RAG with Differential Privacy

Nicolas Grislain

December 5, 2024

Abstract

Retrieval-Augmented Generation (RAG) has emerged as the dominant technique to provide *Large Language Models* (LLM) with fresh and relevant context, mitigating the risk of hallucinations and improving the ovarall quality of responses in environments with large and fast moving knoledge bases. However, the integration of external documents into the generation process raises significant privacy concerns. Indeed, when added to a prompt, it is not possible to guarantee a reponse will not inadvertently expose confidential data, leading to potential breaches of privacy and ethical dilemmas. This paper explores a practical solution to this problem suitable to general knowledge extraction from personnal data. It shows *differentially private token generation* is a viable approach to private RAG.

Introduction

Retrieval-Augmented Generation (RAG, (Lewis et al. 2021)) has become a popular approach to enhance the capabilities of Large Language Models (LLMs) by supplying them with up-to-date and pertinent information. This method is particularly valuable in environments where knowledge bases are large and rapidly evolving, such as news websites, social media platforms, or scientific research databases. By integrating fresh context, RAG helps mitigate the risk of "hallucinations"—instances where the model generates plausible but factually incorrect information—and significantly improves the overall quality and relevance of the responses generated by the LLM.

However, incorporating external documents into the generation process introduces substantial privacy concerns. When these documents are included in the input prompt for the LLM, there is no foolproof way to ensure that the generated response will not accidentally reveal sensitive or confidential data (Qi et al. 2024). This potential for inadvertent data exposure can lead to serious breaches of privacy and presents significant ethical challenges. For instance, if an LLM is used in a healthcare setting and it accidentally includes patient information from an external document in its response, it could violate patient confidentiality and legal regulations.

This paper describes a practical solution (DP-RAG) aimed at addressing these privacy concerns with *Differential Privacy* (DP). The solution is based on two pillars:

- A method to collect documents related to the question in a way that does not prevent its output to be used in a DP mechanism.
- A method to use the collected documents to prompt a LLM and produce a reponse with DP guarantees.

The paper describes also some empirical tests and shows that *DP-RAG* is most effective in context where enough documents give elements of response.

Related Work

In general there are two families of approaches to add new knowledge to an LLM. The first is *Fine Tunning* (FT) and the other is *Retrieval Augmented Generation* (RAG). In both these approaches, adding privacy can be done, through simple heuristics with human validation such as *masking* or using a systematic and principle-based approach such as *Differential Privacy*.

Private Fine-Tuning

A straightforward approach to adding knowledge to an existing LLM is to continue its training with the new knowledge or to Fine Tune (FT) it. However, this raises challenges when dealing with private data, as LLMs tend to memorize training data. (see (Shokri et al. 2017) or (Carlini et al. 2021)).

To mitigate this privacy risk, it is possible to redact sensitive content prior to the FT process (aka. *masking*), but this operation is not very reliable and requires judgment on what should be redacted. This is a difficult manual operation based on the perceived sensitivity of each field and how it can be used to re-identify an individual, especially when combined with other publicly available data. Overall, it is very easy to get wrong; leaning too much on the side of prudence can yield useless data, while trying to optimize utility may result in leaking sensitive information.

A solution to this problem is to leverage *Differential Privacy*, a theoretical framework enabling the computation of aggregates with formal privacy garantees (See (Dwork, Roth, et al. 2014)).

The most common approache to Private LLM FT is to use Differentially-Private-Stochastic-Gradient-Descent (DP-SGD, see (Abadi et al. 2016) and (Ponomareva et al. 2023)). DP-SGD is about clipping gradients and adding them some noise while running your ordinary SGD (or standard variants such as *Adam*, etc.). This method requires the data to be organized per *privacy unit* (typically a privacy unit will be a user). Every training example should belong to one and only one privacy unit[1].

But, when new documents are frequently added to the private knowledge base FT may not be the best approach.

[1]Note that observations (examples) can be grouped into composite observations if one user contributes to many observations.

