Differentially Private In-Context Learning with Nearest Neighbor Search

Antti Koskela
Nokia Bell Labs

Tejas Kulkarni
Nokia Bell Labs

Laith Zumot
Nokia

Abstract

Differentially private in-context learning (DP-ICL) has recently become an active research topic due to the inherent privacy risks of in-context learning. However, existing approaches overlook a critical component of modern large language model (LLM) pipelines: the similarity search used to retrieve relevant context data. In this work, we introduce a DP framework for in-context learning that integrates nearest neighbor search of relevant examples in a privacy-aware manner. Our method outperforms existing baselines by a substantial margin across all evaluated benchmarks, achieving more favorable privacy-utility trade-offs. To achieve this, we employ nearest-neighbor retrieval from a database of context data, combined with a privacy filter that tracks the cumulative privacy cost of selected samples to ensure adherence to a central differential privacy budget. Experimental results on text classification and document question answering show a clear advantage of the proposed method over existing baselines.

1 Introduction

In-context learning (ICL) (Brown et al., 2020) is a popular way to tailor a generic language model's response to a specific context/domain. A typical ICL pipeline involves first preparing a guiding prompt that contains several task related examples, such as question-answer pairs, and then asking the language model to generate a response for the query, conditioned on the examples provided. A key feature of ICL is that it does not involve compute heavy operations of updating model weights and typically API or prompt-only access to LLM is sufficient.

Privacy risks in LLMs due to memorization are well known (Zhang et al., 2023; Carlini et al., 2023; Ippolito et al., 2023). One line of research deals with leakage in fine-tuning (Yu et al., 2024; Li et al., 2022) or pretraining (Carlini et al., 2019). Another line of research attempts to recover training records using clever prompt engineering (Davison et al., 2019; Jiang et al., 2020; Nasr et al., 2025). Specifically for ICL, Duan et al. (2024); Wen et al. (2024); Duan et al. (2023) have proposed membership inference attacks to detect the membership of a test data point in a private prompt.

Differentially Private In-Context Learning (DP-ICL) is an active area of research, currently being explored along two parallel directions:

- DP Synthetic example generators (Tang et al., 2024; Gao et al., 2025; Amin et al., 2024): These methods generate synthetic examples token-by-token by privately releasing the mean logits from several partitions sensitive examples. The next token with the largest weight is selected as the next token. The generated examples can be used as demonstrations in multiple downstream ICL tasks without incurring additional privacy costs. While one-time

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

privacy costs is an attractive feature, these methods are computationally expensive and rely on logit outputs from the LLMs, which may not be easily available in many scenarios. Moreover, experiments in these works are limited to simpler tasks such text classification and information extraction.

- Pay-per-use (Wu et al., 2024): The private set of examples is partitioned into k shards, each of which is associated with a given prompt. The model generates one response for each of the k shards. For text classification, the final output is released via private voting using the shard responses. For text generation problems, the private responses are aggregated in either keyword or embedding space and released privately. LLM is then asked to provide the final response based on the top keywords or mean embeddings. While the privacy cost scales with the number of test queries answered, the method is easy to parallelize, does not require access to logits and has the capacity to generate very high-quality responses in a wide range of tasks. Our work improves on this method.

A related line of work, often referred to as private prediction, studies how to obtain differentially private predictions from non-private models (Dwork and Feldman, 2018; Papernot et al., 2018; Bassily et al., 2018; Zhu et al., 2020, 2023). Methods in this class typically perturb model outputs or the voting scores, and some of them also use k-nearest-neighbor (kNN) search (Zhu et al., 2020, 2023). Interestingly, both the DP Synthetic Example Generators the Pay-per-use methods can be viewed as instances of this broader private prediction framework: they provide privacy guarantees only at the prediction stage, while reusing non-private models. However, kNN methods have not yet been incorporated into the DP-ICL setting, which has unique characteristics due to the compositional and prompt-based nature of in-context learning.

The methods by Wu et al. (2024) use Poisson sampling to select the examples for the shards. While sampling amplifies the privacy protection, it can pick examples unrelated to the test query. It has been well-documented (Lu et al., 2022; Agrawal et al., 2023; Bölücü et al., 2023) that the output of ICL is sensitive to the examples used, and randomly sampled examples can lead to increased prediction uncertainty, potentially resulting in worse performance compared to 0-shot predictions. Therefore, example selection has emerged as an important research direction in ICL (Dong et al., 2024).

We also emphasize, that the embedding based kNN search of demonstrations is a standard component in information retrieval systems such as those designed for retrieval augmented generation (RAG) (Liu et al., 2022) and can be easily plugged into an existing ICL pipeline. Surprisingly, we are unaware of any work on DP-ICL that uses kNN indexing despite their popularity.

