# Privacy-Preserving Retrieval-Augmented Generation with Plausible Deniability

**Wenxuan Bao** Visa Research
**Shan Jin** Visa Research
**Vincent Bindschaedler** University of Florida
**Yiwei Cai** Visa Research

## Abstract

We introduce **PD-RAG**, a technique for retrieval-augmented generation (RAG) that leverages a language model's own randomness to safeguard privacy. The algorithm partitions documents into groups, generates a candidate answer from a randomly chosen group, and then uses a privacy test to enforce that the released answer could have been produced by multiple disjoint document groups, thereby ensuring a controllable degree of *plausible deniability*. Compared to existing methods that operate at the token level and therefore incur both substantial utility loss and growing privacy budget per generated token, PD-RAG operates at the answer level, so the privacy it offers does not loosen with output length. We prove that PD-RAG satisfies $(\varepsilon, \delta)$-differential privacy. We experimentally evaluate PD-RAG on three QA benchmarks using three language models and find that it reduces membership inference attack advantage to near random while consistently outperforming alternative methods on all four utility metrics we used and running between 17 and 33$\times$ faster.

## 1 Introduction

The dominant paradigm for knowledge-intensive language tasks is Retrieval-Augmented Generation (RAG) [22], which grounds language model outputs in documents retrieved from an external corpus [11]. RAG is used for applications across medicine [23], science [7], law [31], and enterprise [41] search. However, the retrieval corpus often contains sensitive data (e.g., patient records, legal case files, or proprietary documents) which raises privacy concerns. Recent work has demonstrated that RAG-generated answers leak information about which documents were retrieved. For example, the $S^2$MIA attack [24], which compares RAG outputs to known references, achieves ROC AUC of between 0.78 and 0.97 across various datasets.

To protect RAG systems against membership inference and other privacy threats, a number of methods have been proposed [21, 14] that target differential privacy (DP) [9]. These RAG methods operate at the *token level*, for example running multiple model instances to generate next-token candidates in parallel, and then leveraging a DP mechanism to select each output token. Such designs are inherently limited since the privacy budget grows linearly in terms of the answer's token-length, resulting in poor privacy-utility tradeoffs. In particular, we find that DPSparseVoteRAG [21] at $\varepsilon$=50 retains only between 11% and 28% of (non-private RAG) answer quality. Keyword-level approaches (e.g., DP-KSA [35]) operate on keywords instead of tokens, but we find that for reasonable privacy budgets, output quality approaches data-independent generation.

We propose **PD-RAG** (Figure 1), a RAG method that operates at the *answer level* and uses the language model's own sampling randomness alongside with a privacy test to achieve privacy. PD-RAG randomly partitions retrieved documents into groups, generates a complete candidate answer from a randomly chosen group, and then applies a *privacy test* to check whether the candidate answer could plausibly have been produced from multiple (other) document groups. If this test passes, the answer is released; otherwise, the system retries or falls back to data-independent generation.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

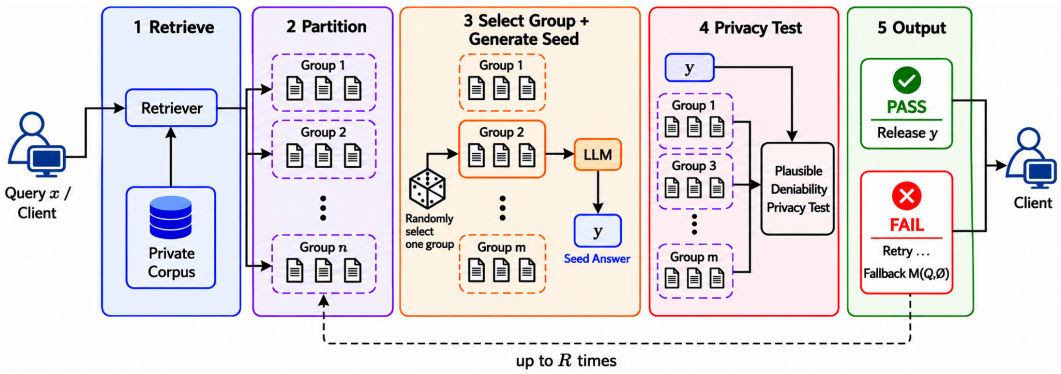



Figure 1: **PD-RAG framework overview.** ① A query retrieves the top $N$ documents from a private corpus. ② Documents are randomly partitioned into $m$ disjoint groups. ③ A seed group is chosen uniformly at random and the LLM generates a candidate answer $y$ from that group. ④ The privacy test scores $y$ under every group via forward passes, counts how many groups yield similar log-probabilities (integer counting), adds geometric noise, and checks whether the noisy count exceeds a threshold $K$. ⑤ If the test passes, $y$ is released; otherwise the system retries with a new partition (up to $R$ times) or falls back to data-independent generation $\mathcal{M}(Q, \varnothing)$.

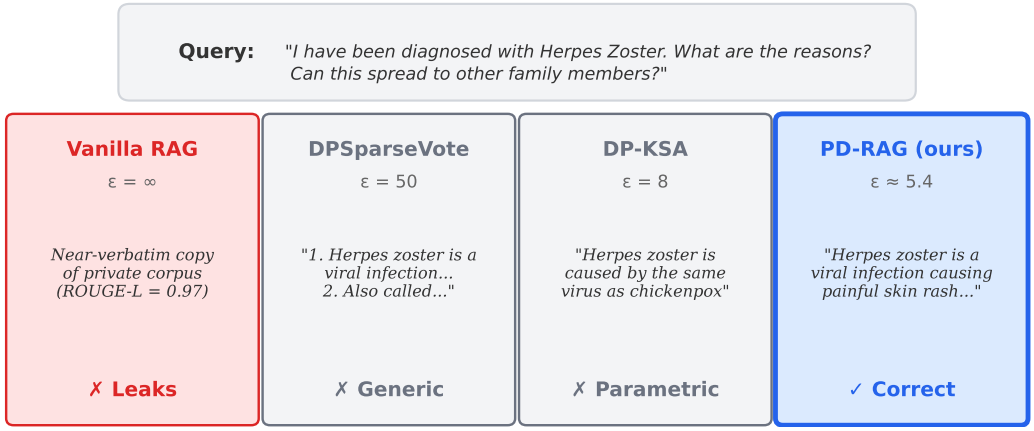



Figure 2: **Example outputs for a medical query (ChatDoctor).** Vanilla RAG leaks private corpus content verbatim. DP baselines produce degraded or generic outputs. PD-RAG generates a correct answer with no verbatim leak at lower $\varepsilon$.

Figure 2 illustrates the problem and our solution on a real medical query (from ChatDoctor). Vanilla RAG copies the corpus answer nearly verbatim (ROUGE-L $= 0.97$), directly exposing private content. Token-level DP (DPSparseVoteRAG) produces a generic numbered list, while keyword-level DP (DP-KSA) falls back to data-independent generation. By contrast, PD-RAG generates a natural, medically accurate answer with no verbatim leak, while requiring a lower privacy budget.

Our contributions:

- We propose PD-RAG, a RAG algorithm that protects privacy at the answer level by leveraging the language model's sampling randomness in combination with a privacy test. We show that PD-RAG achieves $(\varepsilon, \delta)$-DP.
- Experiments on three datasets and three language models reveal that PD-RAG reduces attack advantage to near random-guessing (Adv. AUC $\leq 0.55$) while achieving the best utility among all tested privacy-preserving methods.
- PD-RAG is substantially more efficient and between 17 and 33$\times$ faster than other DP approaches for RAG. Compared to vanilla RAG, PD-RAG incurs an overhead of only 3$\times$.

## 2 Background and Related Work

**Retrieval-Augmented Generation (RAG).** RAG [22, 11, 2] conditions language model responses on documents retrieved from an external corpus. Given a query $Q$, a dense retriever (e.g., DPR [20]) selects the most relevant passages from a document collection $D$, then an autoregressive language model generates an answer grounded in those passages. Privacy concerns arise if the retrieval corpus contains sensitive data. For example, an adversary may be able to determine whether a specific document was used in generating a response. This is called a *membership inference attack*. Recent work [24] shows that such attacks achieve AUC between 0.78 and 0.97 across various datasets.

**Differential Privacy.** Differential privacy (DP) [9, 10] is a framework for privacy protection that defines privacy in terms of the influence of any single record on a computation's output.

**Definition 1.** *A mechanism $\mathcal{M}$ satisfies* $(\varepsilon, \delta)$-differential privacy *if for neighboring datasets $D, D'$ and any measurable set $S \subseteq \mathrm{Range}(\mathcal{M})$:*

$$\Pr[\mathcal{M}(D) \in S] \;\leq\; e^{\varepsilon} \cdot \Pr[\mathcal{M}(D') \in S] + \delta\,. \tag{1}$$

Parameters $\varepsilon > 0$ and $\delta$ determine the privacy level and are often called the privacy budget. Definition 1 can be applied with different notions for neighboring datasets. The "add/remove" (aka "unbounded") setting is if $D'$ can be obtained from $D$ by adding or removing one document, whereas the "substitution" (aka "bounded") setting is if $D'$ can be obtained by replacing one document in $D$ with another document. In this paper, we use the *bounded* setting.

We use composition results to adjust privacy guarantees when using multiple DP mechanisms [10, 18]. In particular, sequential composition of $k$ $(\varepsilon, \delta)$-DP mechanisms yields $(k\varepsilon, k\delta)$-DP [10].

**Plausible Deniability.** Plausible deniability (PD) [3, 4] is an approach to mechanism design that uses a privacy test to discard privacy violating outputs. In this approach, privacy violation is defined in terms of the degree to which a given output could have *plausibly* been produced if the input data had been different. Bao et al. [3] apply this idea to privatize SGD and call their algorithm PD-SGD. Their algorithm works by partitioning training data into batches, selecting a batch and adding noise to its gradient. The privacy tests counts how many alternative batches could have produced similar gradients. A gradient update is then only applied to the parameters if the count exceeds a threshold. The authors show that this method can provide $(\varepsilon, \delta)$-DP. The PD-RAG mechanism we propose in this paper can be viewed as continuing this line of work to also achieve $(\varepsilon, \delta)$-DP in the RAG setting. Notably, however, PD-RAG leverages a language model's own sampling randomness as noise to achieve DP (as opposed to explicitly adding noise).

**Existing Privacy-Preserving RAG Methods.** There are a number of RAG approaches that provide DP at the *token level*. DPSparseVoteRAG [21] uses $m$ parallel LLM instances, each conditioned on a disjoint document subset, and selects each output token via a differentially private voting mechanism. They use the sparse vector technique (SVT) [27, 48, 6, 19] and halt generation early to conserve budget. Grislain [14] applies the exponential mechanism [28] per token; Wu et al. [38] extend to multi-query settings. The fundamental limitation is that privacy budget scales linearly with answer length, forcing either short outputs or large $\varepsilon$. We find in experiments that DPSparseVoteRAG for $\varepsilon$=50 retains only between 11 and 28% of vanilla RAG utility (Section 4.2).

Another RAG approach is to provide DP at the *keyword level*. DP-KSA [35] extracts keywords from an ensemble of LLM responses. It then selects a sanitized top-$k$ keyword set privately using Propose-Test-Release (PTR) [5]. Finally, it performs zero-shot inference using only the sanitized keywords. The idea is that since DP-KSA operates in a smaller keyword space, noise is reduced. In our experiments, for moderate privacy budgets ($\varepsilon \leq 8$), DP-KSA's output quality approaches the data-independent baseline (Section 4.2), indicating that the PTR mechanism filters out most informative keywords.

By contrast, PD-RAG operates at the *answer level* with its privacy cost independent of answer length. We also emphasize that PD-RAG is unique in that it uses the LLM's own sampling randomness as noise to achieve differential privacy.

## 3 Plausible Deniability for Retrieval-Augmented Generation

PD-RAG is a mechanism to protect privacy in the RAG setting. The mechanism computes the probability of producing a given answer from the LLM when given disjoint document groups and releases that answer only if it passes the *plausible deniability* privacy test.

The intuition for privacy is the following. Since adding/removing a document from the corpus changes at most one group, if a given answer from the LLM can be produced from multiple disjoint groups, then that answer cannot possibly reveal much information about any single private document.

### 3.1 Setup and Problem Statement

**Language model (LM).** We consider an autoregressive model $\mathcal{M}$ that produces outputs over a token vocabulary $V$. Given a prompt $x$, $\mathcal{M}$ generates a token sequence $y = (y_1, \ldots, y_n)$ with probability

$$P_{\mathcal{M}}(y \mid x) \;=\; \prod_{t=1}^{n} P_{\mathcal{M}}(y_t \mid y_{<t}, x)\,, \tag{2}$$

where $y_{<t} = (y_1, \ldots, y_{t-1})$. At each step of the generation, the model's logit scores over $V$ are produced and a softmax with temperature $T > 0$ is applied. A token is selected with probability proportional to its softmax weight. We consider outputs of length at most $n_{\max}$.

We assume that we can compute the log-probabilities $\log P_{\mathcal{M}}(y|x)$ (log-likelihoods) for any $x, y$. This is the case for open-weight models. Note that $P_{\mathcal{M}}(y \mid x) > 0$ for any $x$ and $y$.

An important observation (that PD-RAG's design exploits) is that computing probabilities according to Eq. (2) for a fixed answer $y$ is computationally much cheaper than sampling an answer $y$. This is because concatenating $x$ and $y$, we can compute all the token logits in a single forward pass (leveraging the LM's causal mask).

**RAG system.** Let $D$ denote a private document corpus. A RAG pipeline is composed of a retriever $\mathcal{R}$ and a language model $\mathcal{M}$. Given a query $Q$, the retriever selects the most relevant documents from $D$, and the LM generates an answer conditioned on the query and retrieved documents.

**Threat model.** We consider an adversary that observes a RAG output $y$ (for some potentially known query $R_Q$) and attempts to determine whether a specific document $d^\star$ was included in the external database $D$. In other words, the adversary seeks to distinguish between the hypotheses: $H_0$: $d^\star \notin D$ (non-member), and $H_1$: $d^\star \in D$ (member).

This is a membership inference problem and we note several related works proposing and evaluating membership inference attacks on RAG systems [24, 1, 42, 36].

