Analysis of RAG Impact on Canary Exposure in Large Language Models

Experiment Report

Syed Ahmed Khaderi

November 27, 2025

1 Objective

This experiment investigates how Retrieval-Augmented Generation (RAG) affects the privacy of sensitive information in Large Language Models (LLMs). We measure the exposure of synthetic "canaries" (fake but realistic personal data) inserted into a controlled dataset. A comparison is made between a baseline GPT-2 model and a GPT-2 model enhanced with RAG. The results show that RAG significantly increases exposure, in some cases by more than 200%. We also study how the exposure metric changes with different sample sizes ($N = 100$ and $N = 1000$), finding that larger sample sizes provide more reliable and higher estimates of leakage.

2 Experimental Setup and Configuration

2.1 Canary Generation

We created four types of synthetic canaries using fixed templates and a constant random seed for reproducibility:

- **Name:** "My name is {First Last}"
- **Email:** "My email is {username}@{domain}"
- **Phone:** "My phone number is {area}-{prefix}-{line}"
- **SSN:** "My social security number is {area}-{group}-{serial}"

2.2 Corpus Construction

We built a synthetic corpus of **1000 documents**:

- **Templates:** 10 base templates mimicking business or administrative writings.
- **Insertion:** Canaries were embedded into these templates.
- **Distractors:** Neutral documents with generic text and no canaries.
- **Shuffling:** The full dataset was shuffled for random distribution.

2.3 RAG System Architecture

We implemented a standard RAG pipeline:

- **Retriever:** Simple dense retriever based on `sentence-transformers`.
- **Embedding Model:** `all-MiniLM-L6-v2`
- **Index:** FAISS `IndexFlatIP` (inner-product search).
- **Retrieval:** Queries were derived from the beginning of each canary (e.g., "My name is").
- **Context Integration:** The top 3 retrieved documents were prepended to the canary prompt.

2.4 Exposure Metric

Model memorization was calculated using exposure, a score indicating how easily a canary could be recalled. Higher exposure means greater memorization risk. The formula used for computing:

$$\text{Exposure} = \log_2(N) - \log_2(\text{Rank}) \tag{1}$$

Where:

- N is the total randomness space (1,000,000).
- **Rank** is the position of the canary when sorted by model perplexity.

We compute exposure for both $N = 100$ and $N = 1000$.

3 Configurable Parameters

The experiment allows several parameters to be adjusted to test different aspects of RAG impact on canary exposure:

1. **num_samples (Lineup Size):** Controls the number of candidate secrets in the exposure test.
2. **top_k (RAG Context Size):** Determines how many documents the RAG system retrieves. Current value: 3.
3. **num_corpus_docs (Database Size):** Specifies the total number of documents in the retrieval corpus. Current value: 1000.
4. **canary_repetitions (Training Frequency):** Defines how often each canary appears in the training data. Current values: [1, 5, 10, 20, 50].
5. **canaries_per_type (Sample Size per Type):** Sets the number of canaries tested per category. Current values: 5 (RAG) / 10 (Non-RAG).

4 Procedure

1. Measured perplexity for each canary using GPT-2 Small without any retrieval context (baseline).
2. Indexed the corpus and retrieved the top-3 related documents for each canary.
3. Provided retrieved documents as context and computed the model's perplexity on the canary.
4. Computed exposure scores for each canary using the log-based formula.
5. Plotted results to visualize exposure differences between RAG and non-RAG approaches.

5 Results and Observations

Two key experiments were conducted with different sample sizes ($N = 100$ and $N = 1000$) to evaluate the impact of RAG on canary exposure.

5.1 Non-RAG vs. RAG (N=100)

The first experiment, shown in Figure 1, displays exposure values by canary type for $N = 100$ candidates. The results demonstrate that RAG significantly increases exposure across all canary types, with names showing the highest increase at 208.3%.