Private RAG

When FT is not the best approach to adding new knowledge and RAG would be preferred, DP-FT cannot help with privacy. In these cases, DP can still be levereaged in different ways. A straightforward approach to DP RAG is to generate synthetic documents with differential privacy out of the private knowledge base and then retrieve documents from this synthetic knowledge base instead of the private one. Another approach is to generate the LLM response in a DP way.

The approach of generating synthetic documents usable for RAG in privacy-sensitive contexts has been explored by (Zeng et al. 2024) but without DP garantees. There are three main approaches to the problem of generating DP Synthetic Data (SD):

- Fine-Tuning a pretrained generative model with DP to generate synthetic documents.
- Use some form of automated prompt tuning to generate synthetic prompts or context documents.
- And use DP aggregated generation.

Fine-Tuning a pretrained generative model with DP can be done with DP-SGD ((Abadi et al. 2016) and (Ponomareva et al. 2023)) as mentionned above. An application to synthetic text generation is described there: (Yue et al. 2023). This method is technically complex, as, DP-SGD can be challenging to implement efficiently (Bu et al. 2023).

In (Hong et al. 2024), the authors use an automated prompt tuning technique developped in (Sordoni et al. 2023) and (Zhou et al. 2023) and make it differentially private. From the evaluations presented, it seems to compare favorably to DP-FT synthetic data approaches. Similar methods, based on DP-automated prompt tuning are exposed in (Lin et al. 2024) for images and (Xie et al. 2024) for text.

A last approach to generating synthetic data is based on DP aggregation of data.(Lebensold et al. 2024) or (Wu et al. 2023) show how to aggregate images or text in their embedding space (aka. Embedding Space Aggregation). Aggregating data privately is also the approach of (Tang et al. 2024), but they do it at the token level.

This last method greatly inspired the approach described in this document, though not for SD, but to directly generate RAG output from private documents.

DP-RAG

To overcome the limitations of DP FT or SD-based RAG, we developped and tested DP-RAG: a novel approach, build upon recent works on DP In-Context Learning (ICL) such as (Wu et al. 2023) and particularily (Tang et al. 2024).

- Contrary to (Wu et al. 2023), we aggregate outputs token by token.
- Our token aggregation method is different from both methods exposed in: (Tang et al. 2024) (*Gaussian* and *Report Noisy Max*).

- Because we implement the full RAG system, we developped a method to collect the *top-k* most similar documents in a way that does not jeopardize the possibility to run a DP mechanism on them.

Overview of DP-RAG

DP-RAG is made of two main components:

- A method to collect documents related to the question in a way that does not prevent its output to be used in a DP mechanism.
- A method to use the collected documents to prompt a LLM and produce a reponse with DP guarantees.

To understand the need for these components, let's describe what RAG is usually made of (see also (Lewis et al. 2021)) and introduce some notations (see Fig. 1).

A LLM: $\mathcal{L}$ is a function, taking some text, in the form of a sequence of tokens: $x = \langle x_1, x_2, ..., x_n \rangle$ as input and outputing a probability distribution of the next token x_{n+1} conditional on x:

$$\mathcal{L}(s, x) = \mathcal{L}(s, \langle x_1, x_2, ..., x_n \rangle) = \Pr(x_{n+1} = s | \mathcal{L}, x_1, x_2, ..., x_n)$$

We assume we have a set of N documents: $D = \{d_1, d_2, ..., d_N\} \subset \mathcal{D}$ containing domain specific knowledge. These documents are also sequences of tokens: $d_i = \left\langle d_{i,1}, d_{i,2}, ..., d_{i,l_i} \right\rangle$. We will, for simplicity, denote $\left\langle d_i, d_j \right\rangle$ the concatenation of two sequences of token, or a sequence and one token.

