PRIVATE LONG-FORM GENERATION VIA SEMANTIC RELEASE

Anonymous authors
Paper under double-blind review

ABSTRACT

Reports based on private records often ask different questions about the same underlying quantities. Acquiring those quantities separately for each question divides the privacy budget across repeated computations. In this paper, we propose Semantic Release, which acquires shared task states once and reuses them throughout a report. Public question dependencies specify the required states. Record-level mechanisms release them into a typed ledger, a public compiler computes answers, and a writer expresses those answers without further private access. Conditional zCDP composition bounds the privacy cost of the resulting transcript by the acquisition budget. We evaluate a Gaussian-view instance on LongWeave Sales Report 8K. Under the same statistical scope and total budget, report correctness increases from 20.96% with question-wise acquisition to 24.74% with shared releases, while answered rate remains similar. A budget sweep improves both acquisition and report correctness. Exact-view controls and a fidelity analysis show where gains are lost: questions can require unreleased statistics, noisy views can produce wrong answers, and the writer can fail to express a correct answer.

1 INTRODUCTION

Language models use documents and structured records to write reports containing information absent from pretraining (Lewis et al., 2020; Izacard & Grave, 2021). A report may summarize a dataset, compare groups, and draw recommendations from the same findings. When the source records are private, both the report and any exposed intermediate results need privacy protection.

Inference-time differential privacy protects generation without retraining the model. Private prediction, residual-logit methods, and private token voting control record-dependent decoding decisions (Amin et al., 2024; Vinod et al., 2025; Koga et al., 2024). Their budget allocation follows the evolving text.

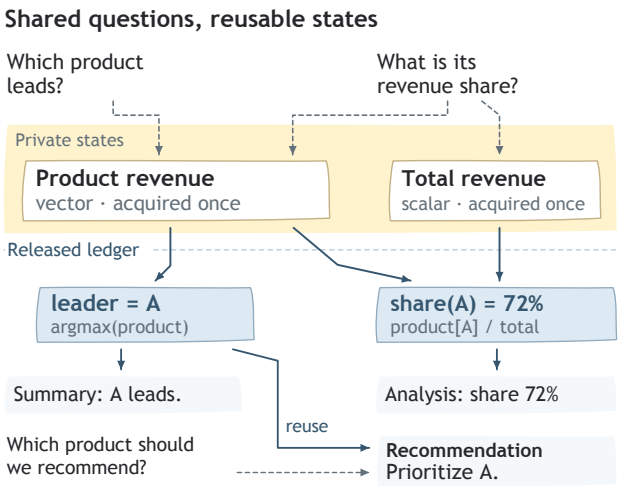


Figure 1: Shared task dependencies. Product revenue supports leader and share; total revenue completes the ratio. Each state is acquired once. Public answers are reused in the report. Values are illustrative.

Consider a sales report that names the leading product, calculates its revenue share, and discusses it in a recommendation. Retaining the product name helps the recommendation, but the share also requires the product's revenue and the total. A released product-revenue vector can support both the ranking and the ratio; a released total completes the latter. These quantities can remain useful as the report grows (Figure 1).

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.

In this paper, we propose Semantic Release to acquire the states specified by public question dependencies. Each state is released once into a ledger. A public compiler derives answers from these stored values, and the writer uses them throughout the report.

The design question is which states the privacy budget should acquire. A question-wise approach can obtain the information before writing, yet still reacquire the same statistic for different questions. Sharing a release concentrates that view's budget in one acquisition. Broader states may cover more questions but require greater sensitivity or a share of the budget.

We instantiate the method with Gaussian statistical views for complete sales reports. On 48 paired writer samples, shared releases improve report correctness from 20.96% to 24.74% over question-wise acquisition at the same scope and budget.

Our contributions are as follows.

1. We formulate private long-form generation around typed task states with public dependencies, so several questions and passages can use one release (Section 4).
2. We establish a transcript privacy guarantee under conditional zCDP composition and analyze how allocation and shared releases affect answer error (Section 5).
3. We evaluate complete reports against matched question-wise acquisition, with a budget sweep and separate measurements of scope, perturbation, and realization losses (Section 6).

2 PRELIMINARIES

Records and released history. Let $D = (d_1, \ldots, d_m)$ be a fixed private dataset with public record count m. Neighbors $D \sim D'$ differ by replacing one record. A record is a document in the evidence instance and a transaction row in the sales instance. Public input X contains the task, public candidate catalogs, and frozen model parameters θ. Retrieval or construction of the private dataset is outside this adjacency definition.

We call a released record-dependent quantity a task state and collect such states in a ledger $\mathcal{L}$. Before acquisition j, the released history is $H_j = (X, \mathcal{L}_{j-1}, Y_{<t_j})$, where t_j is the next writing position when the acquisition is requested. The mechanism releases $Z_j \sim M_j(D; H_j)$. Conditioning on this history allows later requests to use earlier releases and generated text.

Privacy accounting. A mechanism satisfies ρ-zero-concentrated differential privacy (zCDP) if, for every neighboring pair and every $\alpha > 1$,

$$D_\alpha(M(D) \| M(D')) \leq \alpha\rho. \tag{1}$$

Here D_α is Rényi divergence of order α. We use conditional composition and post-processing (Bun & Steinke, 2016).

Language, questions, and acquisitions. Let T count output tokens, N the public questions, and K the private acquisitions. A vector is one release even when it contains many coordinates and supports several questions. Its dimension and sensitivity still affect its utility.

3 MOTIVATION

We compare budget allocation over text positions with allocation over distinct task states.

Position-dependent allocation. Token-level mechanisms assign budget to record-dependent decoding decisions. A public distribution over candidate horizons H can define

$$w_t = \mathbb{E}_H\left[\frac{\mathbf{1}\{t \leq H\}}{H}\right], \qquad \rho_t = \rho_\star w_t, \qquad \sum_{t \geq 1} w_t = 1.$$

Here H is a positive, almost surely finite integer. A point mass gives uniform allocation over one horizon; a mixture can reserve budget for later positions. We use this construction as the Scale-Mixed

Finite-Influence Decoding (SM-FID) baseline. State acquisition instead assigns ρ_j to each distinct requested state, with $\sum_j \rho_j \le \rho_\star$. Appendix E compares these allocation choices at three evidence positions.

Choosing the state. Grouping statistics into fewer releases does not by itself reduce their noise. Suppose r views each have sensitivity bounded by Δ. A joint isotropic Gaussian release using the bound $\sqrt{r}\Delta$ and total budget ρ has per-coordinate variance

$$\sigma^2_{\text{joint}} = \frac{r\Delta^2}{2\rho} = \frac{\Delta^2}{2(\rho/r)} = \sigma^2_{\text{split}}. \tag{2}$$

Under these bounds, separate releases with equal budget shares have the same variance. The useful structure is in the workload: a product-revenue vector serves several rankings and ratios, while a city comparison requires a different view. Section 6.2 tests this reuse under fixed scope and budget.

4 METHOD

Semantic Release receives public task X, a fixed record set D, a public request catalog, and total budget $\rho_\star$. Figure 2 shows the sales-report instance.