1.1 Our contributions

- Through use of privacy filters (Feldman and Zrnic, 2021), we integrate nearest neighbor search into existing DP-ICL framework by Wu et al. (2024). The modified solution composes prompts with k-nearest neighbors of each test point instead of randomly sampled examples like in the baseline methods by Wu et al. (2024) and Tang et al. (2024).

- As a theoretical contribution, we provide a fully adaptive δ-approximate RDP analysis of so called individual RDP filters.

- We carry out experiments on text classification and question answering on benchmark datasets with LLMs such as Llama3.3-70B-it and Gemini-1.5-flash-8B. Our experiments clearly show that the overall privacy-utility trade-off is drastically improved with kNN.

2 DP-ICL with kNN

We give the required background on DP and the problem setting of ICL in Appendix Section B. Our method is based on the basic primitives of report-noisy-max with Gaussian noise (RNM-Gaussian) and DP keyword space aggregation (DP-KSA) that are also the building blocks of the baseline methods by Wu et al. (2024). Those methods are described in detail in Appendix Section C. We next describe how to combine those methods with kNN nearest neighbor search of examples.

2.1 Retrieval of Most Similar Examples

Instead of retrieving examples via subsampling, we combine the RNM-Gaussian and DP-KSA mechanisms with retrieval of the k most similar examples from the sensitive dataset X. A similar approach is taken by Zhu et al. (2023) for private prediction, though not in combination with in-context learning. Specifically, for each query q, we construct the retrieved set $\mathcal{R}(X)$ by selecting the k elements $x_i \in X$ most similar to q under a chosen similarity metric. The sampled set is then partitioned into M disjoint batches $B_1, \ldots, B_M$ for in-context prompting.

2.2 Retrieval with Limited Sensitivity

The challenge with kNN retrieval is how to carry out the privacy accounting. The required tool is given by the individual RDP accounting (Feldman and Zrnic, 2021). To this end, we also require from the retrieval function $\mathcal{R}$ that its output can change at most by one element in case we change the dataset X by one element. More formally, the output of the LLM $\mathcal{A}$ consist of the retrieval and the DP-ICL algorithm. So we can think of it as a composition $\mathcal{A} = \mathcal{M} \circ \mathcal{R}$, where $\mathcal{M}$ is the DP-ICL mechanism (e.g., DP-KSA) that takes as an input the set of batches $\{B_1, \ldots, B_M\}$, and $\mathcal{R}(X)$ is the retrieval algorithm that fetches the batches from the input dataset X.

Mathematically, we say that $\mathcal{R}$ is *stable under single-element change* if, whenever $X \simeq X'$, the outputs differ by at most two elements: $|\mathcal{R}(X) \setminus \mathcal{R}(X')| + |\mathcal{R}(X') \setminus \mathcal{R}(X)| \leq 2$. In order to limit the sensitivity of the aggregation happening in the mechanism $\mathcal{A}$, we require this property from the retrieval $\mathcal{R}$. In this work, we focus on the FLAT index for simplicity, as it performs a full exhaustive search and trivially satisfies the mentioned stability property. Extending our proposed method to approximate indexing like IVF or HNSW is a compelling avenue for future work. For example, DP k-means methods (Chang et al., 2021) could be used to implement IVF search, incurring an additional privacy cost while still meeting the stability requirements of the retrieval.

2.3 Individual RDP Accounting for DP-ICL with kNN

The rigorous privacy accounting for DP-ICL with kNN retrieval can be carried out using an individual (α, ε)-RDP privacy filter that keeps track of individual privacy losses and drops from the analysis the data elements for which the cumulative privacy loss is about to cross the pre-determined budget $\varepsilon_{\max}$ (Feldman and Zrnic, 2021). To this end, we first give the following definitions.

Define $\mathrm{Sub}(S, x_i)$ as the set of datasets obtained from S by substituting the data element x_i by another data element, i.e.,

$$\mathrm{Sub}(S, x_i) = \{S' \mid S' = (S \setminus \{x_i\}) \cup \{x_i'\},\ x_i' \in \mathcal{X}\}$$

The individual δ-approximate (α, ε)-RDP privacy filter is described in the pseudocode of Algorithm 1. Notice that in each step t, the adaptively chosen mechanism $\mathcal{A}_t$ is of the form $\mathcal{A}_t = \mathcal{M} \circ \mathcal{R}_t$, where the retrieval function $\mathcal{R}_t$ depends adaptively on the query q_t chosen at iteration t. I.e., the data elements that are used by the DP-ICL mechanism $\mathcal{M}$ at step t, depend on the query and the set of data elements that still have their privacy budget left. When using the DP-KSA algorithm, we fix the iteration-wise failure probability δ_i. However, it could also be chosen adaptively.