### 3.2 Privacy Testing and $\alpha$-Similarity

PD-RAG uses the language model's own sampling randomness to protect privacy and follows the mechanism design approach of using a privacy test based on plausible deniability [3, 4]. The key insight is that if an answer is similarly likely to be produced from disjoint document groups, then it cannot reveal much information about (the membership of) any specific document. Said differently, one can *plausibly deny* the use of a particular document group by pointing out a different (disjoint) group that plausibly explains the observed answer. We formalize this through the concept of $\alpha$-similarity on the LM's generation probabilities.

**Definition 2** ($\alpha$-Similarity of Document Groups)**.** *For a fixed answer $y$, query $Q$, and two document groups $G_i, G_j$, we say that $G_i$ and $G_j$ are $\alpha$-**similar** with respect to $y$ if*

$$\frac{1}{\alpha} \;\le\; \frac{P_{\mathcal{M}}(y \mid Q, G_i)}{P_{\mathcal{M}}(y \mid Q, G_j)} \;\le\; \alpha\,, \tag{3}$$

*where $\alpha \ge 1$ is a privacy parameter.*

We omit $Q$ (and $y$) and write $G_i \simeq_\alpha G_j$ when clear from context.

Intuitively, if $G_i \simeq_\alpha G_j$, then the model produces $y$ with comparable probabilities under both context groups. Therefore, an adversary observing $y$ cannot determine which group was used, beyond a

likelihood ratio of $\alpha$. Letting $\gamma = \log(\alpha)$, the condition of Eq. (3) can be efficiently checked using the log-probabilities:

$$\left|\log P_{\mathcal{M}}(y \mid Q, G_i) - \log P_{\mathcal{M}}(y \mid Q, G_j)\right| \;\leq\; \gamma\,. \tag{4}$$

Since $\alpha$-similarity can be checked efficiently during RAG generation, we can safeguard privacy by rejecting (i.e., discarding) answers $y$ for which we cannot find sufficiently many $\alpha$-similar disjoint document groups. This effectively redistributes the probability mass from some (not plausibly deniable) answers $y$ to "no answer" ($\perp$) or a fallback answer.

**Privacy testing.** We define a privacy test with four parameters $(\gamma, K, \beta, \psi)$ to control the privacy level. Given a seed group $G_i$ for some $i \in \{1, 2, \ldots, m\}$ (with answer $y$) and the full partition $\mathfrak{G} = (G_1, \ldots, G_m)$:

$$\textsc{PrivacyTest}(y, i, \mathfrak{G};\, \gamma, K, \beta, \psi): \quad \begin{cases} \textsc{pass} & \text{w.p. } 1{-}\psi, \text{ if } \tau + Z \geq K\,, \\ \textsc{fail} & \text{otherwise,} \end{cases} \tag{5}$$

where $\tau = \textsc{CountSimilar}(y, i, \mathfrak{G};\, \gamma)$ counts the number of $\alpha$-similar plausible alternative groups. Here $Z \sim \mathrm{Geom}(\beta)$ is geometric noise [12] on the integers with $\Pr(Z{=}x) = \frac{\beta-1}{\beta+1}\beta^{-|x|}$ and $\beta = e^{\varepsilon_0} > 1$, and $\psi \in (0, 1)$ is a ceiling probability.

The test has the following components:

- **Counting** ($\tau$): measures how many alternative groups could plausibly have produced $y$.
- **Geometric noise** ($Z \sim \mathrm{Geom}(\beta)$): randomizes the effective threshold, preventing the adversary from inferring $\tau$ precisely from the pass/fail outcome.
- **Ceiling** ($1 - \psi$): even when $\tau + Z \geq K$, the test probabilistically fails (based on $\psi > 0$). This caps the acceptance probability at $1 - \psi$, which reduces information leakage of rejection events.

Setting $\beta \to \infty$ (no geometric noise) and $\psi = 0$ (no ceiling) recovers a deterministic test that passes only when $\tau \geq K$.

**Counting plausible deniability.** Counting plausible alternative groups according to Eqs. (3) and (4) does not allow for the most stringent privacy guarantees (in the DP sense). Instead, we use the integer counting method [3, 4]. That is, $\tau = \textsc{CountSimilar}(y, i, \mathfrak{G}; \gamma)$ is computed as

$$\tau \;=\; \left|\{j \in [m] \setminus \{i\} : \lfloor \ell_j/\gamma \rfloor = \lfloor \ell_i/\gamma \rfloor\}\right|, \tag{6}$$

with $\ell_j = \log P_{\mathcal{M}}(y \mid Q, G_j)$.

With this counting method, two groups are $\alpha$-similar iff their log-probabilities fall in the same bin of width $\gamma = \log \alpha$. Notably, this counting method has sensitivity 1 under addition/removal/substitution of any group. The probability $P_{\mathcal{M}}(y \mid Q, G)$ for any group $G$ falls into exactly one bin and therefore adding/removing one group increases/decreases $\tau$ (for that bin or any other) by at most 1.

### 3.3 The PD-RAG Algorithm

Algorithm 1 presents the complete PD-RAG pipeline (illustrated in Fig. 1). We summarize key notation in Table 5. On input corpus $D$ and query $Q$, the algorithm first uses the retriever $\mathcal{R}$ to obtain the $N$ most relevant documents from $D$. The algorithm then makes up to $R+1$ attempts at producing an answer $y$. In each attempt, the retrieved documents are randomly partitioned into $m$ equal-sized groups $\mathfrak{G} = (G_1, \ldots, G_m)$. A *seed* group $G_i$ ($i \in [m]$ chosen uniformly at random) is selected and the language model $\mathcal{M}$ generates a candidate answer $y$ from it. The log-probabilities $\ell_j$ of producing $y$ from each group $G_j$ ($j = 1, \ldots, m$) are then computed, and the plausible alternative count $\tau$ is obtained. Geometric noise is added to $\tau$ and the noisy count is compared to the threshold $K$. If this threshold is met, a Bernoulli random variable with success probability $1 - \psi$ simulates the ceiling. If the privacy test passes, the answer $y$ is output. Otherwise, the attempt fails and the process repeats. If none of the $R + 1$ attempts succeed, the language model $\mathcal{M}$ produces a data-independent fallback answer *without* access to any documents from $D$.

The algorithm has a number of features that deserve discussion. In particular, there are four sources of randomness that impact both utility and privacy:

**Algorithm 1** PD-RAG: Plausibly Deniable Retrieval-Augmented Generation

**Input:** Query $Q$; corpus $\mathcal{D}$; retriever $\mathcal{R}$; model $\mathcal{M}$; params $m, \gamma, K, R, T, \beta, \psi$
**Output:** Answer $y$ satisfying the privacy test, or a data-independent fallback

Retrieve top-$N$ documents from $\mathcal{D}$ using $\mathcal{R}$
**for** $r = 0, 1, \ldots, R$ **do** ▷ Up to $R{+}1$ attempts
  Partition retrieved documents into $m$ random groups $\mathfrak{G} = (G_1, \ldots, G_m)$
  $i \leftarrow \mathrm{Unif}([m])$ ▷ Pick seed group
  $y \leftarrow \mathcal{M}(Q, G_i; T)$ ▷ Generate candidate answer from seed group
  $\ell_j \leftarrow \log P_{\mathcal{M}}(y \mid Q, G_j)$ for all $j \in [m]$ ▷ Score $y$ under each group
  $\tau \leftarrow \textsc{CountSimilar}(y, i, \mathfrak{G}; \gamma)$;  $Z \sim \mathrm{Geom}(\beta)$
  **if** $\tau + Z \geq K$ **and** $\mathrm{Bern}(1{-}\psi) = 1$ **then**
    **return** $y$ ▷ Privacy test passed
  **end if**
**end for**
**return** $\mathcal{M}(Q, \varnothing)$ ▷ Data-independent fallback

**Random partitioning.** In each iteration, the retrieved document set is randomly partitioned into $m$ groups uniformly at random so that each document is placed in exactly one group. This ensures that the presence or absence of any document in the corpus impacts at most one group and that adding/removing/substituting one group changes the count of alternatives $\tau$ by at most 1.

**Token sampling.** The algorithm leverages the LM's token sampling for privacy protection. Sampling of the candidate answer $y$ is done using the full token distribution with logits scaled by $1/T$ before the softmax. Therefore, higher temperature increases sampling diversity. This can lower utility [47, 33, 8], but it also flattens the probability distribution making it easier to find $\alpha$-similar groups for a fixed $\alpha$ (and thus increasing the probability of passing the test).

**Threshold randomization.** The count of alternatives $\tau$ is deterministic once the answer $y$ is determined (and the groups are fixed). But the algorithms adds symmetric geometric noise $\mathrm{Geom}(\beta)$ to $\tau$ (or equivalently to the threshold $K$). The greater this noise (i.e., the smaller $\beta > 1$), the less predictable the test becomes since it makes it both more likely to pass the test with too few alternative as well as to fail the test with more alternatives than necessary.

**Ceiling on pass probability.** The privacy test has a ceiling, ensuring that the probability of passing the test even with abundant alternatives is capped at $1 - \psi$. This limits (membership) information leakage from rejections.

Since the privacy test can fail, the algorithm makes up to $R + 1$ attempts at producing an answer. If all these attempts fail, it produces a *data-independent* fallback answer $\mathcal{M}(Q, \varnothing)$ that accesses no private documents and therefore does not leak any private information (its output distribution is *identical* under any corpus $D$). However, the number of attempts (and the fact that some number of them may have failed) does potentially leak information since it depends on the document corpus. We show in the next section, that this leakage is bounded and we obtain an overall $(\varepsilon, \delta)$-DP guarantee.

### 3.4 Privacy Analysis

PD-RAG satisfies $(\varepsilon, \delta)$-DP by composing over the (up to) $R + 1$ attempts. Let $\mathcal{F}$ denote one attempt of Algorithm 1 with parameters $(T, m, K, \gamma, \beta, \psi)$. The range of $\mathcal{F}$ is $\Omega \cup \{\perp\}$, where $\Omega$ includes all tokens sequences of length at most $n_{\max}$ and special symbol $\perp$ indicates failure to pass the test.

**Theorem 1** (Per-Attempt Privacy Budget)**.** *For any two neighboring corpora $D, D'$ that differ in exactly one document and any integer $1 \leq t < K$, $\mathcal{F}$ satisfies differential privacy, i.e.:*

$$\Pr(\mathcal{F}(D) \in S) \leq e^{\varepsilon(t)} \Pr(\mathcal{F}(D') \in S) + \delta(t) \,,$$

*where $\varepsilon(t)$ and $\delta(t)$ satisfy the following.*

*For any output set $S \subseteq \Omega$:*

$$\varepsilon(t) = \varepsilon_{\mathrm{pass}}(t) = \log \beta + \log\left(1 + \frac{e^{\gamma}}{t}\right) \,, \qquad \delta(t) = \frac{(1-\psi)\,\beta^{-(K-t)}}{(1+\beta^{-1})\,m} \,. \tag{7}$$

*For $S = \{\perp\}$ (failure to pass):*

$$\varepsilon = \varepsilon_\perp = \log\Big(\zeta\big(1-\tfrac{1}{m}\big) + \tfrac{1}{m\psi}\Big)\,, \qquad \delta = \delta_\perp = 0\,. \tag{8}$$

*with $\zeta = \frac{\beta+\psi}{1+\beta\psi}$ (and $\zeta \le \beta$ whenever $\psi > 0$).*

Note that we operate in the bounded DP setting ($D'$ can be obtained from $D$ by substituting one document). Also, it is easy to see that Theorem 1 implies that one attempt of PD-RAG satisfies $(\max\{\varepsilon(t), \varepsilon_\perp\}, \delta(t))$-DP.

The free parameter $t$ in Theorem 1 allows us to navigate the tradeoff between $\varepsilon$ and $\delta$ in the case where the test passes. We defer the proof of Theorem 1 to Appendix B.

**Composition.** To obtain a privacy guarantee for Algorithm 1 (all attempts), we use sequential composition. Out of (up to) $R+1$ attempts, at most one passes the test (since we exit the loop if this happens). Therefore, the total budget is (at most) $\varepsilon_{\text{total}} = R\,\varepsilon_\perp + \max\{\varepsilon_\perp, \varepsilon_{\text{pass}}(t)\}$ and $\delta_{\text{total}} = \delta(t)$. Note that the $\delta$ terms does not depend on $R$ since the $\delta_\perp = 0$. The fallback answer $\mathcal{M}(Q, \varnothing)$ contributes no privacy loss and can be viewed as post-processing of $(R+1)$ $\perp$-events.

In experiments, we set the parameters such that the privacy loss of producing a data-dependent answer dominates, i.e., $\varepsilon_\perp \le \varepsilon_{\text{pass}}(t)$. The total budget is (at most) $R\,\varepsilon_\perp + \varepsilon_{\text{pass}}(t)$. We also often set $t = 1$ to minimize $\delta$. We discuss this and parameter tuning in Appendices C.4 and C.10.

# 4 Experiments

We evaluate PD-RAG across three diverse QA domains using three different language models. We measure privacy using the privacy budget but also the AUC for membership inference attacks (MIA). We measure utility using a number of answer quality metrics. We compare these results to privacy-preserving RAG baselines [21, 35]. We provide full experimental setup details in Appendix C.

## 4.1 Setup

**Datasets and models.** We use three QA datasets: **ChatDoctor** [23] (medical dialogues, 50k corpus), **QASPER** [7] (scientific paper QA, 1.5k corpus), and **ASQA** [34] (ambiguous factoid QA, 2.7k corpus). The primary model is **Llama-3.1-8B** [13] (base) with DPR [20] retrieval. We provide cross-model results for two other LMs (Mistral-7B [16], Llama-3.2-3B-Instruct) in Appendix C.5. As membership inference attack, we use **S$^2$MIA-G** [24] (BLEU-based gold-answer comparison) a dedicated attack for RAG systems, following [21]. Unless otherwise specified, each experiment samples 100 member + 100 non-member queries. We average results over 3 random seeds.