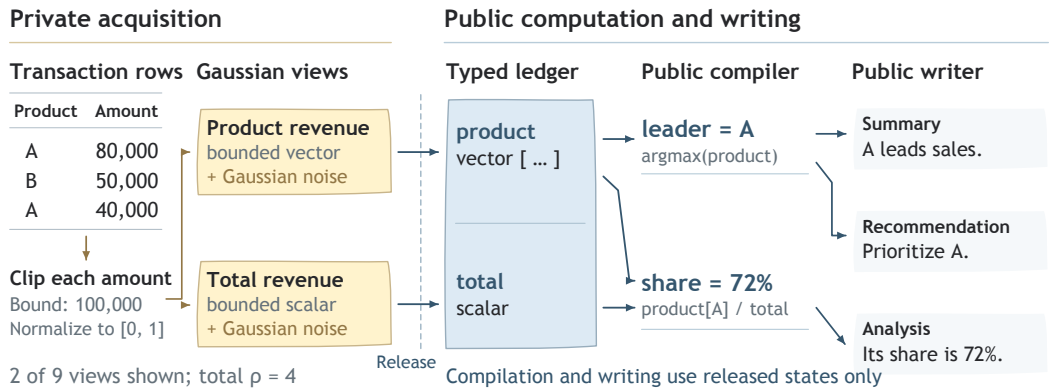


Figure 2: Gaussian sales-report instance. Clipped aggregates receive Gaussian noise before entering the typed ledger. A public compiler derives answers that the writer reuses across passages. Two of nine views are shown at total $\rho_\star = 4$, with illustrative records and outputs.

4.1 TYPED REQUESTS

A request specifies a type τ_j, a scope, and a normalized key k_j. For product revenue, the type is a vector over the public product catalog and the scope specifies the population and revenue field. Requests for this same quantity use the same key. Types can also be scalar aggregates or categorical evidence states. Keys and scopes are determined from public input or released history.

Each public question q_ℓ has a dependency set B_ℓ of required states and a public answer function g_ℓ. Once the states are available, the compiler computes

$$\widehat{a}_\ell = g_\ell\big(X, \{Z_j : j \in B_\ell\}\big). \tag{3}$$

If a required view is absent, the compiler returns unresolved. Noisy values can also lead to an unresolved answer under its fixed public abstention rules.

Scope and identity. A key identifies the quantity together with the population over which it is computed. Total product revenue for one region and total product revenue for all regions therefore require different keys, even if their coordinate labels coincide. Conversely, a question about the leading product and a question about that product's share can use the same product-revenue key. The dependency set records this relationship before the compiler evaluates either answer. This makes reuse a property of the requested information rather than a match between the wording of questions.

A change in scope can also change the contribution bound. A new request must therefore carry its own sensitivity calibration and allocation, while an existing key retains the value already released. The fixed-dataset setting keeps this interpretation stable throughout a report. Applying the interface to a changing dataset would additionally require an explicit version in the state identity and accounting for releases from each version.

4.2 Bounded acquisition

For a new key and fixed history H_j, the request defines record-local contributions and an aggregate,

$$S_j(D; H_j) = \sum_{i=1}^{m} \phi_j(d_i; H_j), \qquad \sup_{D \sim D'} \|S_j(D; H_j) - S_j(D'; H_j)\|_2 \le \Delta_j. \tag{4}$$

Replacing one record changes only its contribution. Before accessing records, the controller fixes the request and reserves its budget ρ_j. It then releases

$$Z_j \sim M_j(S_j; H_j, \rho_j). \tag{5}$$

Each conditional mechanism must satisfy ρ_j-zCDP. Record-local scoring is included in ϕ_j, and the conditional privacy bound covers the full acquisition mechanism M_j.

4.3 Ledger and writer

The controller stores each release at the public boundary, $\mathcal{L}[k_j] = (\tau_j, \mathrm{scope}_j, Z_j, \rho_j)$. When another question requests the same key, the controller returns the stored value rather than sampling again. The stored value may be an abstention $\perp$, which supplies no definite answer.

Let $g(X, \mathcal{L})$ collect the answers from Eq. 3 and their unresolved markers. The writer then samples

$$Y_t \sim p_\theta\big(\cdot \mid X, \mathrm{Serialize}(g(X, \mathcal{L}_{a(t)})), Y_{<t}\big), \tag{6}$$

where $a(t)$ counts completed acquisitions. The writer has access only to public inputs and compiled answers. It is prompted to preserve their values and uncertainty.

4.4 Shared statistical views

The sales instance clips each transaction amount r_i at a public bound B_a and normalizes it to $x_i = \mathrm{clip}(r_i, 0, B_a)/B_a \in [0, 1]$. A global revenue contribution is x_i, and a product-revenue contribution is $x_i e_{p_i}$, where e_{p_i} is the one-hot vector for product p_i in the public catalog.

The global sum has replacement sensitivity 1. For a histogram, a replacement changes $x e_p$ into $x' e_q$; its Euclidean norm is at most $\sqrt{2}$. We release

$$Z_j = S_j + \mathcal{N}(0, \sigma_j^2 I), \qquad \sigma_j^2 = \frac{\Delta_j^2}{2\rho_j}. \tag{7}$$

This mechanism satisfies ρ_j-zCDP for each view.

The sales instance acquires nine shared statistical views before writing. Appendix B.1 lists the active views and their privacy allocations.

Compiling a shared comparison. Consider a report that asks for the leading product, its revenue share, and the gap to the runner-up. From one released product vector Z, the compiler selects the largest and second-largest coordinates. It uses their difference for the gap and, when the declared scope permits, divides the leading coordinate by the sum of the vector for the share. All three answers then refer to the same perturbed quantities. Their dependencies include the product catalog and population scope, so a share within one region is not silently interpreted as a share of company-wide revenue.

This example also separates two sources of disagreement. Independent acquisitions can select different leaders because each draw perturbs the ordering again. Shared acquisition fixes the ordering available to the compiler, but the writer can still omit the runner-up or describe the gap incorrectly. The ledger addresses the former source; the realization analysis measures the latter. If the requested share uses a denominator outside the released scope, the compiler leaves that answer unresolved even when the ranking is available.

4.5 CATEGORICAL EVIDENCE STATES

Some tasks require a discrete judgment, such as whether a set of documents supports a proposition. A fixed reader maps each record to $v_{i,j} \in [-R_j, R_j]^p$, and the mechanism averages these contributions as $u_j = m^{-1} \sum_i v_{i,j}$. For output space $\mathcal{A}_j$, a logit map F_j and public bias b_j^{pub} define

$$\Pr[Z_j = z \mid D, H_j] = \frac{\exp b_j(z)}{\sum_{a \in \mathcal{A}_j} \exp b_j(a)}, \qquad b_j = F_j(u_j) + b_j^{\mathrm{pub}}. \tag{8}$$

For supported, refuted, and unresolved outputs $\mathcal{A}_j = \{+, -, \perp\}$, a map F_j that is 1-Lipschitz in ℓ_∞ and radius $R_j = m\sqrt{\rho_j/2}$ give ρ_j-zCDP (Cesar & Rogers, 2021). Appendix A.2 supplies a concrete map and its calibration.