We also assume we have a similarity function $S : \mathcal{D}^2 \mapsto [-1, 1]$ which value is close to 1 when two documents are very similar, close to 0 when independent, and close to -1 when conveying opposite meaning. In this work S will be the cosine similarity between some embedings of the documents, mapping them to some adequate d-dimensional vector space: $\mathbb{R}^d$:

$$S(d_i, d_j) = \frac{\left\langle E(d_i), E(d_j) \right\rangle}{\|E(d_i)\|_2 \|E(d_j)\|_2}$$

When receiving a query in the form of a sequence of token: $q = \left\langle q_1, q_2, ..., q_{n_q} \right\rangle$, the similarity between q and each document is computed and the top k documents in term of similarity are collected:

$$d_{i_1}, d_{i_2}, ... d_{i_k} \text{ with } S(q, d_{i_1}) \geq S(q, d_{i_2}) \geq ... \geq S(q, d_{i_N})$$

Then a new query q_{RAG} is built by concatenating the original query q with the top k documents and other elements (the operation is denoted $\langle \cdot, ..., \cdot \rangle_{RAG}$)

$$q_{RAG} = \left\langle q, d_{i_1}, d_{i_2}, ... d_{i_k} \right\rangle_{RAG}$$

The augmented query is then sent to the LLM to compute the distribution of the next token (the first token of the response)

$$\mathcal{L}\left(r_1, \left\langle q, d_{i_1}, d_{i_2}, \ldots d_{i_k}\right\rangle_{RAG}\right)$$

The token is generated by sampling according to the distribution[2] or by selecting the mode of the distribution[3].

The tokens of the response are then generated one by one in an auto-regressive manner. The generated response tokens are concatenated to the input sequence:

$$\mathcal{L}\left(r_{j+1}, \left\langle \left\langle q, d_{i_1}, d_{i_2}, \ldots d_{i_k}\right\rangle_{RAG}, r_1, r_2, \ldots, r_j\right\rangle\right)$$

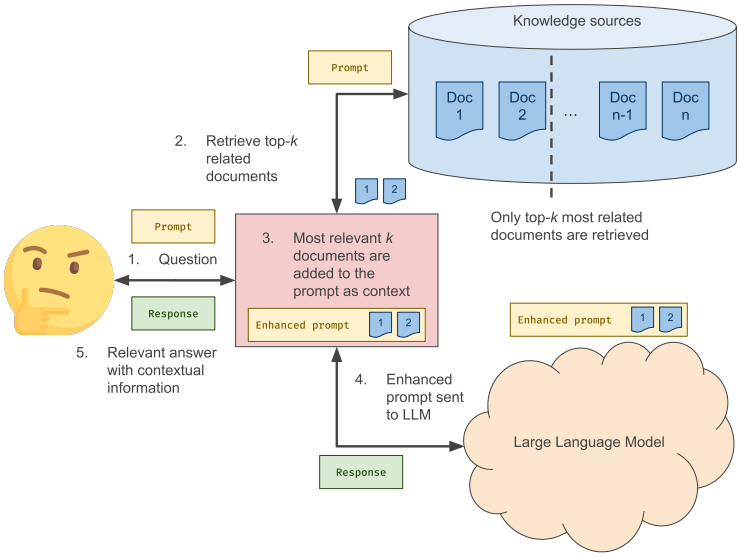

Figure 1: A broad picture of how RAG works

In the private variant of the problem (DP-RAG), we also assume the documents are *privacy sensitive*, and make the additional assumption that each document relates to only one individual that we call *privacy unit* (PU)[4].

Differential Privacy and its application to RAG

A (randomized) algorithm: $\mathcal{A}$ provides (ϵ, δ)-Differential Privacy *if and only if* for all event S and neighboring datasets D_0 and D_1, we have:

$$\Pr[\mathcal{A}(D_0) \in S] \leq e^{\varepsilon} \Pr[\mathcal{A}(D_1) \in S] + \delta$$

[2]or proportionaly to some power $1/T$ of the distribution

[3]the most likely token or the limit when T goes to 0

[4]Such structuration of documents by privacy unit can sometime be achieved by cutting documents and groupping all the content relative to one PU in one document.

This means that for datasets that differ by one individual (i.e. neighboring datasets) the algorithm's outputs are indistinguishable. This property guarantees that no bit of information can reasonably be learned about an individual. See (Dwork, Roth, et al. 2014) for a thorough introduction to DP.