**Metrics.** We use **Adversarial AUC** (Adv. AUC $= \max(\text{AUC}, 1-\text{AUC})$; 0.5 = random guess) and **TPR@1%FPR** (member detection rate at 1% false positive rate) as privacy metrics. For utility we use: **ROUGE-L** (RL) [25], longest-common-subsequence overlap; **ROUGE-1** (R1) [25], unigram recall, which is particularly informative for long-form answers where exact phrasing varies but key terms are preserved; **Token F1** [32], harmonic mean of unigram precision and recall against gold answers; and **LLM-as-Judge** (LJ) [46], GPT-5.4 quality rating (1–5). We also report the **Retrieval-Grounded Answer Rate**: the fraction of queries for which PD-RAG outputs a retrieval-grounded answer (i.e., passes the privacy test) rather than falling back to data-independent generation. We abbreviate it as *Answer Rate* in tables.

**Baselines and configuration.** We compare against three baselines: (1) **Vanilla RAG** (no privacy, upper bound on utility); (2) **DPSparseVoteRAG** [21] (token-level DP voting, $n_{\text{split}}$=50, $\varepsilon$=50); (3) **DP-KSA** [35] (keyword sanitization via Propose-Test-Release, $m$=50, $\varepsilon$=8, $\delta$=$10^{-5}$). For PD-RAG, we set the per-attempt budget $(\varepsilon, \delta)$ according to Theorem 1 and compose over all up $R+1$ attempts (Section 3.4). Concretely, for the original headline ChatDoctor configuration ($\gamma$=2.1, $K$=8, $m$=20, $\beta$=5, $\psi$=0.05, $R$=1): the passing attempt costs $\varepsilon_{\text{pass}} = \log\beta + \log(1+e^\gamma/t) = 3.82$; each failing attempt costs only $\varepsilon_\perp = 1.58$ (since $\zeta$=4.04 $<$ $\beta$, failing leaks less); the total is $\varepsilon_{\text{total}} = R\,\varepsilon_\perp + \varepsilon_{\text{pass}} = 5.40$ with $\delta$=$5.1\times10^{-7}$ (substantially lower than DP-KSA's $\varepsilon$=8 and $\delta$=$10^{-5}$). Per-dataset configurations are in Appendix C.4 (Table 7).

Table 1: **Non-RAG (data-independent only)** vs. Vanilla RAG utility measured by ROUGE-L. Non-RAG generates answers without performing any retrieval, measuring intrinsic domain knowledge of the LM. We measure RAG Gain as: (Vanilla RL − Non-RAG RL) / Non-RAG RL.

| Model | Dataset | Non-RAG RL | Vanilla RL | RAG Gain |
|---|---|---|---|---|
| Llama-3.1-8B | ChatDoctor | 0.115 | 0.346 | +201% |
| | QASPER | 0.065 | 0.204 | +214% |
| | ASQA | 0.170 | 0.494 | +191% |
| Mistral-7B | ChatDoctor | 0.110 | 0.314 | +185% |
| | QASPER | 0.060 | 0.208 | +247% |
| | ASQA | 0.165 | 0.506 | +207% |
| Llama-3.2-3B | ChatDoctor | 0.100 | 0.169 | +69% |
| | QASPER | 0.055 | 0.193 | +251% |
| | ASQA | 0.150 | 0.398 | +165% |

Table 2: **Privacy–utility comparison** (Llama-3.1-8B, $S^2$MIA-G attack). Values are means over 3 random seeds; ± denotes standard deviation. **Bold**: best among privacy-preserving methods (excluding Vanilla RAG). Adv. AUC $= \max(\text{AUC}, 1-\text{AUC})$; 0.5 = random guess. TPR@1%: true positive rate at 1% FPR. RL: ROUGE-L. R1: ROUGE-1. F1: Token F1. LJ: LLM-as-Judge (1–5). PD-RAG: $\varepsilon_{\text{total}} \leq 5.40$, $\delta = 5.1 \times 10^{-7}$ (Theorem 1). DP-KSA [35]: $\varepsilon = 8$, $\delta = 10^{-5}$. DPSparseVoteRAG [21]: $\varepsilon = 50$. Cross-model results in Appendix C.5. Answer Rate: retrieval-grounded answer rate (Section 4.1).

| Dataset | Method | Adv. AUC ↓ | TPR@1% ↓ | RL ↑ | R1 ↑ | F1 ↑ | LJ ↑ |
|---|---|---|---|---|---|---|---|
| ChatDoctor | Vanilla RAG | .823±.046 | .617±.087 | .346±.050 | .480±.044 | .297±.035 | 2.47±.06 |
| | DPSparseVote | .524±.015 | .015±.010 | .095±.002 | .207±.008 | .087±.005 | 2.01±.02 |
| | DP-KSA | .521±.016 | .000±.000 | .108±.003 | .237±.010 | .123±.004 | 2.06±.03 |
| | **PD-RAG** | **.512**±.012 | .037±.025 | **.126**±.004 | **.264**±.008 | **.135**±.005 | **2.35**±.07 |
| QASPER | Vanilla RAG | .931±.026 | .885±.022 | .204±.006 | .553±.015 | .172±.006 | 1.94±.03 |
| | DPSparseVote | .519±.017 | .020±.010 | .057±.005 | .060±.006 | .051±.004 | 1.19±.03 |
| | DP-KSA | .516±.013 | .010±.000 | .062±.001 | .182±.006 | .050±.000 | 1.32±.05 |
| | **PD-RAG** | **.511**±.010 | .017±.015 | **.073**±.002 | **.242**±.002 | **.062**±.001 | **1.57**±.05 |
| ASQA | Vanilla RAG | .985±.008 | .830±.074 | .494±.009 | .724±.015 | .367±.007 | 2.75±.04 |
| | DPSparseVote | .510±.008 | .010±.010 | .053±.002 | .089±.001 | .053±.001 | 1.48±.06 |
| | DP-KSA | **.503**±.000 | .010±.000 | .184±.006 | .300±.005 | .110±.001 | 2.17±.06 |
| | **PD-RAG** | .515±.013 | .017±.006 | **.204**±.004 | **.401**±.005 | **.148**±.005 | **2.52**±.05 |

We found these configurations using a three-step search process (Algorithm 2) described in details in Appendix C.10). This process exploits the privacy budget formulae from Theorem 1 to prune out configurations that do not satisfy the privacy desideratum before any LM inference takes place. In experiments this reduces hyperparameter search cost by up to $13\times$ over brute force. Full dataset, model, and attack details are in Appendix C.

**Non-RAG (Data-Independent Only) Baseline.** A related question is: *does RAG provide a benefit in the considered setting?* We perform experiments to answer this question quantitatively and show the results in Table 1. This is the *data-independent knowledge floor*, i.e., the answer quality that is achievable without any retrieval, measured by ROUGE-L. We observe that all datasets show large RAG gains (between $+191$ and $214\%$ for 8B), proving that retrieval drastically improves answer quality. When PD-RAG's privacy test fails and falls back to data-independent generation, output quality drops to this floor. This is why the utility gap between PD-RAG and Vanilla RAG is largest on specialized domains (QASPER) where data-independent knowledge is weakest.

### 4.2 Main Results: Privacy-Utility Tradeoff

Results are shown in Table 2 and Fig. 3. All three privacy-preserving methods reduce Adv. AUC to near 0.5 (random guess): PD-RAG 0.511–0.515, DP-KSA 0.503–0.521, DPSparseVoteRAG 0.510–0.524. This is down from 0.823–0.985 for (non-private) Vanilla RAG. Empirically, the three methods provide *comparable* protection against membership inference attacks. However, their privacy budgets

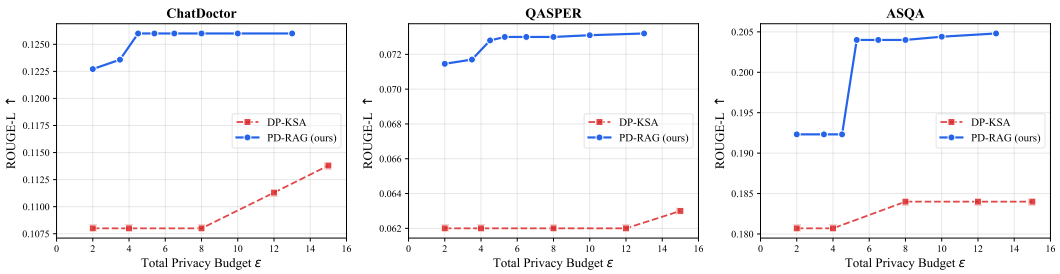



Figure 3: **ROUGE-L vs. total privacy budget $\varepsilon$ (Llama-3.1-8B).** PD-RAG (blue) dominates DP-KSA (red) across the full $\varepsilon$ range on all three datasets. DPSparseVoteRAG ($\varepsilon$=50) falls outside the plotted range.

Table 3: **Component ablation** (ChatDoctor, Llama-3.1-8B, $m$=5, $K$=2, $R$=2). Each row modifies the PD-RAG pipeline. "Deterministic test" removes geometric noise and ceiling (no noise, $\psi$=0). Always passes means the test won't reject any candidate answers (simulated using a very large $\gamma = 100$). "Random rejection" replaces the PD test entirely with a coin flip matching PD-RAG's answer rate.

| **Method** | Adv. AUC ↓ | F1 ↑ | Answer Rate % |
|---|---|---|---|
| PD-RAG (full) | 0.515 | 0.142 | 35.5 |
| No retries ($R$=0) | 0.510 | 0.131 | 16.5 |
| Deterministic test | 0.537 | 0.132 | 25.5 |
| Always Passes | 0.528 | 0.138 | 100.0 |
| Random rejection | 0.533 | 0.125 | 35.5 |

differ substantially. DPSparseVoteRAG requires $\varepsilon$=50 because its token-level voting consumes budget per generated token. DP-KSA operates at $\varepsilon$=8 with $\delta$=$10^{-5}$. By contrast, PD-RAG requires only $\varepsilon_{\text{total}} \leq 5.40$ with $\delta$=$5.1\times10^{-7}$ (a $\approx 9\times$ reduction compared to DPSparseVoteRAG and 32% below DP-KSA with a much lower $\delta$). The gap is driven by the fact that PD-RAG operates at the answer level and incurs privacy cost only once per query (independent of answers' token length).

Furthermore, PD-RAG achieves the best score on *every* utility metric across all three datasets (Table 2, bold): ROUGE-L 0.126/0.073/0.204, Token F1 0.135/0.062/0.148, and LLM-as-Judge 2.35/1.57/2.52. Compared to DP-KSA (the strongest baseline), PD-RAG improves ROUGE-L by 11–18% and Token F1 by 10–35%. DPSparseVoteRAG, despite requiring a much larger privacy budget, produces severely degraded outputs (e.g., ROUGE-L as low as 0.053 on ASQA). The superior performance of PD-RAG is not configuration-specific as Fig. 3 shows. PD-RAG dominates across the entire $\varepsilon \in [2, 15]$ range on all datasets. Surprisingly, increasing the privacy budget of DP-KSA to a moderate range, does not drastically improve utility. We believe the root cause is that DP-KSA compresses documents into a sparse set of keywords (averaging 4–7 per query), so even if PTR passes there is insufficient context for the LLM to generate high quality answers.

### 4.3 Ablation Studies and Additional Experiments

**Component ablation.** We ablate the PD-RAG pipeline on ChatDoctor to determine whether all components in Algorithm 1 are essential. As Table 3 shows, the full pipeline yields the best tradeoff between privacy (Adv. AUC) and utility (F1 and retrieval-grounded answer rate). Retries are critical for utility. Removing them halves the answer rate and drops F1 substantially. Switching from a randomized to a "deterministic test" (i.e., not applying the geometric noise and ceiling) significantly undermines privacy (raising Adv. AUC from 0.515 to 0.537). Always passing the test increases F1 but also Adv. AUC. Finally, rejecting uniformly at random a fixed percentage of candidate answers ("random rejection") yields both worse privacy (Adv. AUC 0.533) *and* worse utility (F1 0.125). Although "Always Passes" already lowers Adv. AUC to 0.528, this does not mean the PD test serves only the formal $(\varepsilon, \delta)$-DP guarantee (Theorem 1). Within the same candidate-generation pipeline, the test selectively rejects attributable answers: single-document verbatim spans ($\geq$20 tokens) occur in 29.1% of rejected candidates but only 10.9% of accepted ones (Appendix C.14). At a matched retrieval-grounded answer rate it also retains higher utility than random rejection (F1 0.142 vs. 0.125).

Table 4: **Efficiency comparison** (ChatDoctor, Llama-3.1-8B, single A100, original headline config $m$=20, $R$=1).

| Method | Time/query | Slowdown | $\varepsilon$ |
|---|---|---|---|
| Vanilla RAG | 4 s | 1× | $\infty$ |
| **PD-RAG** | **12 s** | **3**× | **5.40** |
| DP-KSA | 198 s | 46× | 8.0 |
| DPSparseVoteRAG | 393 s | 91× | 50 |

Such single-document answers are rare, so they barely affect average-case AUC, but without the test, repeated queries could eventually expose them.

**Efficiency.** Table 4 compares methods in terms of time per query and slowdown. The overhead of the original headline PD-RAG configuration over (non-private) Vanilla RAG is only 3×. PD-RAG is also substantially faster than DP-KSA (17× slower) and DPSparseVoteRAG (33× slower). In PD-RAG the time is dominated by candidate answer generation (only one per attempt), because (as explained in Section 3.1) computing log-likelihood needed for the privacy test (one per group per attempt) is done in one forward pass. By contrast, per-token methods instead invoke the model at every decoding step, leading to thousands of forward passes per query. We provide a detailed complexity breakdown in Appendix C.7. The higher-coverage ChatDoctor configuration requires 18.58 s/query, compared with 12 s/query for the headline configuration; the corresponding QASPER and ASQA tuned configurations require 22.46 and 18.22 s/query (Appendix C.11).

**Higher-coverage operating points.** At the same budget ($\varepsilon \leq 5.40$), jointly tuning $(\gamma, K, m)$ raises the retrieval-grounded answer rate from 20.2/17.3/19.7% to 46.0/34.0/69.5% on ChatDoctor/QASPER/ASQA, with Adv. AUC 0.520–0.527 and nearly unchanged Token-F1 (Appendix C.11).