5 PRIVACY AND UTILITY ANALYSIS

Transcript guarantee. **Theorem 1.** Fix public budgets $\rho_1, \ldots, \rho_{K_{\max}}$ with $\sum_j \rho_j \le \rho_\star$. Suppose that, for every possible released history, acquisition j is conditionally ρ_j-zCDP. Requests, question answers, writer tokens, and stopping decisions depend only on public inputs and released history. Then the complete transcript $(Z_1, \ldots, Z_K, Y_{1:T})$, for any finite output length, satisfies $\rho_\star$-zCDP under record replacement.

Proof. Pad unused acquisitions with data-independent constants. Conditional zCDP composition bounds the divergence of the acquisition transcript by $\alpha \sum_j \rho_j$ (Bun & Steinke, 2016). Interleaved writer tokens and public control decisions have identical conditional kernels on neighboring datasets once the released history is fixed, and therefore add no privacy-loss factors. Post-processing preserves the bound for the ledger, answers, and document. Appendix A gives the likelihood-ratio argument and a privacy-filter extension to predictable budgets. □

Budget and release count. For uniform allocation, $\rho_j = \rho_\star/K$. In the Gaussian instance, Eq. 7 then gives $\sigma_j^2 = K\Delta_j^2/(2\rho_\star)$. Acquiring more states therefore increases each view's noise under a fixed budget and sensitivity.

Shared-state error. For a linear answer $a_\ell = \sum_j w_{\ell j}^\top S_j$, let $w_{\ell j}$ be the public coefficient vector on view j. With independent Gaussian releases, the compiled answer $\widehat{a}_\ell = \sum_j w_{\ell j}^\top Z_j$ has perturbation variance

$$\mathrm{Var}(\widehat{a}_\ell - a_\ell) = \sum_j \frac{\Delta_j^2 \|w_{\ell j}\|_2^2}{2\rho_j}. \tag{9}$$

Dividing ρ_j among n_j independent acquisitions gives each copy coordinate variance $n_j\Delta_j^2/(2\rho_j)$. A shared release has n_j-fold lower marginal variance, but its noise is common to all dependent answers.

For two linear answers, sharing also determines their perturbation covariance:

$$\mathrm{Cov}(\widehat{a}_\ell - a_\ell, \widehat{a}_h - a_h) = \sum_j \frac{\Delta_j^2 w_{\ell j}^\top w_{hj}}{2\rho_j}. \tag{10}$$

Questions with aligned coefficients inherit positively correlated errors; disjoint views contribute no covariance. Thus a report can repeat the same mistaken comparison consistently. Consistency across passages alone cannot establish correctness, and treating dependent answers as independent observations can understate uncertainty. Our report-level bootstrap keeps questions from the same report together.

For a weighted linear objective, write $A_j = \sum_\ell \omega_\ell \|w_{\ell j}\|_2^2$, where $\omega_\ell \ge 0$ weights question ℓ. Lagrange multipliers give $\rho_j \propto \Delta_j \sqrt{A_j}$ for minimizing $\sum_j \Delta_j^2 A_j/(2\rho_j)$ under the total budget, on positive-weight views. For the sales compiler's nonlinear rankings and ratios, allocations are selected on development data.

From compiled answers to reports. Report correctness depends on both the compiled answer and the writer's use of it. Fix a question and a deterministic answer-level correctness rule, applied to both the compiled answer and its expression in the report. Let ϵ_A and ϵ_E be their error rates, and let ϵ_F be the probability that the two answers differ under a common representation. Then

$$|\epsilon_E - \epsilon_A| \leq \epsilon_F. \tag{11}$$

Their correctness labels can differ only when the answers differ. Faithful repetition of a wrong compiled answer can still produce several wrong passages.

Nonlinear answer sensitivity. The variance calculation describes linear answers directly. Rankings depend instead on the gap between competing coordinates. If two coordinates in one released view have independent noise with variance σ^2, their noisy difference has variance $2\sigma^2$. A small true gap is consequently more susceptible to an order reversal than a large one under the same allocation. Ratios introduce a different issue: a perturbed denominator near zero can make an otherwise moderate numerator error produce a large change in the answer. Public abstention rules can leave such cases unresolved. These operations explain why improved view accuracy need not translate into the same change in answered rate or report correctness.

Allocation and coverage. The linear allocation objective weights views by their contribution to the questions that use them. A view used by many high-weight questions can receive more budget, while a view with no weight need not be acquired for that objective. This optimization presumes that the dependency sets and weights are fixed. Selecting which views to include adds a discrete coverage decision: removing one view may make every question requiring it unresolved, even if their other dependencies are accurate. Consequently, optimizing noise within a selected scope and choosing the scope solve different parts of the workload problem. The exact-scope controls in the experiments measure the consequence of that selection before attributing the remaining loss to acquisition noise.

The shared-noise structure also affects aggregate report judgments. Several incorrect statements may trace back to one perturbed coordinate, so counting each statement as a separate acquisition failure obscures the responsible state. Conversely, correcting the expression of one answer need not improve other answers derived from that state. The dependency map provides a way to connect answer errors to their common inputs. In the present evaluation, the compiled-answer and report measurements expose these two levels separately. A more detailed diagnostic could group failures by released key and examine whether the same ordering error propagates across rankings, shares, and narrative comparisons. Such an analysis would locate concentrated failures without changing the privacy mechanism or making another query to the private records.

6 Experiments

6.1 Setup

Task and model. LongWeave SR8K requests reports of 8192 *words* from synthetic sales records (Xiao et al., 2025). We use Qwen3.5-9B as the writer (Qwen Team, 2026), treating each transaction as a private record with replacement adjacency. The default total budget is $\rho_\star = 4$; views and allocations are selected on development data and fixed before evaluation.

We evaluate acquisition on two disjoint held-out cohorts of 32 tables each and compare five report conditions on the same 48 writer samples. Appendix D gives the cohort split and generation settings.

Conditions. Public-only receives the task and public questions. Semantic Release additionally receives answers compiled from the shared private ledger. Question-wise acquisition uses independently reacquired views for each question under the same total budget (Section 6.2). Oracle-active receives a clipped, exact ledger restricted to the same active views, isolating perturbation within that scope. Non-private receives the complete raw CSV and official questions. All conditions use the same writer configuration.

Evaluation. Reports are assessed against LongWeave's official questions and target answers. Correctness and answered rate both divide by all official questions. Unless stated otherwise, report results refer to the main evaluation; a local evaluation provides a second rating and the benchmark harmonic

score. Metrics are averaged over samples, with paired sample-bootstrap intervals. Appendix D gives both protocols.