The following result is our main theoretical result and is proven in Appendix B.3. It can be seen as a generalization of the RDP filtering result of (Thm. 4.5 Feldman and Zrnic, 2021) and of the δ-approximate zCDP filtering result of (Thm. 1 Whitehouse et al., 2022).

Theorem 1 (Privacy Filter for δ-approximate Rényi Differential Privacy)**.** *Let $K \in \mathbb{Z}_+$ define the maximum number of compositions and let $\{\mathcal{M}_i\}_{i=1}^K$ be an adaptively chosen sequence of randomized mechanisms, where each $\mathcal{M}_i$ is δ_i-approximate $(\alpha, \varepsilon_i(\alpha))$-RDP for some $\alpha \geq 1$. Let $\varepsilon_{\max}(\alpha) > 0$ and $\delta_{\max} \geq 0$ define the privacy budgets. Then, a* privacy filter *that halts when either $\sum_{i=1}^{T+1} \varepsilon_i > \varepsilon_{\max}(\alpha)$ or $\sum_{i=1}^{T+1} \delta_i > \delta_{\max}$ ensures that, the composed mechanism $\mathcal{M}^{(K)} = \big(\mathcal{M}_1, \ldots, \mathcal{M}_K\big)$ is $\delta_{\max}$-approximate $\varepsilon_{\max}(\alpha)$-RDP.*

Similarly, as from the general RDP filters follow results for individual filters (Feldman and Zrnic, 2021), from the general filtering result of Thm. 1 it trivially follows that Algorithm 1 is is $\delta_{\max}$-approximate $(\alpha, \varepsilon_{\max})$-RDP. This privacy guarantee can be then converted to a (ε, δ)-DP guarantee using the conversion formula given in Appendix Eq. (B.1).

Algorithm 1 Adaptive composition $\mathcal{A}^{(T)}$ with Rényi filter

Input: Dataset X, and privacy budget $(\varepsilon_{\max}, \delta_{\max})$.
Set the active set of data elements to be the whole dataset: $S = X$.
for $t = 1, \ldots, T$ **do**
  for all data entries $x_i \in S$ **do**
    Compute individual δ_i-approximate RDP parameters for the chosen mechanism $\mathcal{A}_t$. I.e.,

$$\varepsilon_t^{(i)} = \sup_{S' \in \mathrm{Sub}(S, x_i)} D_\alpha^{\delta_i} \left(\mathcal{A}_t(a^{(t-1)}, S) \| \mathcal{A}_t(a^{(t-1)}, S') \right).$$

  end for
  Update the set S of active data elements:

$$S = \left\{ x_i \;\middle|\; \sum\nolimits_{j=1}^{t} \varepsilon_j^{(i)} \leq \varepsilon_{\max},\ \sum\nolimits_{j=1}^{t} \delta_i \leq \delta_{\max} \right\}.$$

  Compute $a_t = \mathcal{A}_t(a_{1:t-1}, S)$
end for
Return $(a_1, \ldots, a_T)$

3 Experimental Results

3.1 Text Classification

We first evaluate our kNN-based DP-ICL method on public benchmark text classification datasets AGNews (Zhang et al., 2015) and TREC (Voorhees, 2004). For simplicity, we set the privacy parameters such that each sample is used only once. Consistent with Wu et al. (2024), our implementation utilizes 10 shards, each featuring 4 demonstrations. The exact prompt used has been provided in Appendix.

For the nearest neighbor search, we use the "all-MiniLM-L6-v2" model to produce embeddings of unit length with dimension 384. We use FLAT indices for retrievals, which are constructed using FAISS library (Douze et al., 2024).

We carry out the classification experiments using the open-source model OPT-1.3B by Meta that is also available on the Huggingface platform (Wolf et al., 2019). The predictions are generated deterministically. Figure 1 shows the mean test accuracies for an experiment, where we pick 200 randomly sampled test samples for the AGNews and TREC datasets, respectively, for different values of ε, when $\delta = 10^{-5}$. The results for each ε are means of 5 independent runs, and the error bars depict 1.96 times the standard deviation, giving the asymptotic 95% confidence interval. We exclude the zero-shot results from Figure 1 as they were highly unsatisfactory when using OPT-1.3B: we were able to achieve approximately 58% test accuracy for AGNews whereas the results for TREC were close to random guessing. We remark that the test accuracies for the baseline method and for the zero-shot are similar as in (Wu et al., 2024) that uses the GPT-3 Babbage model which also has approximately 1.3B parameters. Figure 1 also includes the "nearest neighbor only (dummy)", where RNM-Gauss is run directly on a histogram formed using the counts of the nearest neighbors' labels.

Notice that on the TREC example of Fig. 1, the performance of DP-ICL with kNN deteriorates as ε decreases towards 0.5. This can be explained by the fact that as the DP are those of a Gaussian mechanism with sensitivity $\sqrt{2}$, for $\varepsilon = 0.5$ the required noise scale is approximately 10 which equals the number of shards (number of votes), which already significantly randomizes the predictions. Naturally, this could be remedied by using more shards.