**Stronger attacks and robustness.** On ChatDoctor, Mask-Based Attack (MBA) [26], Interrogation Attack (IA) [29], and a response-conditioned adaptive IA reach Adv. AUC 0.850–0.922 against Vanilla RAG but only 0.526–0.535 against PD-RAG (Appendix C.15). PD-RAG also stays near random with 500+500 samples (0.515) and with BGE/GTE retrievers (0.519/0.523) despite ≈22pp higher Recall@15 (Appendix C.16), and it scores highest among private methods under four LLM judges (Appendix C.17).

**Additional experiments.** The appendices further report cross-model results (Appendix C.5), pass vs. fallback utility, where passing queries retain 49–76% of Vanilla RAG quality (Appendix C.6), runtime (Appendix C.7), parameter sensitivity (Appendix C.8), answer length (Appendix C.9), hyperparameter search (Appendix C.10), a matched-budget comparison (Appendix C.12), DP-synthetic-corpus baselines (Appendix C.13), and single-source overlap (Appendix C.14).

## 5 Conclusion

This paper introduces PD-RAG, a RAG method that achieves $(\varepsilon, \delta)$-differential privacy through plausible deniability. PD-RAG uses the language model's own sampling randomness alongside with a privacy test to discard candidate answers that violate privacy desiderata. As a result, PD-RAG operates at the answer level, avoiding limitations of existing DP methods for RAG. Experiments across three datasets and three language models show that PD-RAG outperforms all evaluated privacy-preserving methods. Specifically, PD-RAG reduces membership inference attack advantage to near random-guessing levels while providing higher answer quality and also being the fastest method.

**Limitations and future work.** PD-RAG requires access to token-level log-probabilities. These are available for open-weight models, but not for most commercial APIs (e.g., GPT, Claude). Since computing log-probabilities is straightforward and efficient, this functionality could presumably be added. Still, extending PD-RAG to use black-box APIs via surrogate model scoring or other approximations would be an interesting future direction. Another limitation is that PD-RAG's retrieval-grounded answer rate remains relatively low (though it still provides much higher answer quality than alternatives). Potential future work to address this could include domain-specific parameter tuning or improving the privacy analysis to allow for an increased number of retries.

## Acknowledgments and Disclosure of Funding

We thank the anonymous NeurIPS reviewers and area chair for their helpful feedback and suggestions. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

## References

[1] M. Anderson, G. Amit, and A. Goldsteen. Is my data in your retrieval database? membership inference attacks against retrieval augmented generation. *arXiv preprint arXiv:2405.20446*, 2024.

[2] M. Arslan, H. Ghanem, S. Munawar, and C. Cruz. A survey on rag with llms. *Procedia computer science*, 246:3781–3790, 2024.

[3] W. Bao, S. Jin, H. Abdullah, A. C. Nascimento, V. Bindschaedler, and Y. Cai. Deep learning with plausible deniability. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[4] V. Bindschaedler, R. Shokri, and C. A. Gunter. Plausible deniability for privacy-preserving data synthesis. *Proceedings of the VLDB Endowment*, 10(5), 2017.

[5] V.-E. Brunel and M. Avella-Medina. Propose, test, release: Differentially private estimation with high probability. *arXiv preprint arXiv:2002.08774*, 2020.

[6] Y. Chen and A. Machanavajjhala. On the privacy properties of variants on the sparse vector technique. *arXiv preprint arXiv:1508.07306*, 2015.

[7] P. Dasigi, K. Lo, I. Beltagy, A. Cohan, N. A. Smith, and M. Gardner. A dataset of information-seeking questions and answers anchored in research papers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4599–4610, 2021.

[8] W. Du, Y. Yang, and S. Welleck. Optimizing temperature for language models with multi-sample inference. *arXiv preprint arXiv:2502.05234*, 2025.

[9] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[10] C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and trends® in theoretical computer science*, 9(3-4):211–487, 2014.

[11] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

[12] A. Ghosh, T. Roughgarden, and M. Sundararajan. Universally utility-maximizing privacy mechanisms. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 351–360, 2009.

[13] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[14] N. Grislain. Rag with differential privacy. In *2025 IEEE Conference on Artificial Intelligence (CAI)*, pages 847–852. IEEE, 2025.

[15] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[16] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[17] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus. *IEEE transactions on big data*, 7(3):535–547, 2019.

[18] P. Kairouz, S. Oh, and P. Viswanath. The composition theorem for differential privacy. In *International conference on machine learning*, pages 1376–1385. PMLR, 2015.

[19] H. Kaplan, Y. Mansour, and U. Stemmer. The sparse vector technique, revisited. In *Conference on Learning Theory*, pages 2747–2776. PMLR, 2021.

[20] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 6769–6781, 2020.

[21] T. Koga, R. Wu, Z. Zhang, and K. Chaudhuri. Privacy-preserving retrieval-augmented generation with differential privacy. *arXiv preprint arXiv:2412.04697*, 2024.

[22] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

[23] Y. Li, Z. Li, K. Zhang, R. Dan, S. Jiang, and Y. Zhang. Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge. *Cureus*, 15(6), 2023.

[24] Y. Li, G. Liu, C. Wang, and Y. Yang. Generating is believing: Membership inference attacks against retrieval-augmented generation. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[25] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[26] M. Liu, S. Zhang, and C. Long. Mask-based membership inference attacks for retrieval-augmented generation. In *Proceedings of the ACM on Web Conference 2025*, pages 2894–2907, 2025.

[27] M. Lyu, D. Su, and N. Li. Understanding the sparse vector technique for differential privacy. *Proceedings of the VLDB Endowment*, 10(6), 2017.

[28] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 94–103. IEEE, 2007.

[29] A. Naseh, Y. Peng, A. Suri, H. Chaudhari, A. Oprea, and A. Houmansadr. Riddle me this! stealthy membership inference for retrieval-augmented generation. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*, pages 1245–1259, 2025.

[30] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[31] N. Pipitone and G. H. Alami. Legalbench-rag: A benchmark for retrieval-augmented generation in the legal domain. *arXiv preprint arXiv:2408.10343*, 2024.

[32] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 2383–2392, 2016.

[33] M. Renze. The effect of sampling temperature on problem solving in large language models. In *Findings of the association for computational linguistics: EMNLP 2024*, pages 7346–7356, 2024.

[34] I. Stelmakh, Y. Luan, B. Dhingra, and M.-W. Chang. Asqa: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, 2022.

[35] T. Tang, J. Flemings, Y. Wang, and M. Annavaram. Differentially private retrieval-augmented generation. *arXiv preprint arXiv:2602.14374*, 2026.

[36] G. Wang, J. He, H. Li, M. Zhang, and D. Feng. Rag-leaks: difficulty-calibrated membership inference attacks on retrieval-augmented generation. *Science China Information Sciences*, 68(6):160102, 2025.

[37] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020.

[38] R. Wu, E. Wang, and Y.-X. Wang. Beyond per-question privacy: Multi-query differential privacy for RAG systems. In *NeurIPS 2025 Workshop: Reliable ML from Unreliable Data*, 2025.

[39] S. Xiao, Z. Liu, P. Zhang, N. Muennighoff, D. Lian, and J.-Y. Nie. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 641–649, 2024.

[40] C. Xie, Z. Lin, A. Backurs, S. Gopi, D. Yu, H. A. Inan, H. Nori, H. Jiang, H. Zhang, Y. T. Lee, B. Li, and S. Yekhanin. Differentially private synthetic data via foundation model APIs 2: Text. In *Forty-first International Conference on Machine Learning*, 2024.

[41] A. Xu, T. Yu, M. Du, P. Gundecha, Y. Guo, X. Zhu, M. Wang, P. Li, and X. Chen. Generative ai and retrieval-augmented generation (rag) systems for enterprise. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 5599–5602, 2024.

[42] P. Yang, J. Yin, H. Zheng, X. Bai, H. Wang, Y. Sun, X. Li, S. Pei, Y. Huang, and T. Qi. Mrm: Black-box membership inference attacks against multimodal rag systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 34295–34303, 2026.

[43] D. Yu, S. Naik, A. Backurs, S. Gopi, H. A. Inan, G. Kamath, J. Kulkarni, Y. T. Lee, A. Manoel, L. Wutschitz, S. Yekhanin, and H. Zhang. Differentially private fine-tuning of language models. *Journal of Privacy and Confidentiality*, 14(2), 2024.

[44] X. Yue, H. Inan, X. Li, G. Kumar, J. McAnallen, H. Shajari, H. Sun, D. Levitan, and R. Sim. Synthetic text generation with differential privacy: A simple and practical recipe. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1321–1342, 2023.

[45] X. Zhang, Y. Zhang, D. Long, W. Xie, Z. Dai, J. Tang, H. Lin, B. Yang, P. Xie, F. Huang, et al. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 conference on empirical methods in natural language processing: industry track*, pages 1393–1412, 2024.

[46] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

[47] Y. Zhu, J. Li, G. Li, Y. Zhao, J. Li, Z. Jin, and H. Mei. Hot or cold? adaptive temperature sampling for code generation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 437–445, 2024.

[48] Y. Zhu and Y.-X. Wang. Improving sparse vector technique with renyi differential privacy. *Advances in neural information processing systems*, 33:20249–20258, 2020.

# A Notation

Table 5: Notation for PD-RAG.

| Symbol | Meaning |
|---|---|
| $D$ | Private document corpus |
| $N$ | Total number of documents retrieved (multiple of $m$) |
| $G_1, \ldots, G_m$ | Disjoint document groups |
| $\mathfrak{G}$ | Partition $(G_1, \ldots, G_m)$ of retrieved docs ($\forall i \; \lvert G_i \rvert = N/m$) |
| $\alpha$ | Similarity factor ($\alpha \geq 1$) |
| $\gamma$ | Log-threshold ($\gamma = \log \alpha$) |
| $K$ | Count threshold (min. similar groups to pass) |
| $T$ | Model softmax temperature |
| $R$ | Max retry attempts |
| $y$ | Selected candidate answer |
| $n$ | Length of $y$ in tokens |
| $\tau$ | Count of $\alpha$-similar groups |
| $\beta$ | Geometric noise parameter ($\beta$) |
| $\varepsilon_0$ | Privacy loss of passing the test ($\varepsilon_0 = \log \beta$) |
| $\psi$ | One minus ceiling probability |
| $\zeta$ | Rejection probability ratio bound ($\zeta = (\beta+\psi)/(1+\beta\psi)$) |

# B Privacy Analysis and Differential Privacy Guarantees

In this section, we expand on the privacy analysis for PD-RAG Algorithm 1 and prove that it satisfies $(\varepsilon, \delta)$-differential privacy in the bounded neighbors setting.

The data-independent fallback answer can be viewed as post-processing that can never increase privacy loss, we can think of the range of PD-RAG as $\Omega \cup \{\perp\}$, where $\perp$ denotes a rejection (or all attempts) and $\Omega$ is the set of all answers of length at most $n_{\max}$ tokens, where each token belongs to $V$, the language model's vocabulary.

We prove the per-attempt $(\varepsilon, \delta)$-DP bound (Theorem 1) for both outputs in $\Omega$ and $\perp$-outputs. Finally, we expand the discussion (of Section 3.4) of composition across retries.

## B.1 Setup and Grouping Bijection

Consider two arbitrary neighboring corpora $D$ and $D'$ that differ in exactly one document: there exist $d \neq d'$ such that $d = D \setminus D'$ and $d' = D' \setminus D$ (document substitution). We assume that the retriever's top-$N$ relevant document set only changes by at most one document under this substitution. This is the case for any retriever that scores each document independently such as deterministic dense retrievers (e.g., DPR).

Algorithm 1 first retrieves retrieves $N$ documents from its input corpus. Let $\mathcal{F}$ denote one attempt of Algorithm 1. This mechanism randomly partitions retrieved documents into $m$ ($\geq 2$) equal-sized groups $\mathfrak{G} = (G_1, \ldots, G_m)$, picks a seed uniformly, generates an answer, and runs the privacy test.

Notably, each partition/grouping is equally likely to be observed and the order of documents within a group does not matter. Observe that for every grouping $\mathfrak{G}$ under $D$, there is a unique grouping $\mathfrak{G}' = (G'_1, \ldots, G'_m)$ under $D'$ where $d$ is substituted by $d'$ in exactly one group $G_j$ (for some $j$). That is: $G_i = G'_i$ for $i \neq j$ and $G'_j = (G_j \setminus \{d\}) \cup \{d'\}$. This bijective mapping allows us to translate conditional bounds to unconditional bounds: if for every paired $(\mathfrak{G}, \mathfrak{G}')$ and any measurable output set $S$,

$$\Pr[\mathcal{F}(D) \in S \mid \mathfrak{G}] \leq e^{\varepsilon} \Pr[\mathcal{F}(D') \in S \mid \mathfrak{G}'] + \delta \,,$$

then $\Pr[\mathcal{F}(D) \in S] \leq e^{\varepsilon} \Pr[\mathcal{F}(D') \in S] + \delta$.

This means it suffices to prove Theorem 1 for an arbitrary pair of matching groupings $(\mathfrak{G}, \mathfrak{G}')$.

For the proof, we will reason about the probability that a candidate answer $y$ is produced (and passes the test and is released) from $\mathfrak{G}$ as opposed to $\mathfrak{G}'$. Write

$$q(y, \mathfrak{G}) = \sum_{i=1}^{m} p_{G_i}(y) \cdot p_{\tau(y, G_i, \mathfrak{G})} \,.$$

Here $p_G(y) = P_{\mathcal{M}}(y \mid Q, G)$. Let $\tau(y, G_i, \mathfrak{G})$ be the plausible alternative count for $y$, group $G_i$ and partitions $\mathfrak{G}$. With slight abuse of notation, we write $p_{\tau(c)} = (1 - \psi) \Pr[Z \geq K - c]$ to denote the pass probability with $c$ alternatives, interchangeably denoting the number of alternatives as $c$ or $\tau(y, G_i, \mathfrak{G})$. Recall that the probability of passing the test depends on the number of alternatives and that $Z \sim \mathrm{Geom}(\beta)$ is symmetric geometric noise on $\mathbb{Z}$ with $\beta = e^{\varepsilon_0} > 1$.

In other words, for $y$ to be released, it must first be generated by the LM (from some group $G_i$) and then the privacy test must pass.

The probability that the mechanism outputs $y$ (conditional on grouping $\mathfrak{G}$) is:

$$\Pr[\mathcal{F}(D) = y \mid \mathfrak{G}] = \frac{1}{m} q(y, \mathfrak{G}) \,,$$

since the seed group is selected uniformly at random.