6.2 COMPLETE REPORTS

Matched question-wise acquisition. The baseline uses the same views, clipping, question dependencies, compiler, writer, and evaluation as Semantic Release. Its only change is to reacquire a view for each question that needs it. If view j serves n_j distinct question dependencies, each independent acquisition receives

$$\rho_{j,u} = \rho_j / n_j, \qquad u = 1, \ldots, n_j. \tag{12}$$

Both methods therefore spend $\sum_j \rho_j = 4$ on the same statistical scope. Semantic Release spends ρ_j once and shares that released value across the dependencies.

	Main		Local	
Condition	Correct (%)	Answered (%)	Score	Correct (%)
Public-only	5.96	55.86	0.273	20.68
Question-wise acquisition	20.96	55.31	0.318	25.91
Semantic Release	24.74	54.43	0.336	27.88
Oracle-active	41.16	58.06	0.441	41.37
Non-private	26.97	84.61	0.546	41.50

Table 1: Complete-report results on 48 paired samples. Private conditions use $\rho_\star = 4$; Oracle-active and Non-private are non-private controls. Entries are report means.

Report correctness. Sharing releases raises report correctness from 20.96% to 24.74% (Table 1). The paired gain is 3.78 points, with 95% interval $[0.82, 6.76]$. Answered rate falls by 0.88 points, so the gain comes from more correct realized answers rather than more questions being attempted. The local evaluation gives a smaller mean correctness gain of 1.97 points, with interval $[-0.62, 4.53]$.

Both acquisition interfaces improve substantially over Public-only. For Semantic Release, the correctness gain is 18.78 points, with interval $[16.51, 21.08]$; the local evaluation gives a 7.20-point gain, with interval $[1.82, 12.39]$. The question-wise condition already captures much of this benefit.

Oracle-active reaches 41.16% correctness, 16.42 points above Semantic Release, showing the remaining value of accurate answers within the active scope. Non-private answers more questions (84.61%) but reaches 26.97% correctness under this evaluator. The local evaluation rates Oracle-active and Non-private at 41.37% and 41.50%, respectively.

Semantic Release reports average 7346 words. Table 8 gives the achieved word and token lengths.

6.3 COMPILED-ANSWER QUALITY

The schema-supported set for Table 2 contains question types recognized by the public compiler, including those whose required views are not released. Coverage is the fraction resolved, precision is correctness among resolved answers, and correct coverage is correctness over the schema-supported set. All-question recall also includes questions outside these types.

Metric (%)	A	B
Coverage	60.22	59.98
Precision	70.32	67.80
Correct coverage	42.35	40.67
Median monetary error	5.45	6.62
All-question recall	29.81	28.37

Table 2: Compiled answers on two held-out cohorts at $\rho_\star = 4$. Monetary error is relative error.

Compared with question-wise acquisition, sharing raises correct coverage from 34.33% to 42.35% in Cohort A and from 32.82% to 40.67% in Cohort B. Resolved coverage differs by less than one point in either cohort; the gains come from higher precision. Cohort B precision is 67.80%, with bootstrap interval $[62.33, 72.88]$. Across both cohorts, the compiler answers 902 of 3101 official questions correctly before writing. Appendix D.1 gives the full matched comparison.

6.4 PRIVACY–UTILITY SWEEP

We vary $\rho_\star \in \{1, 2, 4, 8\}$ while holding the active views, allocation proportions, and downstream pipeline fixed.

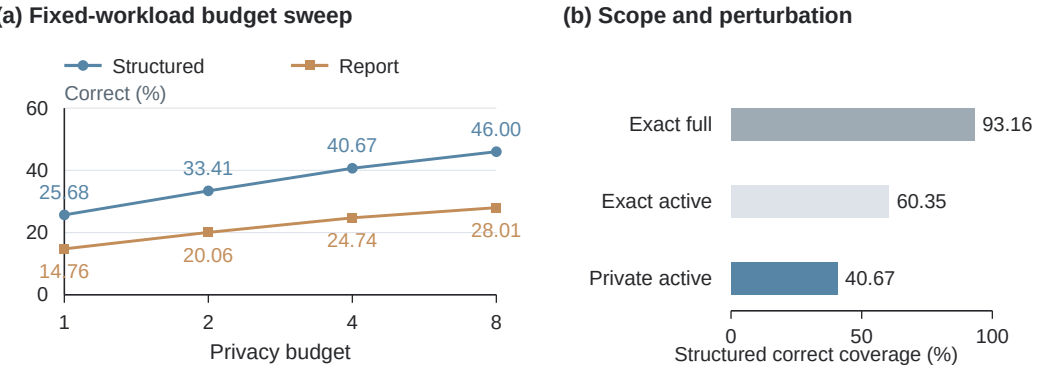


Figure 3: Utility and its limits. (a) Fixed-workload budget sweep: Cohort B structured coverage and report correctness on 48 reports, with successive budget doublings. (b) Cohort B structured coverage separates scope restriction from perturbation at $\rho_\star = 4$.

As the budget rises from 1 to 8, structured correct coverage increases from 25.68% to 46.00%, and report correctness from 14.76% to 28.01% (Figure 3(a)). Answered rate changes only from 51.91% to 54.12%.

The observed report gain is smaller at the upper step: $2 \to 4$ adds 4.68 correctness points, with 95% interval $[1.22, 8.04]$, while $4 \to 8$ adds 3.27 points, with interval $[-0.12, 6.57]$. Locally evaluated correctness changes from 27.88% to 27.61% at the upper step. The acquisition curve continues to rise at the upper budget, while report-level gains are less consistent.

6.5 RELEASE SCOPE AND PERTURBATION

Restricting exact statistics to the active views costs 32.81 points of correct coverage, compared with 19.69 points lost to perturbation within that scope (Figure 3(b)). Even at $\rho_\star = 8$, private coverage remains 14.35 points below exact active scope. More budget improves active-view accuracy but cannot supply missing dependencies. Broader scope can recover them, at the cost of changed sensitivity or budget divided across more acquisitions.

6.6 REALIZATION FIDELITY

On Cohort B, the local evaluator marks 63.41% of questions with correct compiled answers as correct in the report, versus 34.45% whose compiled answers are wrong. Omissions, incomplete answers, and contradictions contribute to the realization error in Eq. 11.

Interpreting the controls together. The controls distinguish interventions at different stages of the report pipeline. Exact full scope asks what the compiler can answer when its statistical dependencies are available without perturbation. Exact active scope retains the chosen view set and removes only its noise. The private active condition then measures performance after acquisition under the privacy budget. A failure that persists with exact active views points to an unavailable dependency or a compiler limitation; increasing the budget on those same views cannot resolve it. A failure introduced by perturbation instead motivates a different allocation or measurement strategy. Finally, a correct compiled answer that is omitted or altered in the report calls for changes to realization. These distinctions give separate targets for improving coverage, acquisition, and writing.

The matched reuse comparison addresses a narrower question within this pipeline. Both methods acquire the same kinds of statistics, so their difference reflects how the available budget is distributed across repeated demands. It does not estimate the benefit of selecting a broader catalog. Likewise, the budget sweep preserves the workload and tests the effect of more accurate releases within it. Reading

these comparisons together identifies which intervention was actually varied and prevents a gain at one stage from being attributed to a change at another.