3.2 Document Question Answering

We next compare the methods on the task of questions answering, on two datasets:

Federated version of DocVQA (Tobaben et al., 2024): This dataset was curated for a competition organized at NeurIPS 23. Each dataset record contains a triplet of the form ⟨image, question, answer⟩. Each image is a sensitive invoice with confidential details (e.g. payer/payee names, invoice amount,

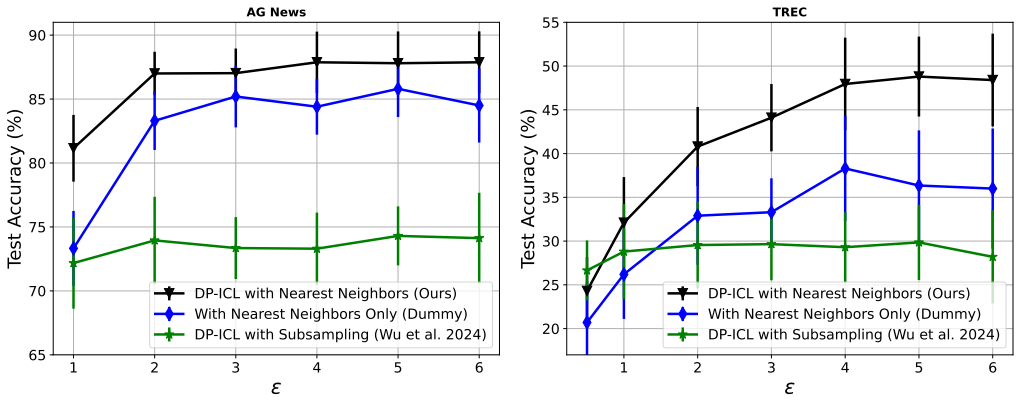

Figure 1: Mean test accuracies for 200 randomly sampled test samples. Left: AGNews text classification task with 4 classes, averaged over 5 experiments. Right: TREC text classification task with 6 classes, averaged over 5 experiments.

purpose). The original task was to answer multiple questions on each image with limited information leakage. The scope of this work is limited to textual ICL. Therefore, we proxy each image with OCR tokens supplied in the same dataset. We decode and concatenate those tokens to form text sentences ignoring their original position in the image. As concatenated sentences may not form a cohesive paragraph this makes it a challenging dataset.

SQUAD v1.1 (Rajpurkar et al., 2016): This is a standard reading comprehension dataset, consisting of questions posed on Wikipedia articles. The records are triplets of the form ⟨paragraph, question, answer⟩.

Both DP-KSA and DP-KSA-kNN satisfy record-level DP which protects presence of a *single* triplet (document, question, answer). However, both datasets contain multiple questions for each image/paragraph. Therefore, we randomly sample a single question-answer pair for each paragraph and assume that each record belongs to a single user.

We continue to use the all-MiniLM-L6-v2 model for embeddings. For DP-KSA-kNN method, we build FLAT index with the text paragraphs using FAISS library.

Dataset	Federated DocVQA	SQUAD
Demonstration Set	69,785	18,891
Test Query Set	100	100

Table 1: Comparison of dataset sizes between Federated DocVQA and SQUAD.

Language models used: The comparison of distribution of prompt lengths for both datasets is shown in Figure 3 of Appendix. We use Llama3.3-70B-it and Gemini-1.5-flash-8B and fix the temperature parameter to 0.7 in our API calls for both models. However, we did not observe much variance in the responses due to 'to-the-point nature' of the questions.

Accuracy metrics: Our performance metrics include standard Rouge and Bleu scores. We have described the metrics for completeness in Table 2 in Appendix. All metrics range from 0 to 1. Higher scores imply a higher degree of similarity between two answers.

Experimental Results: Figure 2 shows plots for document QA task for 4-shot ICL with shard sizes 10 and 20 for several ε's. Plots for the other two (model,dataset) combinations are given in Appendix J. We use the same randomly sampled 100 test queries for all methods and ε's. We also include 0-shot responses (obtained without any demonstrations) computed with the same number of shards. Outperforming this baseline is important for any method to justify the use of private demonstrations. Points with $\varepsilon = \infty$ correspond to non-private version of KSA and KSA-kNN.

The main high-level observation across both figures is that most metrics have higher values for the Llama model compared to Gemini. We also note that DP-KSA remains less sensitive to ε's, whereas, DP-KSA-kNN improves in many cases specially for high ε's. Figure 4 in Appendix show additional results.