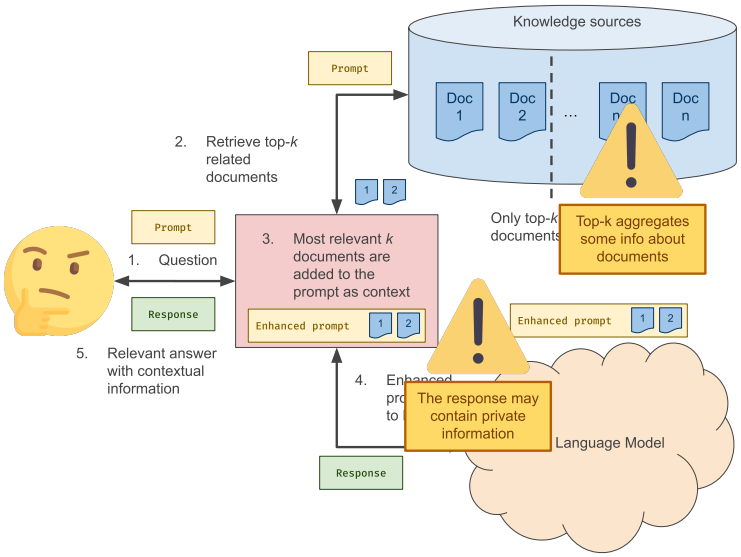

Figure 2: A broad picture of the main problems to overcome when considering DP RAG

There are two main challenges to implementing RAG with DP guarantees (see Fig. 2). One is to aggregate the knowledge from many documents with DP, the other, more subtle, consists in selecting the most relevant documents without jeopardizing our ability to apply a DP mechanism downstream.

Privacy Unit Preserving Document Retrieval

As mentionned above, DP deals with the concept of *neigboring* datasets. For this reason, it is convenient to assign each document to one and only one individual, or *privacy unity* (PU). Adding or removing one PU, comes down to adding or removing one document. In this context, one should be careful with the selection of the top-k most relevant document. Indeed, when selecting the top-k documents, adding or removing one document may affect the selection of other documents.

In DP-RAG, the similarity of each document with the query is computed:

$$s_1, s_2, \dots, s_N = S(q, d_1), S(q, d_2), \dots, S(q, d_N)$$

To estimate a threshold to select the top k documents with DP, we designed a utility function to be plugged into an exponential mechanism (Dwork, Roth, et al. 2014) (see Fig. 4).

$$U_{top-k}(\tau) : [0, 1] \mapsto \mathbb{R} = -\left|\sum_i \mathbb{1}_{[0,s_i]}(\tau) - k\right|$$

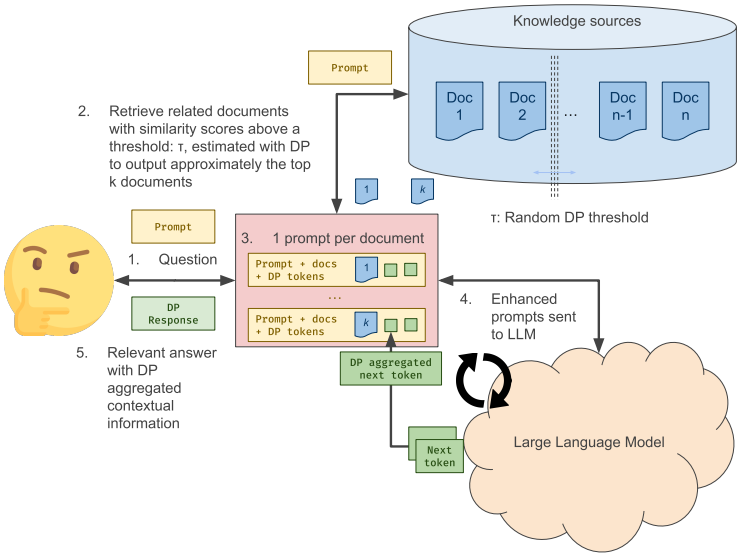

Figure 3: In DP-RAG, k smaller queries are sent to the LLM, rather than a single query (approximately) k times larger.