7 Related Work

Private prediction and retrieval-augmented generation. Retrieval-augmented generation conditions a language model on retrieved evidence, allowing source information to influence an answer without incorporating it into the model's parameters (Lewis et al., 2020; Izacard & Grave, 2021). With private sources, this conditioning creates a record-dependent output channel. Private prediction reserves private access for decisions insufficiently explained by a public model (Amin et al., 2024). InvisibleInk bounds reference-induced logit residuals (Vinod et al., 2025); DP-RAG uses private token votes and sparse-vector tests (Koga et al., 2024). These methods place the private interface close to decoding, where the current question and generated prefix determine the next decision. A report can revisit the same quantity through different prefixes, so the unit of decoding need not coincide with the unit of information being requested.

Reusable private representations. PATE aggregates private predictions for downstream learning (Papernot et al., 2018), separating access to private teachers from subsequent use of a student. DP-KSA releases keywords to condition an answer (Tang et al., 2026). EPSVec sanitizes dataset vectors once for later synthetic generation (Banayeeanzade et al., 2026). These approaches illustrate several choices of reusable object: a learned predictor, a compact textual representation, or a dataset representation. The choice determines what later computations can recover. For a report with explicit questions, the dependencies themselves provide another way to specify that object. A product-revenue vector, for example, has an identifiable population and coordinate system, and its entries can support multiple answer functions without another private query.

Statistical workloads. The matrix mechanism and HDMM optimize measurements for linear-query workloads (Li et al., 2015; McKenna et al., 2018). Their workload perspective makes the relationship among requested statistics central to measurement design. Our sales instance connects such measurements to a report: released views feed public computations for totals, rankings, and ratios, which then become inputs to a language model. A ranking and a ratio can depend on the same linear measurements even though neither final answer is itself a linear query. This distinction separates measurement quality from the quality of the derived answer. Alternative measurement strategies can be used within the acquisition layer, provided that the reconstructed views retain their stated types and conditional privacy guarantees.

Adaptive privacy accounting. Concentrated differential privacy provides a composition framework for repeated acquisitions (Bun & Steinke, 2016). Privacy filters and odometers address accounting when the sequence of mechanisms is chosen during execution (Rogers et al., 2016); subsequent work develops Rényi filters and fully adaptive composition (Feldman & Zrnic, 2021; Whitehouse et al., 2023). In a report workflow, earlier releases may influence which state is requested next or whether acquisition stops. The relevant accounting unit remains a new record-dependent release. Looking up an existing value and applying a public answer function are post-processing. Our main theorem uses fixed public allocations; Appendix A describes the extension to predictable allocations controlled by a filter.

Semantic Release combines a workload of typed states with a public answer compiler and a report writer. Its contribution is the connection between the information shared by report questions and the objects that consume privacy budget. The matched acquisition experiment isolates this connection by holding statistical scope and total budget fixed. The exact-view and realization analyses then distinguish the accuracy of those objects from the writer's ability to use them.

8 Conclusion

In this paper, we propose Semantic Release for private reports with overlapping information demands. In the sales workload, sharing task-state releases improves correctness over matched question-wise acquisition. At the default budget, scope restriction costs more correct coverage than perturbation.

AI USE STATEMENT

AI assistants were used for manuscript drafting and structural revision, language editing, mathematical exposition and consistency checks, figure and table code, and typesetting. Language models also serve as the report writer and evaluators in the experiments, as described in Section 6 and Appendix D. The authors take full responsibility for the final content of this work.

REFERENCES

Kareem Amin, Alex Bie, Weiwei Kong, Alexey Kurakin, Natalia Ponomareva, Umar Syed, Andreas Terzis, and Sergei Vassilvitskii. Private prediction for large-scale synthetic text generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, volume EMNLP 2024 of *Findings of ACL*, pp. 7244–7262. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-EMNLP.425. URL https://doi.org/10.18653/v1/2024.findings-emnlp.425.

Amin Banayeeanzade, Qingchuan Yang, Deqing Fu, Spencer Hong, Erin Babinsky, Alfy Samuel, Anoop Kumar, Robin Jia, and Sai Praneeth Karimireddy. EPSVec: Efficient and private synthetic data generation via dataset vectors. *CoRR*, abs/2602.21218, 2026. doi: 10.48550/ARXIV.2602.21218. URL https://doi.org/10.48550/arXiv.2602.21218.

Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In Martin Hirt and Adam D. Smith (eds.), *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part I*, volume 9985 of *Lecture Notes in Computer Science*, pp. 635–658, 2016. doi: 10.1007/978-3-662-53641-4_24. URL https://doi.org/10.1007/978-3-662-53641-4_24.

Mark Cesar and Ryan Rogers. Bounding, concentrating, and truncating: Unifying privacy loss composition for data analytics. In Vitaly Feldman, Katrina Ligett, and Sivan Sabato (eds.), *Algorithmic Learning Theory, 16-19 March 2021, Virtual Conference, Worldwide*, volume 132 of *Proceedings of Machine Learning Research*, pp. 421–457. PMLR, 2021. URL http://proceedings.mlr.press/v132/cesar21a.html.

Vitaly Feldman and Tijana Zrnic. Individual privacy accounting via a rényi filter. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 28080–28091, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/ec7f346604f518906d35ef0492709f78-Abstract.html.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In Paola Merlo, Jörg Tiedemann, and Reut Tsarfaty (eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pp. 874–880. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EACL-MAIN.74. URL https://doi.org/10.18653/v1/2021.eacl-main.74.

Tatsuki Koga, Ruihan Wu, and Kamalika Chaudhuri. Privacy-preserving retrieval augmented generation with differential privacy. *CoRR*, abs/2412.04697, 2024. doi: 10.48550/ARXIV.2412.04697. URL https://doi.org/10.48550/arXiv.2412.04697.

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. Retrieval-augmented generation for knowledge-intensive NLP tasks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Chao Li, Gerome Miklau, Michael Hay, Andrew McGregor, and Vibhor Rastogi. The matrix mechanism: optimizing linear counting queries under differential privacy. *VLDB J.*, 24(6): 757–781, 2015. doi: 10.1007/S00778-015-0398-X. URL https://doi.org/10.1007/s00778-015-0398-x.

Ryan McKenna, Gerome Miklau, Michael Hay, and Ashwin Machanavajjhala. Optimizing error of high-dimensional statistical queries under differential privacy. *Proc. VLDB Endow.*, 11(10): 1206–1219, 2018. doi: 10.14778/3231751.3231769. URL http://www.vldb.org/pvldb/vol11/p1206-mckenna.pdf.

Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. Scalable private learning with PATE. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=rkZB1XbRZ.

Qwen Team. Qwen3.5-9B model card, 2026. URL https://huggingface.co/Qwen/Qwen3.5-9B.

Ryan M. Rogers, Salil P. Vadhan, Aaron Roth, and Jonathan R. Ullman. Privacy odometers and filters: Pay-as-you-go composition. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 1921–1929, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/58c54802a9fb9526cd0923353a34a7ae-Abstract.html.