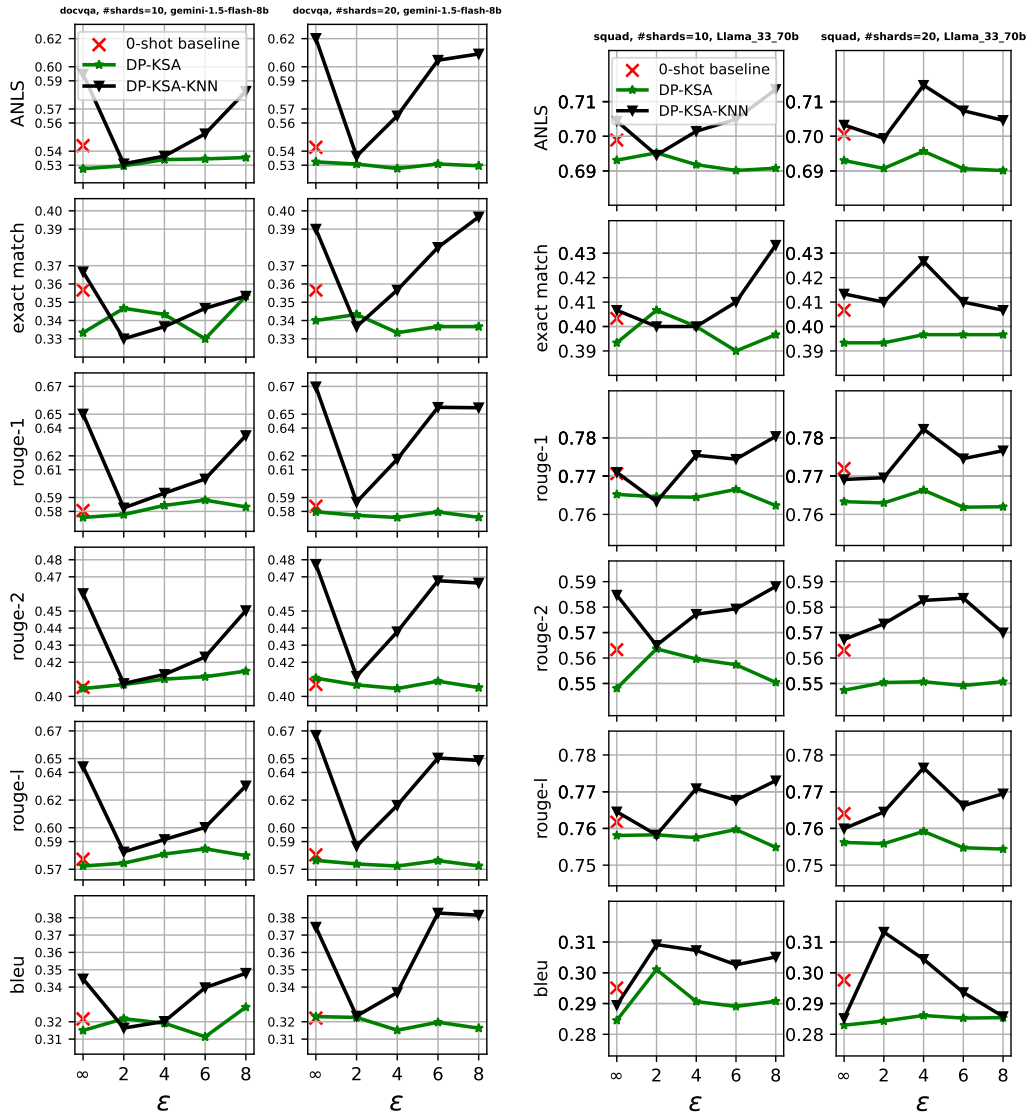

Figure 2: Left: A comparison of DP-KSA and DP-KSA-kNN on a 4-shot Q&A task on the docVQA dataset using Gemini-1.5-flash-8B. Right: A comparison of DP-KSA and DP-KSA-kNN on a 4-shot Q&A task on the SQUAD dataset using the Llama 3.3-70B-It model. The averages are computed over individual metrics for 100 test queries. The higher number indicates a higher degree of similarity between algorithm's final response and ground truth. We see that the proposed method (DP-KSA-kNN) is superior compared to the baseline (DP-KSA).

4 Conclusions

In this work, we integrate nearest neighbor search based indexing into an existing DP-ICL framework. This is obtained by using the so called fully adaptive privacy analysis and individual differential privacy filters. Our experiments on private text classification and private question answering tasks show the substantial advantage of our approach. Our method clearly outperforms the 0-shot and also the DP baseline method by Tang et al. (2024) despite not having the privacy amplification by subsampling as the method by Tang et al. (2024). Interesting research directions in this topic include building DP-ICL solutions utilizing alternative sample indexing and retrieval methods, such as those based on hierarchical clustering like k-means or hierarchical navigable small worlds (HNSW).