We will make use of the following Lemmas about the probability of passing/failing the privacy test and how it changes when the number of alternatives increases/decreases by one. These results follow existing work on plausible deniability [3], but we restate them here and provide proofs to ensure this section is self-contained.

**Lemma 1.** *Let $p_{\tau(c)} = (1 - \psi) \Pr[Z \geq K - c]$ denote the probability of passing the test with $c \geq 0$ alternatives, with $Z \sim \mathrm{Geom}(\beta)$. We have:*

$$1 \leq \frac{p_{\tau(c+1)}}{p_{\tau(c)}} \leq \beta \,,$$

*and for integer $1 \leq t < K$:*

$$p_{\tau(t)} = (1 - \psi) \Pr[Z \geq K - t] \leq (1 - \psi) \frac{\beta}{\beta + 1} \beta^{-(K-t)} \,.$$

*Proof of Lemma 1.* For the first part we have that:

$$\Pr[Z \geq K - (c + 1)] = \Pr[Z = K - (c + 1)] + \Pr[Z \geq K - c] \,,$$

so $p_{\tau(c+1)} \geq p_{\tau(c)}$. For $c < K$, $\Pr[Z \geq K - c] = \frac{\beta - 1}{\beta + 1} \sum_{i \geq K - c} \beta^{-i} = \frac{\beta}{\beta + 1} \beta^{-(K-c)}$. Therefore:

$$\frac{p_{\tau(c+1)}}{p_{\tau(c)}} = \frac{\Pr[Z \geq K - c - 1]}{\Pr[Z \geq K - c]} = \frac{\beta^{-(K-c-2)}}{\beta^{-(K-c-1)}} = \beta \,.$$

where we used $\Pr[Z \geq k] = \frac{\beta - 1}{\beta + 1} \cdot \frac{\beta^{-k}}{1 - \beta^{-1}} = \frac{\beta^{-(k-1)}}{\beta + 1}$ for $k \geq 1$ (geometric series sum).

For $c \geq K$, both numerator and denominator involve probabilities over the full distribution (including negative values), and the ratio approaches 1 from above. Hence $1 \leq p_{\tau(c+1)} / p_{\tau(c)} \leq \beta$.

For the second part:

$$p_{\tau(t)} = (1 - \psi) \Pr[Z \geq K - t] = (1 - \psi) \frac{\beta^{-(K-t-1)}}{\beta + 1} = (1 - \psi) \frac{\beta}{\beta + 1} \beta^{-(K-t)} \,.$$

□

Lemma 1 shows that adding/removing a single plausible alternative does not change the probability of passing the test by more than a factor of $\beta \geq 1$, and that the probability of passing the test with only $t < K$ alternatives is exponentially small as function of $(K - t)$.

We also need an analogous results for the probability of failing to pass the test.

**Lemma 2.** *Let $p_{\perp(c)} = 1 - p_{\tau(c)}$ denote the probability of failing the test with $c \geq 0$ plausible alternatives. We have:*

$$1 \leq \frac{p_{\perp(c)}}{p_{\perp(c+1)}} \leq \zeta \,,$$

*where $\zeta = \frac{\beta+\psi}{1+\beta\psi} > 1$ and $\zeta < \beta$ whenever $\psi > 0$.*

*Proof of Lemma 2.* By definition, $p_{\perp(c)} = 1 - p_{\tau(c)} = \psi + (1-\psi)\Pr[Z < K - c]$. Using $\Pr[Z < k] = 1 - \frac{\beta^{-(k-1)}}{\beta+1}$ for $k \geq 1$:

$$\frac{p_{\perp(c)}}{p_{\perp(c+1)}} = \frac{\psi + (1-\psi)\Pr[Z < K - c]}{\psi + (1-\psi)\Pr[Z < K - c - 1]} \,.$$

Since $\Pr[Z < K - c] \geq \Pr[Z < K - c - 1]$, the ratio is at least 1. For the upper bound, the idea is to bound the ratio over $c$, observing that the maximum occurs for $c = K - 1$. Using $\Pr[Z < k] - \Pr[Z < k-1] = \Pr[Z = k-1] = \frac{\beta-1}{\beta+1}\beta^{-(k-1)}$, we see that $p_{\perp(c)}/p_{\perp(c+1)} \leq \zeta$ where $\zeta = (\beta+\psi)/(1+\beta\psi)$. Finally, $\zeta < \beta$ whenever $\psi > 0$ because $\beta+\psi < \beta+\beta^2\psi \iff 1 < \beta$. $\square$

We are now in a position to prove Theorem 1.

**Theorem 1** (Per-Attempt Privacy Budget)**.** *For any two neighboring corpora $D, D'$ that differ in exactly one document and any integer $1 \leq t < K$, $\mathcal{F}$ satisfies differential privacy, i.e.:*

$$\Pr(\mathcal{F}(D) \in S) \leq e^{\varepsilon(t)} \Pr(\mathcal{F}(D') \in S) + \delta(t) \,,$$

*where $\varepsilon(t)$ and $\delta(t)$ satisfy the following.*

*For any output set $S \subseteq \Omega$:*

$$\varepsilon(t) = \varepsilon_{\text{pass}}(t) = \log\beta + \log\left(1 + \frac{e^{\gamma}}{t}\right) \,, \qquad \delta(t) = \frac{(1-\psi)\,\beta^{-(K-t)}}{(1+\beta^{-1})\,m} \,. \tag{7}$$

*For $S = \{\perp\}$ (failure to pass):*

$$\varepsilon = \varepsilon_{\perp} = \log\Big(\zeta\big(1 - \tfrac{1}{m}\big) + \tfrac{1}{m\psi}\Big) \,, \qquad \delta = \delta_{\perp} = 0 \,. \tag{8}$$

*with $\zeta = \frac{\beta+\psi}{1+\beta\psi}$ (and $\zeta \leq \beta$ whenever $\psi > 0$).*

### B.2 Passing Case Bound (Non-$\perp$ Output)

We show that for any $y \in \Omega$ and integer $1 \leq t < K$: $q(y, \mathfrak{G}) \leq e^{\varepsilon(t)} q(y, \mathfrak{G}') + \delta_y(t)$.

Decompose over the differing group $G_j \neq G'_j$. We have: $q(y, \mathfrak{G}) = q_{-j}(y, \mathfrak{G}) + p_{G_j}(y) \cdot p_{\tau(y, G_j, \mathfrak{G})}$, where $q_{-j}$ excludes the $j$-th term. For $i \neq j$, $G_i = G'_i$ and $|\tau(y, G_i, \mathfrak{G}) - \tau(y, G_i, \mathfrak{G}')| \leq 1$ (the count of alternatives differs by at most one). Therefore by Lemma 1:

$$\beta^{-1} \leq q_{-j}(y, \mathfrak{G}) / q_{-j}(y, \mathfrak{G}') \leq \beta \,.$$

To deal with the remaining term, we split into two cases depending on whether the number of alternatives for $G_j$ exceeds $t$.

Case 1: $\tau(y, G_j, \mathfrak{G}) > t$. By the definition of the alternative counts (Eq. (6)), if $\tau(y, G_j, \mathfrak{G}) > t$, then there must exist a set $\mathfrak{G}^{(t)}$ of at least $t$ groups in $\mathfrak{G} \setminus \{G_j\}$ sharing the same bin as $G_j$. For each such group $G \in \mathfrak{G}^{(t)}$: $p_{G_j}(y) \leq \alpha \cdot p_G(y)$, and $p_{\tau(y, G_j, \mathfrak{G})} = p_{\tau(y, G, \mathfrak{G})}$ since the probability of passing the test depends only the bin/number of alternatives. Therefore:

$$p_{G_j}(y) \cdot p_{\tau(y, G_j, \mathfrak{G})} \leq \frac{\alpha}{t} \sum_{G \in \mathfrak{G}^{(t)}} p_G(y) \cdot p_{\tau(y, G, \mathfrak{G})} \leq \frac{\alpha}{t}\, q_{-j}(y, \mathfrak{G}) \,.$$

Combining this with $q_{-j}(y, \mathfrak{G}) \leq \beta\, q_{-j}(y, \mathfrak{G}')$, we obtain that

$$q(y, \mathfrak{G}) \leq \beta\Big(1 + \frac{\alpha}{t}\Big)\, q(y, \mathfrak{G}') \,.$$

Case 2: $\tau(y, G_j, \mathfrak{G}) \le t$. In this case, there may not be sufficiently many groups in $\mathfrak{G} \setminus \{G_j\}$ sharing the same bin as $G_j$. However, the probability of passing the test itself with no more than $t < K$ alternatives is bounded (Lemma 1): $p_{\tau(y,G_j,\mathfrak{G})} \le (1-\psi) \cdot \frac{\beta}{\beta+1} \cdot \beta^{-(K-t)}$. Therefore:

$$q(y, \mathfrak{G}) \le \beta\, q_{-j}(y, \mathfrak{G}') + \delta_y(t)\,,$$

where $\delta_y(t) = p_{G_j}(y)\,(1-\psi)\,\frac{\beta}{\beta+1}\,\beta^{-(K-t)}$.

**Combining both cases.** Taking the worst case over the two cases: $q(y, \mathfrak{G}) \le e^{\varepsilon(t)} q(y, \mathfrak{G}') + \delta_y(t)$, where $\varepsilon(t) = \log(\beta(1 + \alpha/t))$.

Recall that for any group $G$, query $Q$ and subset $Y \subseteq \Omega$: $\sum_{y\in Y} p_G(y) = \sum_{y\in Y} P_{\mathcal{M}}(y \mid Q, G) \le 1$. Taking the sum over $y \in Y$ and scaling by $1/m$ we obtain:

$$\Pr[\mathcal{F}(D) \in Y \mid \mathfrak{G}] \le e^{\varepsilon(t)} \Pr[\mathcal{F}(D') \in Y \mid \mathfrak{G}'] + \delta(t)\,,$$

where $\varepsilon(t) = \log(\beta(1 + \alpha/t))$ and $\delta(t) = \frac{1}{m}\,\frac{(1-\psi)\beta}{\beta+1}\,\beta^{-(K-t)} = \frac{(1-\psi)\beta^{-(K-t)}}{(1+\beta^{-1})m}$, which completes (this part of) the result.

### B.3 Rejection Case Bound ($\perp$ Output)

With slight abuse of notation, we write $q(\perp, \mathfrak{G}) = \sum_y \sum_i p_{G_i}(y) \cdot p_{\perp(y,G_i,\mathfrak{G})}$. The probability of outputting $\perp$ conditional on $\mathfrak{G}$ is $\Pr[\mathcal{F}(D) = \perp \mid \mathfrak{G}] = \frac{1}{m} q(\perp, \mathfrak{G})$.

The approach for this case is to use Lemma 2 and bound the ratio $q(\perp, \mathfrak{G})/q(\perp, \mathfrak{G}')$ over the probability of failing the test.

Consider the ratio:

$$\frac{q(\perp, \mathfrak{G})}{q(\perp, \mathfrak{G}')} = \frac{q_{-j}(\perp, \mathfrak{G}) + \sum_y p_{G_j}(y) p_{\perp(y,G_j,\mathfrak{G})}}{q_{-j}(\perp, \mathfrak{G}') + \sum_y p_{G'_j}(y) p_{\perp(y,G'_j,\mathfrak{G}')}}$$

This ratio is maximized when the numerator term involving $G_j$ is maximized ($\sum_y p_{G_j}(y) p_{\perp(y,G_j,\mathfrak{G})} \le 1$) and the denominator term involving $G'_j$ is minimized ($\sum_y p_{G'_j}(y) p_{\perp(y,G'_j,\mathfrak{G}')} \ge \psi$). Using Lemma 2 we have that for $i \ne j$: $\zeta^{-1} \le p_{\perp(y,G_i,\mathfrak{G})}/p_{\perp(y,G_i,\mathfrak{G}')} \le \zeta$. Bounding the $i \ne j$ terms by $\zeta$, we obtain:

$$\frac{q(\perp, \mathfrak{G})}{q(\perp, \mathfrak{G}')} \le \frac{\zeta(m-1)x + 1}{(m-1)x + \psi}\,,$$

where $x$ is the average failure probability over non-$j$ groups under $\mathfrak{G}'$ ($\psi \le x < 1$). Since $\zeta\psi = \psi(\beta + \psi)/(1 + \beta\psi) < 1$ whenever $\psi < 1$ (always), the ratio is decreasing as a function of $x$ and reaches its maximum value at $x = \psi$:

$$\frac{q(\perp, \mathfrak{G})}{q(\perp, \mathfrak{G}')} \le \zeta\left(1 - \frac{1}{m}\right) + \frac{1}{m\psi}\,.$$

Since $\Pr[\mathcal{F}(D) = \perp \mid \mathfrak{G}] = \frac{1}{m} q(\perp, \mathfrak{G})$, we get that the $\perp$-output case satisfies $(\varepsilon, \delta)$-DP as required, for $\varepsilon_\perp = \log\left(\zeta\left(1 - \frac{1}{m}\right) + \frac{1}{m\psi}\right)$ and $\delta = 0$.

**Combined Guarantee.** Combining the two cases above completes the proof of Theorem 1.

## C Experimental Details

### C.1 Datasets

**ChatDoctor [23].** We use the HealthCareMagic-100k subset, consisting of 112k doctor–patient dialogue pairs. Each entry contains a patient question and a detailed doctor response (average length ≈80 tokens). We build a 50k-document RAG corpus by shuffling all entries and using the first half; the second half serves as non-members. The FAISS [17] index uses DPR [20] (768-dimensional inner-product search).

Table 6: **Dataset statistics.** Baseline AUC: Vanilla RAG attack AUC (higher = more vulnerable).

| Dataset | Domain | Avg. tokens | Corpus | Baseline AUC |
|---|---|---|---|---|
| ChatDoctor | Medical | ≈80 | 50k | 0.823 |
| QASPER | Scientific | ≈50–300 | ≈1.5k | 0.931 |
| ASQA | General | ≈220 | ≈2.7k | 0.985 |

**QASPER [7].** A dataset of 5,049 questions over 1,169 NLP research papers (CC BY 4.0 license). Each question is written by an NLP practitioner who read only the paper's title and abstract. Answers are provided by separate annotators with access to the full text. We use only entries with non-empty `free_form_answer` ($\geq$10 words), yielding ≈3,000 QA pairs. The corpus is built from the first half of the shuffled QA pairs (≈1,500 entries, "Question: ... Answer: ..." format); the second half serves as non-members. QASPER exhibits the strongest MIA signal (baseline AUC 0.931), as academic paper QA requires highly specific domain knowledge.