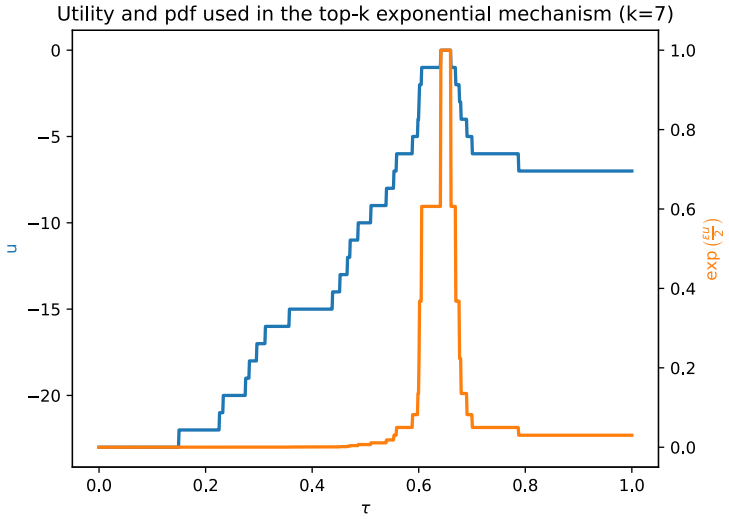

Figure 4: The exponential mechanism for the top-k DP-threshold. For the sake of clarity we chose a small number of documents: 30, and a large ϵ: 1

This *top-k* utility has sensitivity 1, we can sample a threshold τ_{DP} from the probability density funtion:

$$\tau_{top-k} \propto \exp\left(\frac{\epsilon U_{top-k}(\tau)}{2}\right)$$

It is easy to show τ_{top-k} is ϵ-DP (see. (Dwork, Roth, et al. 2014)).

The DP top-k threshold τ_{top-k} sampled from the exponential mechanism is then used to select all the documents whose similarity is above τ_{top-k}.

While this threshold, works well in practice, it selects a fixed number of documents (~k). We may be interested in selecting fewer when the top scores are more concentrated on few documents (the query is *selective*), and select more when the scores are evenly spread accross many documents (the query has a low *selectivity*). To adjust to this need, we designed a slightly different utility function:

$$U_{top-p}(\tau) : [0,1] \mapsto \mathbb{R} = -\left|\sum_i \mathbb{1}_{[0,s_i]}(\tau) w(s_i) - p \sum_i w(s_i)\right|$$

with:

$$w(s) = \exp\left(\alpha \frac{s - s_{\max}}{s_{\max} - s_{\min}}\right) \in [0,1] \text{ when } \alpha > 0$$

and similarily:

$$\tau_{top-p} \propto \exp\left(\frac{\epsilon U_{top-p}(\tau)}{2}\right)$$

This utility function (see Fig. 5) is parametrized by α which contrasts the differences in scores, and p which select the share of *total document weight* we want to select with the mechanism.

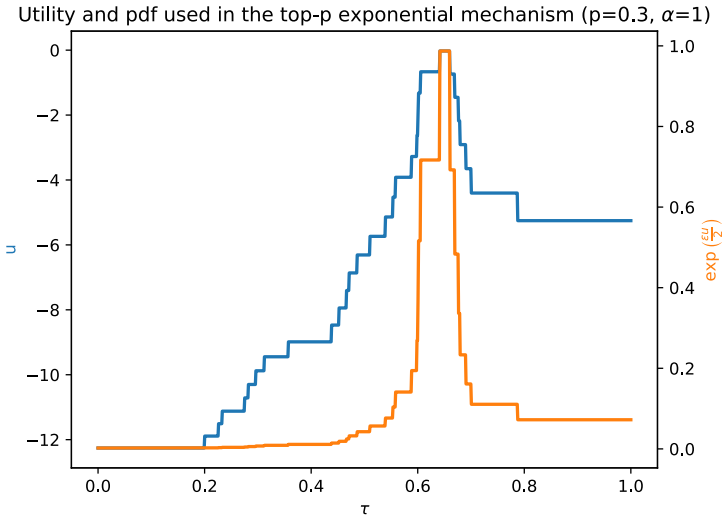

Figure 5: The exponential mechanism for the top-p DP-threshold. For the sake of clarity we chose a small number of documents: 30, and a large ϵ: 1

Once the τ_{top-p} threshols is sampled with DP, incurring a small *privacy loss*, it is safe to select the documents the similarity scores of which are above it. They are then *aggregated* with DP in the DP ICL phase.