References

Agrawal, S., Zhou, C., Lewis, M., Zettlemoyer, L., and Ghazvininejad, M. (2023). In-context examples selection for machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8857–8873. Association for Computational Linguistics.

Amin, K., Bie, A., Kong, W., Kurakin, A., Ponomareva, N., Syed, U., Terzis, A., and Vassilvitskii, S. (2024). Private prediction for large-scale synthetic text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7244–7262.

Balle, B. and Wang, Y.-X. (2018). Improving the Gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*, pages 394–403.

Bassily, R., Thakkar, O., and Thakurta, A. G. (2018). Model-agnostic private learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31.

Bölücü, N., Rybinski, M., and Wan, S. (2023). impact of sample selection on in-context learning for entity extraction from scientific writing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Amodei, D., Radford, A., Sutskever, I., and Clark, J. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Bun, M. and Steinke, T. (2016). Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pages 635–658. Springer.

Carlini, N., Ippolito, D., Jagielski, M., Lee, K., Tramèr, F., and Zhang, C. (2023). Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023*. OpenReview.net.

Carlini, N., Liu, C., Erlingsson, U., Kos, J., and Song, D. (2019). The secret sharer: evaluating and testing unintended memorization in neural networks. In *Proceedings of the 28th USENIX Conference on Security Symposium*, SEC'19, page 267–284. USENIX Association.

Cesar, M. and Rogers, R. (2021). Bounding, concentrating, and truncating: Unifying privacy loss composition for data analytics. In *Algorithmic Learning Theory*, pages 421–457. PMLR.

Chang, A., Ghazi, B., Kumar, R., and Manurangsi, P. (2021). Locally private k-means in one round. In *International conference on machine learning*, pages 1441–1451. PMLR.

Davison, J., Feldman, J., and Rush, A. M. (2019). Commonsense knowledge mining from pretrained models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, 2019*.

Dong, Q., Li, L., Dai, D., Zheng, C., Ma, J., Li, R., Xia, H., Xu, J., Wu, Z., Chang, B., Sun, X., Li, L., and Sui, Z. (2024). A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*.

Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.-E., Lomeli, M., Hosseini, L., and Jégou, H. (2024). The faiss library.

Duan, H., Dziedzic, A., Papernot, N., and Boenisch, F. (2023). Flocks of stochastic parrots: Differentially private prompt learning for large language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Duan, H., Dziedzic, A., Yaghini, M., Papernot, N., and Boenisch, F. (2024). On the privacy risk of in-context learning.

Dwork, C. and Feldman, V. (2018). Privacy-preserving prediction. In *Proceedings of Machine Learning Research, Conference on Learning Theory*, pages 1693–1702.

Dwork, C. and Roth, A. (2014). The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407.

Feldman, V. and Zrnic, T. (2021). Individual privacy accounting via a Rényi filter. *Advances in Neural Information Processing Systems*, 34.

Gao, F., Zhou, R., Wang, T., Shen, C., and Yang, J. (2025). Data-adaptive differentially private prompt synthesis for in-context learning. In *The Thirteenth International Conference on Learning Representations*.

Gillenwater, J., Joseph, M., Munoz, A., and Diaz, M. R. (2022). A joint exponential mechanism for differentially private top-k. In *International Conference on Machine Learning*, pages 7570–7582. PMLR.

Guo, R., Luan, X., Xiang, L., Yan, X., Yi, X., Luo, J., Cheng, Q., Xu, W., Luo, J., Liu, F., et al. (2022). Manu: a cloud native vector database management system. *Proceedings of the VLDB Endowment*, 15(12):3548–3561.

Hao, W. and Zhang, H. (2024). Faster differentially private top-k selection: A joint exponential mechanism with pruning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Ippolito, D., Tramer, F., Nasr, M., Zhang, C., Jagielski, M., Lee, K., Choquette Choo, C., and Carlini, N. (2023). Preventing generation of verbatim memorization in language models gives a false sense of privacy. In *Proceedings of the 16th International Natural Language Generation Conference*. Association for Computational Linguistics.

Jiang, Z., Xu, F. F., Araki, J., and Neubig, G. (2020). How can we know what language models know. *Trans. Assoc. Comput. Linguistics*, 8:423–438.

Johnson, J., Douze, M., and Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Koga, T., Wu, R., and Chaudhuri, K. (2024). Privacy-preserving retrieval augmented generation with differential privacy. *arXiv preprint arXiv:2412.04697*.

Li, X., Tramèr, F., Liang, P., and Hashimoto, T. (2022). Large language models can be strong differentially private learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*.

Liu, J., Shen, D., Zhang, Y., Dolan, B., Carin, L., and Chen, W. (2022). What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*. Association for Computational Linguistics.

Lu, Y., Bartolo, M., Moore, A., Riedel, S., and Stenetorp, P. (2022). Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098.

Mironov, I. (2017). Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275.