**ASQA [34].** The Ambiguous Short-form QA dataset contains 4,353 training and 948 development entries (MIT license). Each entry has an ambiguous factoid question and a long-form answer (≈170 words) synthesized from multiple Wikipedia passages. We merge train and dev splits, shuffle, and use the first half for corpus construction. ASQA's Wikipedia-based content is naturally compatible with DPR retrieval.

**Member/non-member split.** For all datasets, we follow a consistent protocol: shuffle the full dataset with a fixed seed, assign the first half to the corpus (members), and hold out the second half (non-members). For each experiment, we sample $N$ members and $N$ non-members from these pools. This ensures no overlap between member and non-member sets.

### C.2 Models and Infrastructure

- **Llama-3.1-8B** [13]: Meta's 8B base model. Used as the *primary model* for all main results, ablation studies, and DP RAG methods comparisons.
- **Mistral-7B-v0.1** [16]: Mistral AI's 7B base model. Used for cross-model validation (Appendix C.5) to demonstrate that PD-RAG generalizes across model families.
- **Llama-3.2-3B-Instruct** [13]: Meta's 3B instruction-tuned model. Used for cross-model validation (Appendix C.5) to demonstrate that PD-RAG generalizes across model scales and model types (base vs. instruction-tuned).

All comparisons between PD-RAG and other baselines use the same model to ensure fairness.

**Retriever.** We use DPR (Dense Passage Retriever) [20] with separate question and context encoders (768-dimensional). The FAISS index uses flat inner-product search.

**Generation.** Unless otherwise specified, all generation uses HuggingFace Transformers [37] with `max_new_tokens` $= 128$ and `torch.float16` precision.

### C.3 Attack Configuration

We use $S^2$MIA (Semantic Similarity Membership Inference Attack) [24]. In $S^2$MIA, the attacker computes the similarity between the RAG answer and the known gold answer for each query, then uses the similarity score to distinguish members from non-members.

**Similarity metric.** We use BLEU [30], following [21]. ChatDoctor answers are long (on average ≈ 80 tokens), so $n$-gram overlap is a reasonable measure of answer-gold similarity.

**Evaluation protocol.** For each experiment, we sample $N$ member queries (whose gold-answer is in the corpus) and $N$ non-member queries (whose gold-answer is not in the corpus). We compute ROC AUC over the combined similarity scores. We report two privacy metrics: TPR@1%FPR and Advantage AUC $= \max(\text{AUC}, 1-\text{AUC})$. The Advantage AUC corrects for inverted scores that arise when the defense makes member answers substantially *worse* than non-member answers (e.g., due to data-independent fallback).

### C.4 Original Headline Configurations

Table 7 reports the per-dataset headline PD-RAG configurations used in Table 2. Privacy budgets are computed via Theorem 1 with $t$=1; the total budget with $R$ retries is $\varepsilon_{\text{total}} = R\,\varepsilon_{\perp} + \varepsilon_{\text{pass}}$ and $\delta_{\text{total}} = \delta$.

Table 7: **Original per-dataset headline PD-RAG configurations.** All yield $\varepsilon_{\text{total}} \leq 5.40$ and $\delta_{\text{total}} = 5.1 \times 10^{-7}$, well below DP-KSA ($\varepsilon$=8, $\delta = 10^{-5}$).

| Dataset | $\gamma$ | $K$ | $m$ | $N/m$ | $\beta$ | $\psi$ | $R$ | $\varepsilon_{\text{total}}$ | $\delta_{\text{total}}$ | Answer Rate |
|---|---|---|---|---|---|---|---|---|---|---|
| ChatDoctor | 2.1 | 8 | 20 | 1 | 5 | 0.05 | 1 | 5.40 | $5.1 \times 10^{-7}$ | 20.2%±3.1% |
| QASPER | 2.0 | 8 | 20 | 3 | 5 | 0.05 | 1 | 5.31 | $5.1 \times 10^{-7}$ | 17.3%±0.6% |
| ASQA | 2.0 | 8 | 20 | 3 | 5 | 0.05 | 1 | 5.31 | $5.1 \times 10^{-7}$ | 19.7%±1.3% |

All configurations share $K$=8, $\beta$=5 ($\varepsilon_0 = \log 5 \approx 1.61$), $m$=20, $\psi$=0.05, and $R$=1. The $\perp$ privacy bound is $\varepsilon_{\perp} \approx 1.58$ (Eq. (8)). ChatDoctor uses $\gamma$=2.1 with $N/m$=1 ($\varepsilon_{\text{total}}$=5.40). QASPER and ASQA share $\gamma$=2.0 with $N/m$=3 ($\varepsilon_{\text{total}}$=5.31). All configurations yield $\delta_{\text{total}} = 5.1 \times 10^{-7}$, well below DP-KSA ($\varepsilon$=8, $\delta = 10^{-5}$).

### C.5 Cross-Model Validation

Experiments in the main text use Llama-3.1-8B as LM. Here we additionally evaluate PD-RAG using Mistral-7B-v0.1 (a different model family) and Llama-3.2-3B-Instruct (a smaller, instruction-tuned model) to understand cross-model generalization.

Table 8 shows the results, which indicate that PD-RAG generalizes across model families, scales, and types (base vs. instruction-tuned).

**Mistral-7B.** Mistral-7B vanilla RAG exhibits comparable vulnerability to Llama-3.1-8B (Adv. AUC 0.813–0.977 across datasets), suggesting that membership leakage is not model-specific. PD-RAG reduces Adv. AUC to $\leq 0.537$ on all datasets with TPR@1% $\leq 0.030$, consistent with the 8B results. Utility metrics (RL, R1, F1) follow a similar pattern to the original headline configurations: PD-RAG preserves between 36 and 43% of Vanilla RAG's ROUGE-L while reducing privacy leakage to near random guessing.

**Llama-3.2-3B-Instruct.** This 3B model exhibits weaker vanilla baselines (e.g., ChatDoctor Adv. AUC 0.567 vs. 0.800 for 8B). PD-RAG still reduces Adv. AUC to near random guessing (0.520–0.551). However, utility suffers on specialized domains: QASPER ROUGE-L drops to 0.059 (vs. 0.075 for Mistral-7B). We suspect this is because the 3B model generates less consistent log-probabilities across groups, causing a lower PD answer rate. These results suggests that smaller models may benefit from looser parameters in specialized domains.

### C.6 Pass vs. Fallback Utility Decomposition and Data-Independent Generation

When the PD test fails after $R$+1 attempts, PD-RAG falls back to data-independent generation (no retrieval). With original headline answer rates of 17–20%, roughly 80% of queries use the fallback path. Note that other DP method like DP-KSA [35] only has limited key words generated after PTR, providing insufficient context for quality generation—effectively reducing to data-independent generation in most cases. A natural question is: *how much utility does PD-RAG retain on queries that actually pass?*

Table 9 decomposes PD-RAG's utility into passed and fallback cases. Two findings stand out in particular. First, **passed queries retain 49–76% of Vanilla RAG quality**, showing that the PD privacy mechanism does *not* substantially degrade answer quality when it accepts. Second, **fallback RL closely matches the Non-RAG baseline** (e.g., ASQA: 0.169 vs. 0.170), confirming that fallback is essentially data-independent generation. These results indicate that the utility gap between PD-RAG and Vanilla RAG is dominated by the fallback rate. Therefore, a potential path to narrow or close this gap could be to relax some privacy parameters (e.g., $\gamma$), perform some domain-specific tuning, or even increase the number of retries.

Table 8: **Cross-model validation** PD-RAG with parameters $\gamma$=2.0, $\beta$=5, $\psi$=0.05, $K$=2, $m$=5, $N/m$=3, $R$=3.

| Model | Dataset | Method | Adv. AUC ↓ | TPR@1% ↓ | RL ↑ | R1 ↑ | F1 ↑ |
|---|---|---|---|---|---|---|---|
| Mistral-7B | ChatDoctor | Vanilla RAG | 0.813 | 0.570 | 0.344 | 0.462 | 0.294 |
| | | PD-RAG | 0.530 | 0.030 | 0.141 | 0.276 | 0.147 |
| | QASPER | Vanilla RAG | 0.899 | 0.840 | 0.208 | 0.531 | 0.174 |
| | | PD-RAG | 0.537 | 0.020 | 0.075 | 0.214 | 0.065 |
| | ASQA | Vanilla RAG | 0.977 | 0.840 | 0.506 | 0.702 | 0.376 |
| | | PD-RAG | 0.537 | 0.000 | 0.217 | 0.404 | 0.167 |
| Llama-3.2-3B | ChatDoctor | Vanilla RAG | 0.567 | 0.290 | 0.169 | 0.280 | 0.178 |
| | | PD-RAG | 0.551 | 0.030 | 0.099 | 0.197 | 0.122 |
| | QASPER | Vanilla RAG | 0.832 | 0.500 | 0.193 | 0.483 | 0.169 |
| | | PD-RAG | 0.520 | 0.040 | 0.059 | 0.138 | 0.057 |
| | ASQA | Vanilla RAG | 0.691 | 0.540 | 0.398 | 0.558 | 0.313 |
| | | PD-RAG | 0.541 | 0.000 | 0.124 | 0.221 | 0.104 |

Table 9: **Utility decomposition: passed vs. fallback queries** (Llama-3.1-8B, original headline configurations). Passed: queries where the PD test accepts the RAG answer. Fallback: queries answered by data-independent generation. Retention: Passed RL / Vanilla RL.

| Dataset | Ans. Rate | Passed RL | Fallback RL | Non-RAG RL | Vanilla RL | Retention |
|---|---|---|---|---|---|---|
| ChatDoctor | 20.2% | 0.170 | 0.105 | 0.115 | 0.346 | 49% |
| QASPER | 17.3% | 0.155 | 0.052 | 0.065 | 0.204 | 76% |
| ASQA | 18.8% | 0.302 | 0.169 | 0.170 | 0.494 | 61% |

### C.7 Computational Efficiency

A notable feature of PD-RAG is that is drastically more efficient than other DP RAG methods and this is clear in runtime experiments (Table 4).

**Complexity breakdown.** Each attempt of PD-RAG (Algorithm 1) first produces one candidate answer $y$ from the seed group. Then, the algorithm computes the log-likelihoods $l_j = \log P_{\mathcal{M}}(y \mid Q, G_j)$ for all $m$ groups. However, the latter step is much cheaper than the former. This is because once the candidate answer $y$ is known, we concatenate $Q, G_j, y$ into a single token sequence and obtain the logits (from which the sum of log of softmax gives the log-likelihoods) in one forward pass per group. Sampling the candidate answer, however, requires each of the $n$ tokens to be produced sequentially.

In experiments, we find that the cost of computing the log-likelihood for one group is approximately one-tenth of the cost of generating the candidate answer. Therefore the total cost of running PD-RAG is approximately $(R{+}1)(1 + \frac{m}{10})$ the cost of one answer generation. To make this concrete, Table 10 reports wall-clock time per query. PD-RAG ($m$=5, $R$=2) incurs only $\approx$3$\times$ overhead across all three models.

The table also shows the impact of $\gamma$ on runtime. Increasing $\gamma$ slightly reduces runtime (13.7 s at $\gamma$=1.0 to 11.0 s at $\gamma$=5.0) because looser similarity bounds allow more queries to pass on the first attempt, avoiding retries.

**Comparison to other DP RAG methods.** By contrast, DPSparseVoteRAG requires $n_{\text{split}} \times n = 50 \times 128 = 6{,}400$ forward passes per query (one per token per split), and DP-KSA requires multiple LM calls for keyword extraction, PTR noise testing, and final zero-shot generation. PD-RAG's cost is three orders of magnitude fewer model invocations than DPSparseVoteRAG.

Table 10: **Wall-clock time per query** (ChatDoctor, single A100 GPU, ablation config $m$=5, $R$=2). Top: cross-model comparison. Bottom: $\gamma$ effect on latency (Llama-3.1-8B).

| **Method** | **Model** | **Time/query** | **Slowdown** | **RL** ↑ |
|---|---|---|---|---|
| Vanilla RAG | Llama-3.1-8B | 4.0 s | 1× | 0.333 |
| PD-RAG ($\gamma$=2.0) | Llama-3.1-8B | 12.2 s | 3× | 0.135 |
| Vanilla RAG | Mistral-7B | 5.0 s | 1× | 0.314 |
| PD-RAG ($\gamma$=2.0) | Mistral-7B | 14.3 s | 3× | 0.120 |
| Vanilla RAG | Llama-3.2-3B | 3.6 s | 1× | 0.169 |
| PD-RAG ($\gamma$=2.0) | Llama-3.2-3B | 11.6 s | 3× | 0.130 |
| *$\gamma$ effect on running time (Llama-3.1-8B):* | | | | |
| PD-RAG ($\gamma$=1.0) | Llama-3.1-8B | 13.7 s | 3.4× | 0.130 |
| PD-RAG ($\gamma$=2.0) | Llama-3.1-8B | 12.2 s | 3× | 0.135 |
| PD-RAG ($\gamma$=5.0) | Llama-3.1-8B | 11.0 s | 2.8× | 0.133 |

### C.8 Parameter Sensitivity Analysis

We ablate each hyperparameter on ChatDoctor (Llama-3.1-8B, 100+100 member + non-member samples unless noted).[1] As default parameters we set: $\gamma$=2.0, $K$=2, $\beta$=5, $\psi$=0.05, $m$=5, $N/m$=3, $T$=0.3. Each ablation varies one parameter while holding the rest at their defaults. Vanilla RAG exposes 57% of members at 1% FPR, whereas a random classifier achieves only ≈1%.