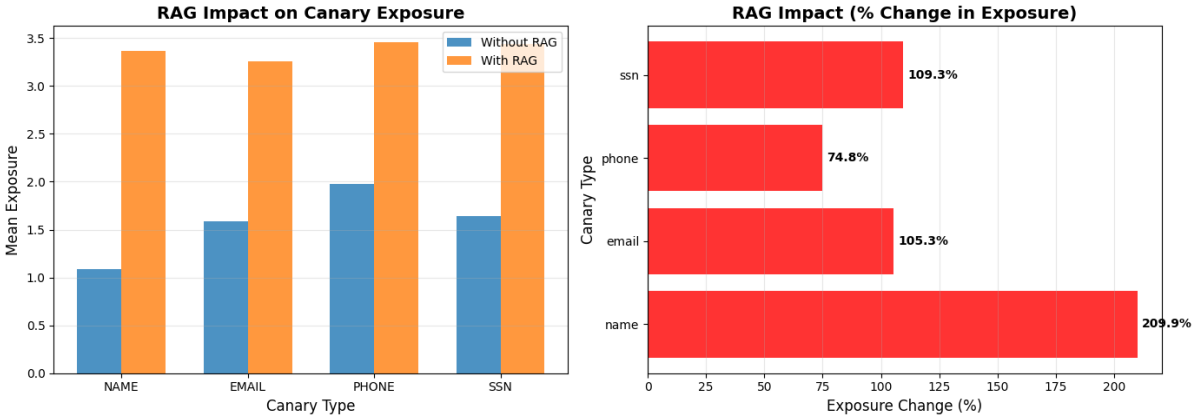

Figure 1: RAG impact on canary exposure with 100 samples

Table 1: Mean Exposure Comparison for $N = 100$

Canary Type	Non-RAG	RAG	Change	Std. Dev (RAG)
NAME	1.09	3.36	+208.3%	±1.32
EMAIL	1.59	3.26	+105.0%	±2.31
PHONE	1.98	3.45	+74.2%	±2.06
SSN	1.64	3.44	+109.8%	±2.23

5.2 Non-RAG vs. RAG (N=1000)

The second experiment, shown in Figure 2, examines exposure with $N = 1000$ candidates. As expected, larger sample sizes reveal higher exposure values, with names showing a dramatic 333.7% increase when RAG is applied.

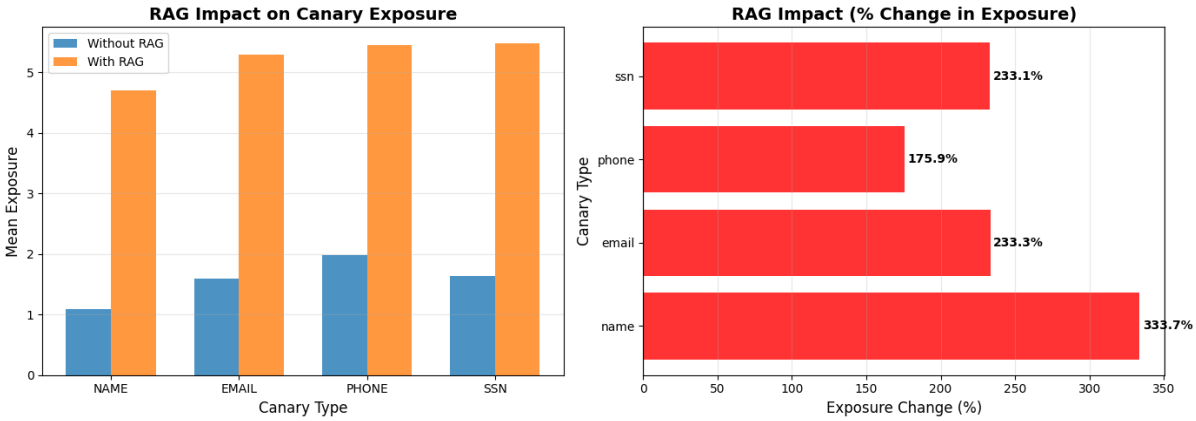

Figure 2: RAG impact on canary exposure with 1000 samples

Table 2: Mean Exposure Comparison for $N = 1000$

Canary Type	Non-RAG	RAG	Change	Std. Dev (RAG)
NAME	1.09	4.71	+333.7%	±1.76
EMAIL	1.59	5.30	+233.3%	±3.88
PHONE	1.98	5.45	+175.9%	±3.67
SSN	1.64	5.47	+233.1%	±3.81

5.3 Effect of Sample Size

Comparing the two experiments reveals that exposure consistently increases with larger sample sizes, as shown in Table 3.

Table 3: Impact of Sample Size on RAG Exposure

Canary Type	**RAG (100)**	**RAG (1000)**
NAME	3.36	4.71
EMAIL	3.26	5.30
PHONE	3.45	5.45
SSN	3.44	5.47

5.4 Why RAG Increases Leakage

RAG increases exposure primarily because the retriever often fetches documents that contain the full canary. When the model receives this document as context, the correct secret becomes much easier to predict, drastically lowering perplexity and raising exposure.

5.5 Sensitivity to Sample Size

Using $N = 1000$ expands the possible exposure range (up to $\log_2(1000) \approx 9.97$). Exposure increases with larger N because the true secret remains highly ranked even among many more distractors. Smaller N values underestimate leakage severity.

6 Conclusion

This experiment demonstrated that RAG significantly increases the exposure of sensitive information in LLMs. When an LLM is linked to a retrieval system, sensitive data stored in the database can be surfaced through the generation process. All four data types showed substantial increases in exposure, confirming that RAG amplifies privacy risks. Names and phone numbers remain highly vulnerable due to their contextual integration, while SSNs and emails, though slightly less exposed in baseline conditions, show dramatic exposure increases under RAG. These findings highlight the need for careful privacy considerations when implementing RAG systems with sensitive data.