Nasr, M., Rando, J., Carlini, N., Hayase, J., Jagielski, M., Cooper, A. F., Ippolito, D., Choquette-Choo, C. A., Tramèr, F., and Lee, K. (2025). Scalable extraction of training data from aligned, production language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025*.

Papernot, N., Song, S., Mironov, I., Raghunathan, A., Talwar, K., and Úlfar Erlingsson (2018). Scalable private learning with pate. In *International Conference on Learning Representations (ICLR)*. arXiv preprint arXiv:1802.08908.

Papernot, N. and Steinke, T. (2022). Hyperparameter tuning with renyi differential privacy. In *International Conference on Learning Representations*.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). 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.

Tang, X., Shin, R., Inan, H. A., Manoel, A., Mireshghallah, F., Lin, Z., Gopi, S., Kulkarni, J., and Sim, R. (2024). Privacy-preserving in-context learning with differentially private few-shot generation. In *The Twelfth International Conference on Learning Representations*.

Tobaben, M., Souibgui, M. A., Tito, R., Nguyen, K., Kerkouche, R., Jung, K., Jälkö, J., Kang, L., Barsky, A., d'Andecy, V. P., et al. (2024). Neurips 2023 competition: Privacy preserving federated learning document vqa. *arXiv preprint arXiv:2411.03730*.

Voorhees, E. M. (2004). Overview of the trec 2004 robust retrieval track. In *Proceedings of The Thirteenth Text REtrieval Conference, TREC 2004*, volume 500-261 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).

Wang, J., Yi, X., Guo, R., Jin, H., Xu, P., Li, S., Wang, X., Guo, X., Li, C., Xu, X., et al. (2021). Milvus: A purpose-built vector data management system. In *Proceedings of the 2021 International Conference on Management of Data*, pages 2614–2627.

Wen, R., Li, Z., Backes, M., and Zhang, Y. (2024). Membership inference attacks against in-context learning. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, CCS 2024*. ACM.

Whitehouse, J., Ramdas, A., Rogers, R., and Wu, Z. S. (2022). Fully adaptive composition in differential privacy. *arXiv preprint arXiv:2203.05481*.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Wu, T., Panda, A., Wang, J. T., and Mittal, P. (2024). Privacy-preserving in-context learning for large language models. In *The Twelfth International Conference on Learning Representations*.

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

Zhang, C., Ippolito, D., Lee, K., Jagielski, M., Tramèr, F., and Carlini, N. (2023). Counterfactual memorization in neural language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NeurIPS '23.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, volume 28.

Zhu, Y. et al. (2020). Private-knn: Practical differential privacy for computer vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhu, Y. and Wang, Y. (2022). Adaptive private-k-selection with adaptive K and application to multi-label PATE. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2022*, Proceedings of Machine Learning Research. PMLR.

Zhu, Y., Zhao, X., Guo, C., and Wang, Y.-X. (2023). Private prediction strikes back! private kernelized nearest neighbors with individual rényi filter. In *Uncertainty in Artificial Intelligence*, pages 2586–2596. PMLR.

A Further Related Literature

Private in-context learning is typically applied to tasks such as text classification, question answering, and summarization. It often relies on techniques that privatize the counts of output tokens, using either additive noise mechanisms or top-k mechanisms (Tang et al., 2024; Wu et al., 2024). There are data-adaptive refinements of the methdod by Tang et al. (2024), given by Gao et al. (2025), and recently also by Amin et al. (2024) who use the accurate concentrated differential privacy accounting presented in (Cesar and Rogers, 2021) for the exponential mechanism to improve the privacy-utility trade-offs. Recent paper (Koga et al., 2024) considers a DP RAG method, however does not seem to incorporate the ranking of the augmenting samples into the DP mechanism. Several papers consider a version of exponential mechanism tailored for private top-k selection (Gillenwater et al., 2022) called jointEM. Recently, faster version of jointEM has been proposed by Hao and Zhang (2024). Another work that is close to our work is that by Zhu et al. (2023) who also consider individual privacy accounting and kNN similarity search for private prediction although in a way that is not directly applicable to existing DP-ICL methods.

A.1 Existing Indexing Methods for Similarity Search

Commonly used nearest neighbor search methods used in ICL and RAGs include

- FLAT, Brute-force search where all vectors are stored and compared exhaustively. Suitable for small datasets but not scalable for large-scale search.
- IVF (Inverted File Index) Partitions the dataset into clusters (Voronoi cells) using the k-means algorithm. During a search, only a subset of clusters is probed to reduce computation. Efficient but requires careful tuning of the number of clusters
- HNSW (Hierarchical Navigable Small World) Graph-based indexing where points are connected in a proximity graph. Provides fast nearest neighbor search with logarithmic complexity.