Differentially Private In-Context Learning

In DP ICL, instead of sampling the next token from a query enhanced with many documents:

$$L_{j+1}(\cdot) = \mathcal{L}\left(\cdot, \left\langle \left\langle q, d_{i_1}, d_{i_2}, \dots d_{i_k} \right\rangle_{RAG}, r_1, r_2, \dots, \ r_j \right\rangle\right)$$

we compute the distributions of the next token for many enhanced queries, each of them with just one document:

$$\begin{cases} L_{j+1,i_1}(\cdot) &= \mathcal{L}\left(\cdot, \left\langle \left\langle q, d_{i_1} \right\rangle_{RAG}, r_1, r_2, \dots, \ r_j \right\rangle\right) \\ L_{j+1,i_2}(\cdot) &= \mathcal{L}\left(\cdot, \left\langle \left\langle q, d_{i_2} \right\rangle_{RAG}, r_1, r_2, \dots, \ r_j \right\rangle\right) \\ \quad \vdots & \\ L_{j+1,i_k}(\cdot) &= \mathcal{L}\left(\cdot, \left\langle \left\langle q, d_{i_k} \right\rangle_{RAG}, r_1, r_2, \dots, \ r_j \right\rangle\right) \end{cases}$$

We also compute the distribution of the next token, with some public context:

$$L_{j+1,\text{pub}}(\cdot) = \mathcal{L}\left(\cdot, \left\langle \left\langle q, d_{\text{pub}} \right\rangle_{RAG}, r_1, r_2, \dots, \ r_j \right\rangle\right)$$

Following (Tang et al. 2024), we sample a next token based on a DP aggregation of the $k+1$ distributions.

Contrary to (Tang et al. 2024) where they compare two mechanisms: *Gaussian* and *Report Noisy Max*, and use a public prior with *Reduce Vocab Publicly* (RVP) we introduce a different mechanism:

- We use an exponential mechanism with a utility aggregating transformed log-probability vectors from all the enhanced queries.
- We do not use Reduce Vocab Publicly (RVP), but a soft version, consisting in using the log-probabilities of a public response to boost or mute some tokens in a soft way.

We sample the next token from an exponential mechanism where the utility is the aggregation of some c modulated by the log-probabilities associated with the public query. The larger, the θ, the closer the response will be to the one without the private documents.

$$U_{ICL}(r) = \theta \cdot \ln\left(L_{j+1,\text{pub}}(r)\right) + \sum_j c_{j+1,i_j}(r)$$

Where c is h with its sup norm (or sensitivity) clipped to some C:

$$c_{j+1,i_j}(r) = h_{j+1,i_j}(r) \min\left(1, \frac{C}{\max_s \left|h_{j+1,i_j}(s)\right|}\right)$$

Where h is a centered version of g to minimize its sup norm without changing its impact in the mechanism:

$$h_{j+1,i_j}(r) = g_{j+1,i_j}(r) - \frac{\max_s g_{j+1,i_j}(s) + \min_s g_{j+1,i_j}(s)}{2}$$

Where g is a transformation of L putting more emphasis on the large values of L.

$$g_{j+1,i_j}(r) = \frac{\exp\left[\alpha \cdot \left(\ln L_{j+1,i_j}(r) - \ln \max_s L_{j+1,i_j}(s)\right)\right] - 1}{\alpha}$$