Tingting Tang, James Flemings, Yongqin Wang, and Murali Annavaram. Differentially private retrieval-augmented generation. *CoRR*, abs/2602.14374, 2026. doi: 10.48550/ARXIV.2602.14374. URL https://doi.org/10.48550/arXiv.2602.14374.

Vishnu Vinod, Krishna Pillutla, and Abhradeep Guha Thakurta. InvisibleInk: High-utility and low-cost text generation with differential privacy. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/61432775ec681f7ac6e7d335fa933e4f-Abstract-Conference.html.

Justin Whitehouse, Aaditya Ramdas, Ryan Rogers, and Steven Wu. Fully-adaptive composition in differential privacy. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 36990–37007. PMLR, 2023. URL https://proceedings.mlr.press/v202/whitehouse23a.html.

Zikai Xiao, Fei Huang, Jianhong Tu, Jianhui Wei, Wen Ma, Yuxuan Zhou, Jian Wu, Bowen Yu, Zuozhu Liu, and Junyang Lin. LongWeave: A long-form generation benchmark bridging real-world relevance and verifiability. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025, Suzhou, China, November 4-9, 2025*, pp. 10386–10417. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.FINDINGS-EMNLP.549. URL https://doi.org/10.18653/v1/2025.findings-emnlp.549.

A Transcript Privacy

Fix neighboring datasets D, D' and a Rényi order $\alpha > 1$. Order the transcript by private acquisitions and pad unused slots up to the public bound $K_{\max}$ with data-independent constants. Let $p_j(\cdot \mid h)$ and $q_j(\cdot \mid h)$ be the conditional acquisition laws. Finite conditional Rényi divergence in both directions gives common support on reachable histories. Every interleaved writer or public-control transition

has the same conditional law under D and D' at a fixed history. These factors cancel in the transcript likelihood ratio, leaving

$$\log \frac{dP_{\mathsf{Tr}}}{dQ_{\mathsf{Tr}}} = \sum_{j=1}^{K_{\max}} \ell_j, \qquad \ell_j = \log \frac{p_j(Z_j \mid H_j)}{q_j(Z_j \mid H_j)}. \tag{13}$$

Conditional zCDP implies

$$\mathbb{E}_D[e^{(\alpha-1)\ell_j} \mid H_j] = \exp\big((\alpha-1)D_\alpha(p_j\|q_j)\big) \le e^{\alpha(\alpha-1)\rho_j}. \tag{14}$$

The previous likelihood increments are measurable given H_j. Iterated conditional expectations therefore yield

$$\mathbb{E}_D \exp\left((\alpha-1)\sum_j \ell_j\right) \le \exp\left(\alpha(\alpha-1)\sum_j \rho_j\right). \tag{15}$$

Taking the logarithm and dividing by $\alpha - 1$ gives $D_\alpha(P_{\mathsf{Tr}}\|Q_{\mathsf{Tr}}) \le \alpha\rho_\star$. Swapping D, D' gives the reverse direction. This proves Theorem 1 for every finite language horizon. Further writing from the same ledger, including multiple reports, adds only identical conditional kernels and preserves the bound.

Predictable budgets. We apply the privacy-filter approach to adaptive composition (Rogers et al., 2016; Feldman & Zrnic, 2021; Whitehouse et al., 2023) to the acquisition transcript. For history-dependent budgets $\rho_j(H_j)$, require a public filter to enforce $\sum_{j\le K}\rho_j(H_j) \le \rho_\star$ on every possible path, with $K \le K_{\max}$. Set $W_0 = 1$ and

$$W_n = \exp\left((\alpha-1)\sum_{j\le n}\ell_j - \alpha(\alpha-1)\sum_{j\le n}\rho_j(H_j)\right). \tag{16}$$

Since W_{n-1} and $\rho_n(H_n)$ are known before acquisition n,

$$\mathbb{E}_D[W_n \mid H_n] = W_{n-1}e^{-\alpha(\alpha-1)\rho_n(H_n)}\mathbb{E}_D[e^{(\alpha-1)\ell_n} \mid H_n] \le W_{n-1}. \tag{17}$$

Thus W_n is a nonnegative supermartingale under the acquisition filtration. Bounded stopping gives $\mathbb{E}_D W_K \le 1$. The pathwise budget bound then implies

$$\mathbb{E}_D e^{(\alpha-1)\sum_{j\le K}\ell_j} \le e^{\alpha(\alpha-1)\rho_\star}\mathbb{E}_D W_K \le e^{\alpha(\alpha-1)\rho_\star}, \tag{18}$$

which proves the same fixed transcript guarantee.

A.1 Gaussian acquisition

For a fixed history, let $P = \mathcal{N}(S_j(D), \sigma_j^2 I)$ and $Q = \mathcal{N}(S_j(D'), \sigma_j^2 I)$. Completing the square in $\int p^\alpha q^{1-\alpha}$ gives

$$D_\alpha(P\|Q) = \frac{\alpha\|S_j(D) - S_j(D')\|_2^2}{2\sigma_j^2} \le \frac{\alpha\Delta_j^2}{2\sigma_j^2} = \alpha\rho_j. \tag{19}$$

This holds for every $\alpha > 1$ and either neighboring order (Bun & Steinke, 2016). The nine budgets in Appendix B.1 sum to four, so their joint release and all subsequent report generation satisfy 4-zCDP under ideal Gaussian sampling.

A.2 Calibration of a bounded softmax instance

For the mechanism in Eq. 8, suppose record vectors lie in $[-R_j, R_j]^p$, $u_j = m^{-1}\sum_i v_{i,j}$, and logits are $F_j(u_j) + b^{\mathrm{pub}}$. If F_j has ℓ_∞ Lipschitz constant L_j, one replacement gives a logit change bounded by $2L_jR_j/m$. Write P, Q for the neighboring softmax laws and $A(b) = \log\sum_z e^{b(z)}$. Their privacy loss is

$$L(z) = b(z) - b'(z) - [A(b) - A(b')]. \tag{20}$$

The normalizer difference is constant across outputs. Hence

$$\max_z L(z) - \min_z L(z) \le 2\|b - b'\|_\infty \le \eta_j, \qquad \eta_j = 4L_jR_j/m. \tag{21}$$

A privacy-loss range of length η_j implies $(\eta_j^2/8)$-zCDP as follows (Cesar & Rogers, 2021). Let $\mu = \mathbb{E}_P L$. Hoeffding's lemma gives $\log \mathbb{E}_P e^{t(L-\mu)} \le t^2\eta_j^2/8$ for all real t. Since $\mathbb{E}_P e^{-L} = 1$, setting $t = -1$ gives $\mu \le \eta_j^2/8$. Setting $t = \alpha - 1 > 0$ then yields

$$D_\alpha(P\|Q) = \frac{\log \mathbb{E}_P e^{(\alpha-1)L}}{\alpha - 1} \le \mu + \frac{(\alpha-1)\eta_j^2}{8} \le \frac{\alpha\eta_j^2}{8}. \tag{22}$$

The same argument holds in reverse. Thus a sufficient radius is

$$R_j = \frac{m}{L_j}\sqrt{\rho_j/2}. \tag{23}$$

For $L_j > 0$, this choice ensures $\eta_j^2/8 \le \rho_j$. A constant logit map ($L_j = 0$) is data-independent and has zero privacy cost.