Table 11: $\gamma$ ablation (ChatDoctor).

| $\gamma$ | Adv. AUC ↓ | TPR@1% ↓ | F1 ↑ | ROUGE-L ↑ | Ans. Rate |
|---|---|---|---|---|---|
| 0.5 | 0.531 | 0.060 | 0.139 | 0.137 | 18.0 |
| 1.0 | 0.520 | 0.040 | 0.135 | 0.130 | 21.5 |
| 2.0 | 0.515 | 0.060 | 0.142 | 0.135 | 35.5 |
| 3.0 | 0.501 | 0.070 | 0.140 | 0.135 | 43.5 |
| 5.0 | 0.504 | 0.040 | 0.138 | 0.133 | 52.5 |

**Similarity bound $\gamma$.** Table 11 shows that $\gamma$ has a significant impact on answer rate but its impact on privacy and utility metrics is more limited. All values achieve Adv. AUC within 0.50–0.53 and TPR@1% within 0.04–0.07, all near random guess. Token F1 varies only from 0.135 ($\gamma$=1.0) to 0.142 ($\gamma$=2.0). We use $\gamma$=2.0 as default since it strikes a good balance between privacy (Adv. AUC 0.515) and utility (F1 0.142).

Table 12: $K$ and $N/m$ ablation (ChatDoctor).

(a) $K$ sweep ($N/m$=3)

| $K$ | Adv. AUC ↓ | TPR@1% ↓ | F1 ↑ | Ans. Rate |
|---|---|---|---|---|
| 1 | 0.505 | 0.070 | 0.148 | 68.5 |
| 2 | 0.515 | 0.060 | 0.142 | 35.5 |
| 3 | 0.545 | 0.010 | 0.130 | 19.0 |
| 4 | 0.521 | 0.040 | 0.126 | 5.0 |

(b) $N/m$ sweep ($K$=2)

| $N/m$ | top-$N$ | Adv. AUC ↓ | TPR@1% ↓ | F1 ↑ |
|---|---|---|---|---|
| 1 | 5 | 0.536 | 0.050 | 0.136 |
| 3 | 15 | 0.515 | 0.060 | 0.142 |
| 5 | 25 | 0.538 | 0.060 | 0.134 |
| 10 | 50 | 0.537 | 0.010 | 0.146 |

**Count threshold $K$ and retrieval depth $N/m$.** Table 12(a) reveals that $K$ primarily impacts the utility-fallback tradeoff rather than privacy. We observe that Adv. AUC remains within the 0.50–0.55 range, but F1 drops from 0.148 ($K$=1) to 0.126 ($K$=4) as more queries fall back to data-independent generation. We use $K$=2 as default. Table 12(b) shows that retrieval depth $N/m$ (number of documents per group) has a mild effect. F1 is relatively stable (0.134–0.146), with $N/m$=3 achieving the lowest Adv. AUC (0.515).

[1]For this analysis, we use $m$=5, $K$=2 rather than the original headline $m$=20, $K$=8. This moderate regime yields higher answer rates (20–50%), lower computational cost (5× fewer groups), and better reflects practical development where practitioners first tune for quality and then tighten privacy.

Table 13: $m$ ablation (ChatDoctor). $N$: total number of retrieved documents; $N/m$=3 for $m \leq 10$; $N/m$=1 for $m$=50.

| $m$ | $N$ | Adv. AUC ↓ | F1 ↑ | RL ↑ | Ans. Rate | s/query |
|---|---|---|---|---|---|---|
| 3 | 9 | 0.507 | 0.145 | 0.142 | 25.5 | 43 |
| 5 | 15 | 0.515 | 0.142 | 0.135 | 35.5 | 58 |
| 10 | 30 | 0.513 | 0.130 | 0.124 | 53.5 | 100 |
| 50 | 50 | 0.533 | 0.128 | 0.123 | 63.0 | 361 |

**Number of groups $m$.** Increasing $m$ raises the answer rate (25.5% to 63.0%) but utility peaks at $m$=3 and $m = 5$ (F1 0.145–0.142) and declines for larger $m$. We use $m$=5 as default.

Table 14: Temperature ablation (ChatDoctor).

| $T$ | Adv. AUC ↓ | TPR@1% ↓ | F1 ↑ | ROUGE-L ↑ | Ans. Rate |
|---|---|---|---|---|---|
| 0.1 | 0.509 | 0.040 | 0.134 | 0.129 | 32.0 |
| 0.3 | 0.515 | 0.060 | 0.142 | 0.135 | 35.5 |
| 0.5 | 0.515 | 0.030 | 0.140 | 0.133 | 30.0 |
| 0.7 | 0.540 | 0.020 | 0.137 | 0.131 | 42.0 |

**Temperature $T$.** Higher temperature increases answer diversity but has a mixed effect on privacy, although we observe that TPR@1% drops from 4% to 2% as $T$ increases. F1 peaks at $T$=0.3 (0.142). We use $T$=0.3 as default.

Table 15: $\beta$ ablation (ChatDoctor).

| $\beta$ | Adv. AUC ↓ | TPR@1% ↓ | F1 ↑ | ROUGE-L ↑ |
|---|---|---|---|---|
| 1.43 | 0.557 | 0.180 | 0.155 | 0.196 |
| 2 | 0.588 | 0.060 | 0.148 | 0.150 |
| 5 | 0.515 | 0.060 | 0.142 | 0.135 |
| 10 | 0.501 | 0.020 | 0.133 | 0.129 |

**Geometric noise $\beta$.** In theory, the geometric noise has the effect of randomizing privacy test pass/fail decisions since it allows passing the test with too few alternatives, as well prevents passing with more than enough alternatives. Increasing $\beta$ actually decreases this randomness since larger $\beta$ imply lower expected noise. In experiments, we find that small $\beta$ ($\beta$=1.43) is harmful: TPR@1%=18%. At larger values such as $\beta \geq 5$, we observe a reduction in Adv. AUC. We set $\beta$=5 to strike a good privacy-utility balance (Adv. AUC 0.515, F1 0.142).

**Ceiling probability $\psi$.** The ceiling parameter $\psi$ caps the acceptance probability at $1-\psi$, so larger $\psi$ yields smaller $\delta$ (Eq. (7)). In a 500+500 sweep (Table 16), TPR@1% decreases from 3.2% ($\psi$=0.0) to 0.8% ($\psi$=0.7) while F1 stays within 0.131–0.138. We set $\psi$=0.05 as default.

### C.9 Answer Length and the PD Test

Recall that Eq. (6) counts the number of alternatives with a log-probabilities that fall within the same $\gamma$-width bin. Since the answer log-probabilities is the sum of the token log-probabilities, the more tokens (i.e., the longer the answers), the harder it becomes to find sufficient plausible alternatives for the privacy test to pass. However, in experiments we observe that for a fixed $\gamma$, the failure rate of the privacy test increases approximately as the square-root of the answer's token-length.

We vary the maximum generation length across `max_new_tokens` $\in \{32, 64, 128, 256, 512\}$ on ChatDoctor (Llama-3.1-8B, $K$=2, $m$=5, $R$=0). Results are shown in Fig. 4 for $\gamma \in \{2, 5, 8\}$. The figure shows that the answer rate drops with increasing answer length, but that larger $\gamma$ can compensate for a longer answers.

Table 16: $\psi$ ablation (ChatDoctor).

| $\psi$ | Adv. AUC ↓ | TPR@1% ↓ | F1 ↑ | RL ↑ | Ans. Rate |
|---|---|---|---|---|---|
| 0.0 | 0.510 | 0.032 | 0.137 | 0.126 | 31.2 |
| 0.1 | 0.540 | 0.034 | 0.138 | 0.126 | 32.1 |
| 0.3 | 0.501 | 0.020 | 0.135 | 0.121 | 23.2 |
| 0.5 | 0.518 | 0.018 | 0.131 | 0.124 | 18.7 |
| 0.7 | 0.518 | 0.008 | 0.133 | 0.120 | 11.9 |

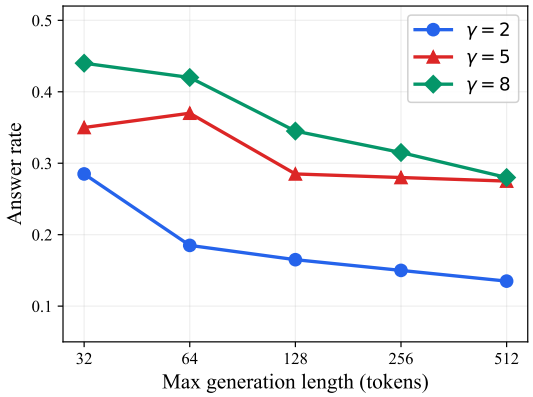



Figure 4: **Answer rate vs. answer length** for different $\gamma$ values (ChatDoctor, Llama-3.1-8B, $K$=2, $m$=5, $R$=0). Longer answers reduce answer rates across all $\gamma$. At 512 tokens, $\gamma$=5 and $\gamma$=8 converge (≈28%), indicating a floor effect.

Note that increasing $\gamma$ to compensate for answer length does *not* necessarily mean increasing the privacy budget, since one can simultaneously adjust other privacy parameters. For example, one could increase $\gamma$, $K$, $t$ (and potentially $R$) simultaneously to maintain (roughly the same) $\varepsilon$ and $\delta$.

In practice, $\gamma$ should be tuned in an application-specific way. For applications requiring long answers (e.g., medical explanations, legal summaries), larger $\gamma$ are desirable to maintain reasonably high answer rates. For short-answer tasks, small $\gamma$ values could be used to achieve stringent privacy guarantees while still allowing for high answer rates.

### C.10 Hyperparameter Search

PD-RAG has six tunable parameters. We summarize their roles and relationships with privacy-utility tradeoff below:

| Parameter | Role | Privacy effect | Utility effect |
|---|---|---|---|
| $\gamma$ | Log-prob similarity bound | $\varepsilon \propto \log(1+e^{\gamma}/t)$ | $\gamma\uparrow \Rightarrow$ answer rate $\uparrow$ |
| $K$ | Count threshold | $\delta \propto \beta^{-(K-t)}$ | $K\uparrow \Rightarrow$ answer rate $\downarrow$ |
| $m$ | Number of groups | $\delta \propto 1/m$ | $m\uparrow \Rightarrow$ context per group $\downarrow$ |
| $R$ | Max retries | $\varepsilon_{\text{total}}{=}R\varepsilon_{\perp}{+}\varepsilon_{\text{pass}}$ | $R\uparrow \Rightarrow$ answer rate $\uparrow$ |
| $\beta$ | Geometric noise | $\varepsilon \propto \log \beta$ | Mixed / Limited |
| $\psi$ | Ceiling probability | $\delta \propto (1{-}\psi)$ | Mixed / Limited |

The parameters $\beta$ and $\psi$ have mixed / limited utility impact and can be fixed ($\beta$=5, $\psi$=0.05) across datasets. The remaining knobs are $\gamma$, $K$, $m$, and $R$, which jointly determine the privacy-utility tradeoff.

We propose a three-phase search procedure to progressively ramp up compute as more promising configurations are identified (and non-viable ones are discarded). We find that we can eliminate large portion of the search space by initially discarding configuration that violate privacy desiderata. Recall from Theorem 1 and its composition the privacy budget:

$$\varepsilon_{\text{total}} = R\,\varepsilon_{\perp} + \log\beta + \log\!\left(1{+}\tfrac{e^{\gamma}}{t}\right) \quad \text{and} \quad \delta_{\text{total}} = \tfrac{(1-\psi)\,\beta^{-(K-t)}}{(1+\beta^{-1})\,m}\,.$$

**Algorithm 2** PD-RAG Hyperparameter Search

**Input:** Privacy budget $\varepsilon_{\max}$, $\delta_{\max}$; dataset $D$; screening sample size $N_s$; full sample size $N_f$
**Input:** Search grid $\mathcal{G} = \Gamma \times \mathcal{K} \times \mathcal{M} \times \mathcal{R} \times \mathcal{N}_g$; fixed $\beta$, $\psi$

**Phase 0: Analytical Pre-filtering (cost: 0 queries)**
$\mathcal{S}_0 \leftarrow \emptyset$
**for** each $(\gamma, K, m, R, N/m) \in \mathcal{G}$ **do**
    Compute $\varepsilon_{\text{total}}, \delta_{\text{total}}$ via Theorem 1
    **if** $\varepsilon_{\text{total}} \leq \varepsilon_{\max}$ **and** $\delta_{\text{total}} \leq \delta_{\max}$ **and** $K < m$ **then**
        $\mathcal{S}_0 \leftarrow \mathcal{S}_0 \cup \{(\gamma, K, m, R, N/m)\}$
    **end if**
**end for**

**Phase 1: Quick Screening (cost: $|\mathcal{S}_0| \times N_s$ queries)**
$\mathcal{S}_1 \leftarrow \emptyset$
**for** each config $c \in \mathcal{S}_0$ **do**
    Run PD-RAG with config $c$ on $N_s$ queries
    **if** answer rate $> 0$ **then**
        $\mathcal{S}_1 \leftarrow \mathcal{S}_1 \cup \{c\}$
    **end if**
**end for**

**Phase 2: Full Evaluation (cost: $|\mathcal{S}_1| \times N_f$ queries)**
**for** each config $c \in \mathcal{S}_1$ **do**
    Run PD-RAG with config $c$ on $N_f$ member + $N_f$ non-member queries
    Record utility (ROUGE-L, F1) and privacy (Adv. AUC)
**end for**
**return** config with highest utility

**Phase rationale.** *Phase 0* exploits the fact that $\varepsilon$ and $\delta$ are functions of the mechanism parameters alone and therefore configuration violating the budget constraints can be eliminated at (essentially) zero computational cost. With our default grid (504 configurations), Phase 0 eliminates 82% of the search space.

*Phase 1* identifies *degenerate* configurations where the PD test never passes (answer rate $\approx 0$). This occurs when $\gamma$ is too small (or $K$ is too large) to allow us to find plausible alternatives given the current answer length and test threshold. For estimating the answer rate only a small number of queries per configuration is needed. For example 20 queries suffices if the true answer rate is at least $15\%$, since in such a case the probability of observing at least one success in 20 trials is $1 - (1 - 0.15)^{20} \approx 96\%$.

*Phase 2* performs full evaluation only on the ($\approx$30) surviving configurations, measuring both utility (ROUGE-L, Token F1) and privacy (Adv. AUC via $S^2$MIA). The configuration with the highest utility is then selected. Since the privacy guarantee holds by construction (Phase 0), utility is the sole remaining selection criterion.