Indeed, for $\alpha = 1$ we simply compute a scaled and shifted version of the probability:

$$g_{j+1,i_j}(r) = \frac{L_{j+1,i_j}(r)}{\max_s L_{j+1,i_j}(s)} - 1$$

for α very small, we compute the log-probabilities:

$$g_{j+1,i_j}(r) \approx \ln L_{j+1,i_j}(r) - \ln \max_s L_{j+1,i_j}(s)$$

and for α very large, we get an indicator function:

$$g_{j+1,i_j}(r) \approx 0 \text{ if } r = \text{argmax}_s L_{j+1,i_j}(s) \text{ and } -1 \text{ elsewhere}$$

After the utility is computed, the next token is sampled from:

$$r \propto \exp\left(\frac{\epsilon U_{ICL}(r)}{2C}\right)$$

In this formula, the larger the ϵ (privacy loss), or the smaller the clipping C the closer we are to the most likely token.

The code in *Torch* is available on github.com/sarus-tech/dp-rag.

Evaluation

Conclusion

Abadi, Martin, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. "Deep Learning with Differential Privacy." In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. CCS'16. ACM. https://doi.org/10.1145/2976749.2978318.

Bu, Zhiqi, Yu-Xiang Wang, Sheng Zha, and George Karypis. 2023. "Differentially Private Optimization on Large Model at Small Cost." In *International Conference on Machine Learning*, 3192–3218. PMLR.

Carlini, Nicholas, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, et al. 2021. "Extracting Training Data from Large Language Models." In *30th USENIX Security Symposium (USENIX Security 21)*, 2633–50. USENIX Association. https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting.

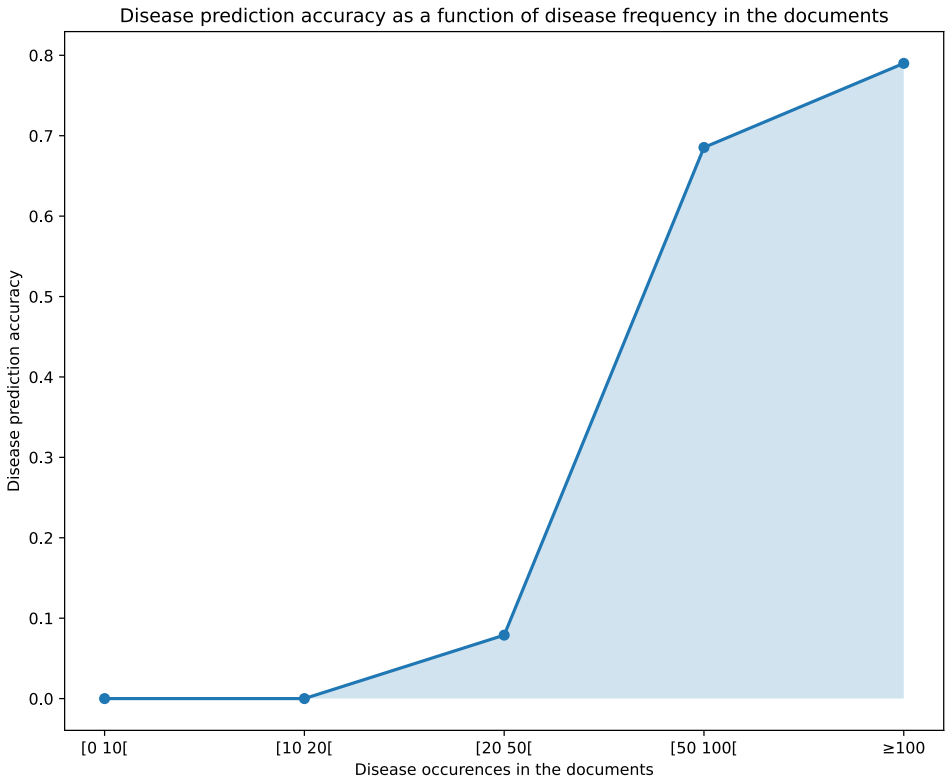

Figure 6: DP-RAG accuracy as a function of knowledge specificity

Dwork, Cynthia, Aaron Roth, et al. 2014. "The Algorithmic Foundations of Differential Privacy." *Foundations and Trends® in Theoretical Computer Science* 9 (3–4): 211–407.