For example, a concrete three-logit map is

$$F(u) = \big((u_+ - u_-)/2,\ (u_- - u_+)/2,\ 0\big). \tag{24}$$

For $\delta = u - u'$, each nonconstant coordinate changes by at most $(|\delta_+| + |\delta_-|)/2 \le \|\delta\|_\infty$, proving $L = 1$. Removing the factors $1/2$ gives $L = 2$. Here u_+ and u_- aggregate bounded support and refutation scores. Dividing aggregate logits by temperature τ multiplies the Lipschitz bound by $1/\tau$; a hard aggregate threshold generally has no finite global Lipschitz constant. Either operation must be analyzed within the full channel. Record-local transformations followed by bounded clipping are instead covered by the contribution bound.

If a block combines h of K values, proportional budgeting gives

$$\rho_h = \rho_\star h/K, \qquad R_h = \frac{m}{L_h}\sqrt{\rho_h/2} = \frac{m}{L_h}\sqrt{\rho_\star h/(2K)}. \tag{25}$$

The factor $1/K$ enters through the block budget ρ_h. At $m = 8$, $\rho_j = 4$, and $L_j = 1$, it gives $R_j = 8\sqrt{2} \simeq 11.31$ and privacy-loss range bound $4\sqrt{2} \simeq 5.66$. Because $\mathbb{E}_P e^{-L} = 1$, the range of L contains zero, so $|L| \le \eta_j$. The same range thus gives a sufficient pure-DP bound of 5.66 for this softmax mechanism.

A.3 Budget interpretation

The Gaussian schedule's total $\rho_\star = 4$ gives the standard sufficient conversion

$$\varepsilon = \rho_\star + 2\sqrt{\rho_\star \log(1/\delta)}. \tag{26}$$

At $\delta = 10^{-5}$, this is approximately 17.57. The sales experiments use this development-selected operating point with transaction-replacement adjacency. The protected unit is one transaction; linked contributions across multiple rows constitute multiple record changes.

For a fixed scope and allocation fractions $a_j = \rho_j/\rho_\star$, changing the total budget gives $\sigma_j = \Delta_j/\sqrt{2\rho_\star a_j}$. Relative to $\rho_\star = 4$, budgets 1, 2, and 8 multiply each standard deviation by 2, $\sqrt{2}$, and $1/\sqrt{2}$, respectively. Section 6.4 evaluates these budgets with the same active scope and allocation fractions; budget 4 remains the default operating point. Reselecting the active views or allocation fractions changes the design as well as the operating budget. For token and categorical softmax releases, a fixed calibration instead makes the clipping radius proportional to the square root of the corresponding per-release budget.

B Instance Details

B.1 Statistical views

The selected workload contains nine of 21 cataloged views, totaling 188 coordinates. Amounts are clipped at the public bound $B_a = 100000$ and normalized to $[0, 1]$ before aggregation. Count views use one-hot contributions e_{p_i}; categories follow a fixed public product-to-category map. Scope and allocations were selected on development material and frozen before both held-out cohorts.

View	Dimension	ρ_j	Δ_j
Global revenue	1	.1875	1
Revenue by day of week	7	.6875	$\sqrt{2}$
Revenue by month half	2	.03125	$\sqrt{2}$
Revenue by weekday/weekend	2	.2500	$\sqrt{2}$
Product revenue	20	.9375	$\sqrt{2}$
Product count	20	.15625	$\sqrt{2}$
Category revenue	6	.7500	$\sqrt{2}$
City revenue	10	.8750	$\sqrt{2}$
Category–product revenue	120	.1250	$\sqrt{2}$
Total	188	4	—

Table 3: The nine-view release schedule. Each view uses independent ideal Gaussian noise with standard deviation $\Delta_j/\sqrt{2\rho_j}$. Twelve cataloged views receive no release.

For a weighted histogram, if old and new records enter different bins then $\|xe_p - x'e_q\|_2^2 = x^2 + (x')^2 \le 2$; if they enter the same bin, the norm is $|x - x'| \le 1$. Each transaction contributes to one bin in each listed histogram. The global sum changes by at most one. These bounds are for fixed-size replacement adjacency.

Development configurations. The development study compares three ways to release statistics for the sales-report workload: a joint Gaussian block, question-level semantic atoms, and reusable views. Coarse shared statistics achieved 46.45% precision and 41.94% correct coverage; fine semantic atoms achieved 37.38% and 35.48%; the selected reusable-view candidate achieved 71.23% and 41.98%. These development configurations differ in scope and allocation as well as release granularity. The selected views retain a similar correct coverage to the coarse design while increasing precision.

C Numerical Categorical Diagnostics

This study explores general categorical channels beyond the analytic softmax family. It evaluates numerical capacity candidates; the Gaussian mechanisms in the report experiments have the analytic guarantee proved above.

A fixed reader scores supported, refuted, and insufficient verbalizers independently on each document and selects a record-local label $\widehat{y}_i \in \{+, -, 0\}$. After public polarity normalization, the count vector is $c = (n_+, n_-, n_0)$ with sum m. For $m = 8$, there are 45 states and 216 directed neighboring edges; one replacement changes one label. A channel row P_c assigns probabilities to $\{+, -, \perp\}$. The numerical search targets the objective

$$\rho_{\mathrm{num}}(P) = \sup_{c \sim c'} \sup_{\alpha > 1} \frac{D_\alpha(P_c \| P_{c'})}{\alpha}. \tag{27}$$

For positive channel rows $P_c, P_{c'}$, the edge cost at order $\alpha > 1$ is

$$\frac{D_\alpha(P_c \| P_{c'})}{\alpha} = \frac{\log \sum_z P_c(z)^\alpha P_{c'}(z)^{1-\alpha}}{\alpha(\alpha - 1)}. \tag{28}$$

The limit as $\alpha \downarrow 1$ is $D_{\mathrm{KL}}(P_c \| P_{c'})$. If $B = \max_z \log(P_c(z)/P_{c'}(z))$, then $D_\alpha/\alpha \le B/\alpha$, which bounds the tail. A finite two-sided cost requires compatible output support on adjacent rows; an output with positive probability on only one side yields infinite divergence in that direction.

Numerical evaluation over all 45 states and 216 directed edges gives an estimated objective of approximately 0.50 per acquisition. This is a numerical estimate of Eq. 27, rather than a certified bound on its continuous supremum.

Evidence transfer. The channel and reader were selected on development data and evaluated on 12 packages from one news topic. They achieve 87.18% precision and 26.58% coverage. No-evidence false release is 9.15%, while controlled-conflict false release is 16.89%. These numerical-channel results identify conflicting evidence as a transfer difficulty.