**Cost analysis example.** With the grid $\gamma \in \{0.5, 1, 1.5, 2, 3, 5, 8\}$, $K \in \{2, 4, 6, 8\}$, $m \in \{5, 10, 20\}$, $R \in \{0, 1, 2\}$, $N/m \in \{1, 3\}$, the full grid contains 504 configurations. At $\varepsilon_{\max}$=8, Phase 0 reduces this to $\approx$88 (5.7$\times$ reduction). Phase 1 with 20 queries further prunes to $\approx$30 viable configuration. Phase 2 evaluates these on $N_f$=100+100 queries. Total cost: $\approx$7,600 queries vs. 100,800 for brute force (13$\times$ reduction). All three phases can be parallelized across GPU workers.

### C.11 Higher-Coverage Operating Points

All additional experiments below average over three random seeds. Answer rate denotes the fraction of queries that receive a retrieval-grounded answer. We examine additional operating points obtained by joint parameter tuning.

**ChatDoctor joint tuning** Table 17 reports the joint parameter sweep on ChatDoctor. C1 increases answer rate from 20.2% to 46.0% at a similar privacy budget. Its runtime is 18.58 s/query, compared with 12 s/query for C0.

Table 17: ChatDoctor joint tuning. C0 is the headline configuration.

| Config. | $\gamma$ | $K$ | $m$ | $R$ | $\varepsilon$ | $\delta$ |
|---|---|---|---|---|---|---|
| C0 | 2.1 | 8 | 20 | 1 | 5.40 | $5.1 \times 10^{-7}$ |
| C1 | 2.2 | 7 | 51 | 1 | 5.39 | $\approx 9.9 \times 10^{-7}$ |
| C2 | 2.4 | 7 | 51 | 1 | 5.57 | $9.9 \times 10^{-7}$ |
| C3 | 3.8 | 8 | 20 | 0 | 5.43 | $5.1 \times 10^{-7}$ |
| C4 | 3.8 | 7 | 51 | 0 | 5.43 | $9.9 \times 10^{-7}$ |

| Config. | Answer rate ↑ | Adv.AUC ↓ | Token-F1 ↑ |
|---|---|---|---|
| C0 | 20.2% | 0.512 | 0.135 |
| C1 | 46.0% | 0.527 | 0.131 |
| C2 | 49.0% | 0.521 | 0.132 |
| C3 | 20.5% | 0.521 | 0.129 |
| C4 | 36.5% | 0.525 | 0.131 |

**Cross-dataset tuning** Table 18 compares the original and tuned configurations across all three datasets. The tuned QASPER and ASQA configurations use $N/m = 1$ and $N/m = 3$, respectively. Their answer rates increase to 34.0% and 69.5%.

Table 18: Original and tuned operating points across datasets. ChatDoctor tuned is C1 in Table 17.

| Dataset | Config. | $\gamma$ | $K$ | $m$ | $R$ | $\varepsilon$ | $\delta$ |
|---|---|---|---|---|---|---|---|
| ChatDoctor | Original | 2.1 | 8 | 20 | 1 | 5.40 | $5.1 \times 10^{-7}$ |
| ChatDoctor | Tuned | 2.2 | 7 | 51 | 1 | 5.39 | $\approx 9.9 \times 10^{-7}$ |
| QASPER | Original | 2.00 | 8 | 20 | 1 | 5.31 | $5.1 \times 10^{-7}$ |
| QASPER | Tuned | 2.18 | 6 | 254 | 1 | 5.31 | $\approx 9.97 \times 10^{-7}$ |
| ASQA | Original | 2.00 | 8 | 20 | 1 | 5.31 | $5.1 \times 10^{-7}$ |
| ASQA | Tuned | 2.10 | 7 | 51 | 1 | 5.30 | $\approx 9.9 \times 10^{-7}$ |

| Dataset | Config. | Answer rate ↑ | Adv.AUC ↓ | Token-F1 ↑ |
|---|---|---|---|---|
| ChatDoctor | Original | 20.2% | 0.512 | 0.135 |
| ChatDoctor | Tuned | 46.0% | 0.527 | 0.131 |
| QASPER | Original | 17.3% | 0.511 | 0.062 |
| QASPER | Tuned | 34.0% | 0.520 | 0.060 |
| ASQA | Original | 19.7% | 0.515 | 0.148 |
| ASQA | Tuned | 69.5% | 0.526 | 0.154 |

**Free-knob sweep** We vary the number of documents per group and the decoding temperature at $(\varepsilon, \delta) = (5.40, 5.1 \times 10^{-7})$. The highest answer rate in Table 19 is 22.5%.

Table 19: Free-knob sweep at the headline privacy budget on ChatDoctor.

| Configuration | Answer rate ↑ | Adv.AUC ↓ | Token-F1 ↑ |
|---|---|---|---|
| $N/m = 1$ | 21.0% | 0.527 | 0.127 |
| $N/m = 3$ | 21.5% | 0.526 | 0.131 |
| $N/m = 5$ | 12.0% | 0.509 | 0.134 |
| $N/m = 3, T = 0.1$ | 17.5% | 0.530 | 0.120 |
| $N/m = 3, T = 0.5$ | 22.5% | 0.535 | 0.135 |

**Higher privacy-budget operating points** Increasing the similarity bound $\gamma$ and the number of retries $R$ provides higher-budget operating points. Table 20 reports their answer rates and Adv.AUC.

Table 20: Higher privacy budgets on ChatDoctor.

| $\varepsilon$ | Answer rate ↑ | Adv.AUC ↓ |
|---|---|---|
| 5.4 | 20% | 0.51 |
| 8.2 | 38% | 0.52 |
| 9.2 | 45% | 0.52 |
| 11.2 | 60% | 0.52 |

### C.12 Matched Privacy-Budget Comparison

We compare the three private methods on ChatDoctor at $(\varepsilon, \delta) = (5.4, 5.1 \times 10^{-7})$. At this common budget, all three remain close to random guessing under the evaluated attack. PD-RAG obtains the highest ROUGE-L, ROUGE-1, and Token-F1.

Table 21: Matched privacy-budget comparison on ChatDoctor.

| Method | Adv.AUC ↓ | TPR@1% ↓ | ROUGE-L ↑ | ROUGE-1 ↑ | Token-F1 ↑ |
|---|---|---|---|---|---|
| PD-RAG | 0.512 | 0.037 | 0.126 | 0.264 | 0.135 |
| DP-KSA | 0.518 | 0.015 | 0.104 | 0.225 | 0.114 |
| DPSparseVoteRAG | 0.525 | 0.030 | 0.020 | 0.033 | 0.024 |

### C.13 DP-Synthetic-Corpus Baselines

We evaluate two synthetic-corpus baselines on ChatDoctor. DP-LoRA fine-tunes a Llama-3.2-1B synthesizer with LoRA adapters [15] using DP-SGD, following [43] and [44]. It then generates a 14k-document synthetic corpus for RAG. Aug-PE [40] uses model-generated text variations and DP nearest-neighbor voting to evolve a synthetic corpus without fine-tuning. Both use Llama-3.1-8B as the downstream generator and retrieve only from the synthetic corpus.

Table 22 reports the results. Synthetic-corpus methods incur a one-time privacy cost for synthesis, whereas PD-RAG incurs a per-query cost.

Table 22: Synthetic-corpus baselines on ChatDoctor. Privacy budgets apply to corpus synthesis for DP-LoRA and Aug-PE, and to each query for PD-RAG.

| Method | $\varepsilon$ | ROUGE-L ↑ | Adv.AUC ↓ |
|---|---|---|---|
| DP-LoRA | 8 | 0.106 | 0.52 |
| DP-LoRA | 32 | 0.105 | 0.53 |
| DP-LoRA | 64 | 0.108 | 0.53 |
| DP-LoRA | 128 | 0.108 | 0.54 |
| Aug-PE | 8 | 0.113 | 0.53 |
| Aug-PE | 32 | 0.114 | 0.53 |
| Aug-PE | 64 | 0.114 | 0.54 |
| Aug-PE | 128 | 0.115 | 0.54 |
| PD-RAG | 5.40 | 0.126 | 0.51 |
| Non-private LoRA | $\infty$ | 0.153 | 0.67 |

### C.14 Source-Document Overlap

For each answer, we compute overlap against each retrieved private document and take the maximum over documents. Table 23 reports the fraction with a verbatim span of at least 20 tokens and the maximum single-source ROUGE-L under the headline configuration. Accepted candidates have less overlap than rejected candidates. The deployed-output row includes data-independent fallback answers.

Table 23: Single-document source overlap under the headline configuration.

| Population | ≥20-token verbatim | Max-source ROUGE-L |
|---|---|---|
| Vanilla RAG | 47.2% | 0.348 |
| Rejected candidates | 29.1% | 0.265 |
| Accepted candidates | 10.9% | 0.194 |
| Deployed outputs (including fallback) | 3.7% | 0.159 |

### C.15 Stronger and Adaptive Attacks

We evaluate the attacks on ChatDoctor with 100 member and 100 non-member queries. Here $q$ is the number of system invocations per target. MBA [26] uses one document-specific masked prompt ($q = 1$), with Llama-3.1-8B as its proxy in place of the published GPT-2-XL. IA [29] uses $q = 10$ probes generated with a Llama-3.2-3B-Instruct shadow model. Adaptive IA also uses $q = 10$, but generates probe $j$ conditional on previously released answers and scores the full interaction transcript.

The PD-RAG configuration has a per-invocation budget $\varepsilon = 5.40$. The ten-query attacks observe a transcript with a reported composed budget of $\varepsilon \leq 54$ and $\delta \leq 5.1 \times 10^{-6}$. Table 24 reports the attack results.

Table 24: Stronger and adaptive attacks on ChatDoctor. TPR is measured at 1% FPR.

| | | Adv.AUC ↓ | | TPR@1% ↓ | |
|---|---|---|---|---|---|
| Attack | $q$ | Vanilla | PD-RAG | Vanilla | PD-RAG |
| $S^2$MIA-G | 1 | 0.823 | 0.512 | 0.617 | 0.037 |
| MBA | 1 | 0.850 | 0.526 | 0.736 | 0.000 |
| IA | 10 | 0.903 | 0.532 | 0.831 | 0.010 |
| Adaptive IA | 10 | 0.922 | 0.535 | 0.882 | 0.020 |

### C.16 Sample-Size and Retriever Robustness

Table 25 compares 100+100 and 500+500 member/non-member samples on ChatDoctor using the headline configuration and $S^2$MIA-G. Adv.AUC changes from 0.512 to 0.515, while answer rate changes from 20.2% to 18.7%.

Table 25: Sample-size robustness on ChatDoctor. Sample counts are members + non-members.

| Sample size | Adv.AUC ↓ | TPR@1% ↓ | ROUGE-L ↑ | ROUGE-1 ↑ | Token-F1 ↑ | Answer rate ↑ |
|---|---|---|---|---|---|---|
| 100+100 | 0.512 | 0.037 | 0.126 | 0.264 | 0.135 | 20.2% |
| 500+500 | 0.515 | 0.015 | 0.123 | 0.258 | 0.131 | 18.7% |

We also replace DPR with BGE-base-v1.5 [39] and GTE-base-v1.5 [45], keeping the mechanism parameters fixed. Table 26 reports retrieval recall and attack performance.

Table 26: Retriever robustness on ChatDoctor with fixed mechanism parameters.

| Retriever | Recall@15 ↑ | Vanilla Adv.AUC ↓ | PD-RAG Adv.AUC ↓ | PD-RAG TPR@1% ↓ |
|---|---|---|---|---|
| DPR | 60.7% | 0.823 | 0.512 | 0.037 |
| BGE-base-v1.5 | 83.3% | 0.922 | 0.519 | 0.037 |
| GTE-base-v1.5 | 83.0% | 0.934 | 0.523 | 0.017 |

### C.17 Multi-Judge Evaluation

We score the same examples with GPT-5.4, Gemini-2.5-flash, Claude Sonnet 5, and kimi-k2.6 on the 1–5 quality scale. The PD-RAG scores use the original headline configurations in Table 7. PD-RAG receives the highest score among the private methods in every dataset-judge combination (Table 27).

Table 27: Answer quality across four LLM judges. Higher scores indicate better quality.

| Dataset | Method | GPT-5.4 | Gemini-2.5-flash | Claude Sonnet 5 | kimi-k2.6 |
|---|---|---|---|---|---|
| ChatDoctor | Vanilla | 2.47 | 2.82 | 3.20 | 2.82 |
| ChatDoctor | DPSparseVote | 2.01 | 1.73 | 1.80 | 1.82 |
| ChatDoctor | DP-KSA | 2.06 | 1.82 | 2.12 | 1.98 |
| ChatDoctor | PD-RAG | 2.35 | 2.34 | 2.62 | 2.38 |
| QASPER | Vanilla | 1.94 | 3.28 | 2.88 | 2.76 |
| QASPER | DPSparseVote | 1.19 | 1.68 | 1.12 | 1.66 |
| QASPER | DP-KSA | 1.32 | 1.80 | 1.26 | 1.81 |
| QASPER | PD-RAG | 1.57 | 2.16 | 1.50 | 1.98 |
| ASQA | Vanilla | 2.75 | 3.32 | 3.70 | 3.06 |
| ASQA | DPSparseVote | 1.48 | 1.46 | 1.35 | 1.46 |
| ASQA | DP-KSA | 2.17 | 2.64 | 2.10 | 2.40 |
| ASQA | PD-RAG | 2.52 | 2.90 | 2.38 | 2.72 |

## D Broader Impacts

PD-RAG is designed to safeguard the privacy of documents in RAG systems. Reducing privacy threats may enable organizations to deploy RAG systems over sensitive corpora (e.g., medical records, legal documents, internal knowledge bases). This has positive societal implications for healthcare, legal, and enterprise applications where retrieval-augmented systems are desirable but privacy concerns currently hinder deployment. We do not foresee any negative societal impacts specific to this work. PD-RAG does not enable new attack capabilities.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our claims match the theoretical and empirical results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss it in the paper.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide full proofs in appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all experiments details in main text and appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code cannot be publicly released due to institutional policy; we provide full implementation details in Section 3 and Appendix C to enable reproduction.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide all details in appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We provide error bars for the results in main text.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide all details in appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research follows the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss it in appendix.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our paper does not have such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite all datasets, code bases and models.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.