused by invalid or non-terminating generations.

Model	**Method**	$I\uparrow$	$U\uparrow$	$C\uparrow$	Cost$\downarrow$
Qwen2.5-7B	Masking (Ours)	72.6	53.4	**35.6**	150.8
	Steering ($\alpha = 0.5$)	31.5	61.6	19.2	160.1
	Steering ($\alpha = 1$)	37.0	63.0	24.7	155.0
Qwen2.5-14B	Masking (Ours)	72.6	48.0	**34.2**	161.1
	Steering ($\alpha = 0.5$)	37.0	53.4	17.8	163.3
	Steering ($\alpha = 1$)	31.5	61.6	17.8	167.2
Llama-3.1-8B	Masking (Ours)	65.8	49.3	**31.5**	217.4
	Steering ($\alpha = 0.5$)	100.0†	0.0	0.0	383.2
	Steering ($\alpha = 1$)	100.0†	0.0	0.0	2048.0
Mistral-7B	Masking (Ours)	50.7	61.6	**30.1**	241.4
	Steering ($\alpha = 0.5$)	100.0†	0.0	0.0	1775.7
	Steering ($\alpha = 1$)	100.0†	0.0	0.0	2030.6
Average	Masking (Ours)	65.4	53.1	**32.9**	192.7
	Steering ($\alpha = 0.5$)	67.1	28.8	9.2	620.6
	Steering ($\alpha = 1$)	67.1	31.2	10.6	1100.2

E.2 Activation Steering with Original Training Data

To control for the possibility that activation steering is disadvantaged by learning directions from our training split, we reproduce the CI-parametric activation-steering method of Wang et al. (2026) using its original training data. No examples from our training split are used to construct the steering directions. We evaluate $\alpha \in \{0.5, 1\}$ on four models using the same 73-example test set, prompts, and decoding configuration as our method.

On the two Qwen models, the best steering configurations obtain Completeness scores of 24.7% and 17.8%, compared with 35.6% and 34.2% for our method. On Llama and Mistral, both steering strengths produce degenerate behavior with zero Utility and Completeness. At $\alpha = 1$, all 73 Llama outputs and 69 of 73 Mistral outputs reach the 2,048-token limit, with no validly formatted answers; their apparent 100% Integrity therefore does not represent useful privacy protection.

Selecting the better α for each model gives steering an average Completeness of 10.6%, compared with 32.9% for masking. This 22.3-point gap indicates that global residual-stream steering transfers less reliably to span-specific contextual disclosure than localized attention masking. Because the steering directions are learned from an external dataset, this experiment should be interpreted as a transfer baseline rather than an in-domain upper bound.

E.3 Compatibility with CI-CoT Prompting

To examine whether probe-guided attention masking is complementary to prompting-based methods, we apply it on top of CI-CoT. Following the same probe-training and validation protocol, we select a layer-specific probe for CI-CoT-formatted inputs. The resulting configurations and held-out performance are reported in Table 8.

Table 9 reports the corresponding generation results. The combined method achieves an average integrity of 83.83% and completeness of 36.99%.

Compared with the original CI-CoT, adding attention masking improves C for all five models and increases its average from 24.66% to 36.99%, a gain of 12.33 percentage points. It also exceeds direct-prompt masking, which obtains an average C of 32.33%, by 4.66 percentage points and performs better on three of the five models. However, the composition increases the average generation cost to 523.4 tokens and reduces average U to 47.40%. These results demonstrate that masking is

Table 8: Test performance of linear probes for applying attention masking on top of CI-CoT. F1 treats restricted spans as the positive class. The probing layer and decision threshold τ are chosen on the validation set. All probes use $\lambda = 10^{-2}$.

Model	Layer	τ	Accuracy	F1
Qwen2.5-7B	22	0.5	90.7	90.7
Qwen2.5-14B	26	0.5	91.7	91.3
Llama-3.1-8B	24	0.5	88.4	87.9
Mistral-7B	18	0.5	88.7	88.2
Gemma-2-9B	24	0.7	90.5	90.0
Average	–	–	90.0	89.6

Table 9: Generation performance of probe-guided attention masking applied on top of CI-CoT. I, U, and C denote Integrity, Utility, and Completeness (%) following the synthetic dataset in Lan et al. (2025); Cost is the average number of generated tokens.

Model	$I\uparrow$	$U\uparrow$	$C\uparrow$	Cost$\downarrow$
Qwen2.5-7B	83.6	41.1	30.1	457.6
Qwen2.5-14B	84.9	54.8	42.5	515.6
Llama-3.1-8B	86.3	43.8	37.0	850.9
Mistral-7B	79.5	60.3	48.0	479.7
Gemma-2-9B	84.9	37.0	27.4	313.2
Average	**83.8**	**47.4**	**37.0**	**523.4**

complementary to CI-CoT, while the direct-prompt configuration provides the more efficient operating point.

E.4 Layer-wise Masking Coverage

Table 10: Effect of layer-wise masking coverage. **Probe Layer → Last** applies masking only from the selected probe layer $l^\star$ through the final layer, while **All-Layer Mask** applies it at every layer. I, U, and C are reported in percentage; Cost is the average number of generated tokens.

Model	Probe Layer → Last				All-Layer Mask			
	$I\uparrow$	$U\uparrow$	$C\uparrow$	Cost$\downarrow$	$I\uparrow$	$U\uparrow$	$C\uparrow$	Cost$\downarrow$
Qwen2.5-7B	68.5	53.4	31.5	150.7	72.6	53.4	**35.6**	150.8
Qwen2.5-14B	71.2	46.6	31.5	161.1	72.6	48.0	**34.2**	161.1
Llama-3.1-8B	63.0	53.4	**31.5**	219.9	65.8	49.3	**31.5**	217.4
Mistral-7B	49.3	65.8	**31.5**	238.6	50.7	61.6	30.1	241.4
Gemma-2-9B	76.7	35.6	26.0	116.9	78.1	39.7	**30.1**	117.1
Average	65.8	**51.0**	30.4	**177.4**	**67.9**	50.4	**32.3**	177.5

We compare our default all-layer masking with a partial variant that applies the attention mask only from the selected probe layer $l^\star$ through the final layer. This ablation tests whether the intervention must operate throughout the model or whether masking only the later layers is sufficient.

All-layer masking improves completeness on three models, matches partial masking on Llama-3.1-8B, and underperforms it on Mistral-7B. On average, it increases I by 2.2 points and C by 1.9 points, while decreasing U by only 0.6 points and leaving generation cost nearly unchanged. These results support all-layer masking as the default, although masking from the probe layer onward already captures most of the intervention benefit.