D Evaluation Details

Acquisition and writer interface. The sales implementation acquires its nine active views before one writer call. Dependencies are derived from official question text and public catalogs without consulting verifier values. Missing dependencies yield unresolved answers, and their questions remain in the evaluation denominator. An acquisition that fails after budget reservation remains charged. Before writing, record-conditioned temporary state is removed; raw records, unreleased aggregates, private scores, random seeds, and record-conditioned caches are excluded from writer inputs. For an acquisition expressed as a mean, the public denominator m scales the contribution sensitivity by $1/m$.

Generation and cohorts. Qwen3.5-9B runs in BF16 on one NVIDIA A40, with writer temperature 0.7, top-p 0.8, and a cap of 16384 generated tokens. Cohort A contains 32 acquisition samples, with 16 prespecified for writing; Cohort B contains 32 different samples, all used for writing. Semantic Release reports average 7346 words and 9476 generated tokens; one of 48 reaches the output cap. Table 8 gives length distributions for Public-only, Semantic Release, Oracle-active, and Non-private.

Implementation. The implementation uses finite-precision Gaussian sampling; the theoretical guarantee assumes ideal independent Gaussian draws.

Metric denominators. For sample s, let N_s be the official question count, $N_{\mathrm{schema},s}$ the schema-supported count, R_s resolved questions, and C_s structured-correct questions. Acquisition coverage, precision, and correct coverage are respectively $\sum_s R_s / \sum_s N_{\mathrm{schema},s}$, $\sum_s C_s / \sum_s R_s$, and $\sum_s C_s / \sum_s N_{\mathrm{schema},s}$. All-question recall is $\sum_s C_s / \sum_s N_s$. For report s, let a_s and c_s be answered rate and correctness. The benchmark score is $2a_s c_s/(a_s + c_s)$, or zero if either rate is zero. The official implementation rounds answer rates to four decimals before calculating each report's harmonic score; the reported local mean follows that implementation.

Evaluator protocol. The main evaluation uses GPT-5.5 with provider-managed model settings, assessing each full report and its questions together in conversation streams that retain context across reports. The local evaluator uses the same Qwen3.5-9B checkpoint as the writer, with temperature 0.7, thinking disabled, and an 8192-token response cap. It retains LongWeave's evaluation code and prompts, processing five question–answer pairs per request with the full report.

Uncertainty and disagreement. Bootstrap intervals resample paired samples; the pooled local analysis preserves the 16/32 cohort structure. The intervals measure variation across samples conditional on the observed ratings; they do not include evaluator variability or dependence from shared conversation context.

D.1 Matched reuse and privacy–utility results

Table 4 gives the full compiled-answer results for the matched comparison in Section 6.2.

Cohort	Condition	Coverage	Precision	Correct cov.	Recall
A	Question-wise acquisition	60.88	56.38	34.33	24.17
A	Semantic Release	60.22	70.32	42.35	29.81
B	Question-wise acquisition	59.06	55.57	32.82	22.90
B	Semantic Release	59.98	67.80	40.67	28.37

Table 4: Matched structured acquisition at $\rho_\star = 4$ (all values in %). Coverage and correct coverage use schema-supported questions; recall uses all official questions. Both cohorts contain 32 tables.

The sweep uses $\rho_j(\rho_\star) = (\rho_\star/4)\rho_j(4)$ with the same nine active views, development-selected allocation fractions, compiler, abstention rules, writer prompt, and evaluator protocol. Structured results use Cohort B; report results use the complete-report evaluation samples. The $\rho_\star = 4$ results reuse the default-budget run from the main comparison.

$\rho_\star$	Coverage	Precision	Correct coverage	Recall
1	50.88	50.46	25.68	17.91
2	56.63	58.99	33.41	23.31
4	59.98	67.80	40.67	28.37
8	61.43	74.88	46.00	32.09

Table 5: Full structured privacy–utility results on Cohort B (%). Metric denominators match Table 4.

$\rho_\star$	Main correct	Main answered	Local score	Local correct
1	14.76	51.91	0.301	23.22
2	20.06	53.61	0.321	25.64
4	24.74	54.43	0.336	27.88
8	28.01	54.12	0.332	27.61

Table 6: Full report privacy–utility results. Correctness and answered rates are percentages; Local score is the benchmark harmonic score.

Contrast	Evaluator	Correctness gain	95% interval
Reuse minus question-wise	Main	$+3.78$	$[+0.82, +6.76]$
Reuse minus question-wise	Local	$+1.97$	$[-0.62, +4.53]$
Budget 2 to 4	Main	$+4.68$	$[+1.22, +8.04]$
Budget 4 to 8	Main	$+3.27$	$[-0.12, +6.57]$

Table 7: Paired correctness effects in percentage points. Intervals resample paired samples conditional on the observed ratings. For reuse versus question-wise acquisition, Main answered rate changes by -0.88 points.

Condition	Median words	Word range	Mean tokens	Cap (%)
Public-only	8,171	5,649–14,000	10,799	10.42
Semantic Release	6,996	5,823–12,737	9,476	2.08
Oracle-active	6,984	4,959–13,935	10,011	10.42
Non-private	6,982	5,067–12,456	11,460	27.08

Table 8: Actual length for 48 reports per condition. The requested target is 8192 words, and the output cap is 16384 generated tokens.

E Token-Level Baseline Settings

We evaluate fixed-budget residual decoding and SM-FID on GovReport and Yelp. GovReport uses Qwen2.5-1.5B-Instruct with accounting horizon $L = 2048$ and evaluated window $G = 512$; Yelp uses TinyLLaMA with $G = 128$. Both use eight selected references, total budget $\rho = 1$, and a radius-aware public Top-k+ sampler with base $k = 50$. The default SM-FID allocation is proportional to $(t + 64)^{-1.4}$, normalized over all positive token positions. Fact@512 measures support-labeled factual recall in the first 512 tokens; TopicAcc@100 measures topic correctness in the first 100 tokens.

Positional acquisition comparison. Table 9 uses the same reference count, total budget, sampler, and SM-FID schedule. A forced public prefix places one private event at positions 1, 2048, or 8192; token decoding uses the positional budget, while Semantic Release acquires the state for that decision.

Allocation results. Fixed-budget residual decoding and SM-FID give GovReport Fact@512 scores of 0.1172 and 0.1448, and Yelp TopicAcc@100 scores of 0.10 and 0.18, respectively. At position 8192, its acquisition score is $5.78\times$ that of SM-FID. These experiments examine the allocation layer under reference replacement; the complete sales-report comparison uses transaction replacement and a matched question-wise baseline.

Condition	Evidence position 1	2048	8192
Fixed token	23.19	15.23	11.44
SM-FID	35.52	15.00	11.14
Semantic Release	99.96	65.46	64.36

Table 9: Acquisition scores (%) at one private event after a fixed public prefix.

Task	Decoder	Realized ρ	Task metric
GovReport	Fixed-budget ($L = 2048$)	.2500	.1172
GovReport	SM-FID	.5836	.1448
Yelp	Fixed-budget ($L = 128$)	1.0000	.1000
Yelp	Fixed-budget ($L = 361$)	.3546	.1000
Yelp	SM-FID	.3543	.1800

Table 10: Token-level baseline results: Fact@512 on GovReport and TopicAcc@100 on Yelp.