Hong, Junyuan, Jiachen T. Wang, Chenhui Zhang, Zhangheng Li, Bo Li, and Zhangyang Wang. 2024. "DP-OPT: Make Large Language Model Your Privacy-Preserving Prompt Engineer." https://arxiv.org/abs/2312.03724.

Lebensold, Jonathan, Maziar Sanjabi, Pietro Astolfi, Adriana Romero-Soriano, Kamalika Chaudhuri, Mike Rabbat, and Chuan Guo. 2024. "DP-RDM: Adapting Diffusion Models to Private Domains Without Fine-Tuning." https://arxiv.org/abs/2403.14421.

Lewis, Patrick, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, et al. 2021. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." https://arxiv.org/abs/2005.11401.

Lin, Zinan, Sivakanth Gopi, Janardhan Kulkarni, Harsha Nori, and Sergey Yekhanin. 2024. "Differentially Private Synthetic Data via Foundation Model APIs 1: Images." https://arxiv.org/abs/2305.15560.

Ponomareva, Natalia, Hussein Hazimeh, Alex Kurakin, Zheng Xu, Carson Denison, H. Brendan McMahan, Sergei Vassilvitskii, Steve Chien, and Abhradeep Guha Thakurta. 2023. "How to DP-Fy ML: A Practical Guide to Machine Learning with Differential Privacy." *Journal of Artificial Intelligence Research* 77 (July): 1113–1201. https://doi.org/10.1613/jair.1.14649.

Qi, Zhenting, Hanlin Zhang, Eric Xing, Sham Kakade, and Himabindu Lakkaraju. 2024. "Follow My Instruction and Spill the Beans: Scalable Data Extraction from Retrieval-Augmented Generation Systems." https://arxiv.org/abs/2402.17840.

Shokri, Reza, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. "Membership Inference Attacks Against Machine Learning Models." In *2017 IEEE Symposium on Security and Privacy (SP)*, 3–18. https://doi.org/10.1109/SP.2017.41.

Sordoni, Alessandro, Xingdi Yuan, Marc-Alexandre Côté, Matheus Pereira, Adam Trischler, Ziang Xiao, Arian Hosseini, Friederike Niedtner, and Nicolas Le Roux. 2023. "Joint Prompt Optimization of Stacked LLMs Using Variational Inference." https://arxiv.org/abs/2306.12509.

Tang, Xinyu, Richard Shin, Huseyin A. Inan, Andre Manoel, Fatemehsadat Mireshghallah, Zinan Lin, Sivakanth Gopi, Janardhan Kulkarni, and Robert Sim. 2024. "Privacy-Preserving in-Context Learning with Differentially Private Few-Shot Generation." https://arxiv.org/abs/2309.11765.

Wu, Tong, Ashwinee Panda, Jiachen T. Wang, and Prateek Mittal. 2023. "Privacy-Preserving in-Context Learning for Large Language Models." https://arxiv.org/abs/2305.01639.

Xie, Chulin, Zinan Lin, Arturs Backurs, Sivakanth Gopi, Da Yu, Huseyin A Inan, Harsha Nori, et al. 2024. "Differentially Private Synthetic Data via Foundation Model APIs 2: Text." https://arxiv.org/abs/2403.01749.

Yue, Xiang, Huseyin A. Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. 2023. "Synthetic Text Generation with Differential Privacy: A Simple and Practical Recipe." https:

//arxiv.org/abs/2210.14348.

Zeng, Shenglai, Jiankun Zhang, Pengfei He, Jie Ren, Tianqi Zheng, Hanqing Lu, Han Xu, Hui Liu, Yue Xing, and Jiliang Tang. 2024. "Mitigating the Privacy Issues in Retrieval-Augmented Generation (RAG) via Pure Synthetic Data." https://arxiv.org/abs/2406.14773.

Zhou, Yongchao, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2023. "Large Language Models Are Human-Level Prompt Engineers." https://arxiv.org/abs/2211.01910.