These are the main methods of the widely used similarity search libraries Faiss (Facebook AI Similarity Search) (Johnson et al., 2019; Douze et al., 2024) and Milvus (Wang et al., 2021; Guo et al., 2022). As outlined by Douze et al. (2024), vector search systems must navigate trade-offs between search accuracy, speed, and memory consumption, which depend heavily on dataset size, vector dimensionality, and the chosen index architecture. Indexing methods like FLAT, IVF and HNSW can be deployed on both CPU and GPU hardware, providing flexibility to optimization in different application contexts.

B Problem Setting and Background

B.1 In-Context Learning

We have a private dataset $X = (x_1, x_2, \cdots, x_N) \in \mathcal{X}^N$ of demonstrations, where $x_i \in [N]$ consists of the content (e.g. an article for classification or a tuple of text description and a related question for QA task) and possibly some ground truth (e.g. label, answer, text summary). We also have a prompt only access to a pretrained (autoregressive) language model LM with a large enough context window. We also have a function $\mathcal{R}$ that retrieves a subset of X as few-shot examples. Given a query content q (e.g. news article), we aim to generate the answer tokens A (e.g. class label, answer to a question) $\operatorname{argmax}_A \mathrm{LM}(A|\mathcal{R}(X) + q)$ in a differentially private manner. The sign '+' denotes the concatenation operation. Specifically, we want to use X to learn the mapping between x's and y's and improve over 0-shot prediction $\operatorname{argmax}_A \mathrm{LM}(A|q)$ for an unknown query q. We further assume that client and ICL server interact only once for a single query, and LM does not retain previous interactions with the same client.

B.2 Differential Privacy

We say input sets X and X' are neighbours if we get one by substituting one element in the other (denoted $X \sim X'$).

A mechanism $\mathcal{M}$ is (ε, δ)-DP if its outputs are (ε, δ)-indistinguishable for neighbouring datasets.

Definition 2. *Let $\varepsilon \geq 0$ and $\delta \in [0,1]$. Mechanism $\mathcal{M} : \mathcal{X}^n \to \mathcal{O}$ is (ε, δ)-DP if for every pair of neighbouring datasets X, X', every measurable set $E \subset \mathcal{O}$,*

$$\mathbb{P}(\mathcal{M}(X) \in E) \leq \mathrm{e}^{\varepsilon}\mathbb{P}(\mathcal{M}(X') \in E) + \delta.$$

We will also use the Rényi differential privacy (RDP) (Mironov, 2017) which is defined as follows. Rényi divergence of order $\alpha \in (1, \infty)$ between two distributions P and Q is defined as

$$D_\alpha(P||Q) = \frac{1}{\alpha - 1} \log \int \left(\frac{P(t)}{Q(t)}\right)^\alpha Q(t)\, \mathrm{d}t.$$

By continuity, we have that $\lim_{\alpha \to 1+} D_\alpha(P||Q)$ equals the KL divergence $KL(P||Q)$.

Definition 3. *We say that a mechanism $\mathcal{M}$ is (α, ε)-RDP, if for all neighbouring datasets X, X', the output distributions $\mathcal{M}(X)$ and $\mathcal{M}(X')$ have Rényi divergence of order α less than ε, i.e.,*

$$\max_{X \simeq X'} \{D_\alpha\big(\mathcal{M}(X)||\mathcal{M}(X')\big), D_\alpha\big(\mathcal{M}(X')||\mathcal{M}(X)\big)\} \leq \varepsilon.$$

Certain applications, like the Propose-Test-Release framework we consider, require a relaxation of RDP that allows a small probability of failure. To address this, we consider a δ-approximate version of RDP, which extends the definition to account for a negligible additive failure probability δ.

Definition 4. *We say a randomized algorithm $\mathcal{M}$ is δ-approximately $(\alpha, \varepsilon(\alpha))$-RDP with order $\alpha \geq 1$, if for all neighboring dataset X, X', there exist events E (depending on $\mathcal{M}(X)$) and E' (depending on $\mathcal{M}(X')$) such that $\Pr[E] \geq 1 - \delta$ and $\Pr[E'] \geq 1 - \delta$, and we have*

$$D_\alpha(\mathcal{M}(D)|E \,\|\, \mathcal{M}(D')|E') \leq \varepsilon.$$

We remark that in the application of text classification, we use the common RDP accounting, and for question answering, we need to use the δ-approximate RDP.

B.3 δ-Approximate RDP

We next review some of the properties of the δ-approximate RDP (see, e.g., Bun and Steinke, 2016; Papernot and Steinke, 2022).

First, recall that a randomized algorithm $\mathcal{M} : \mathcal{X}^n \to \mathcal{Y}$ is δ-approximately (α, ε)-Rényi differentially private if, for all neighbouring pairs of inputs $X, X' \in \mathcal{X}^n$, it is (α, ε)-RDP except for a set of measure at most δ. The definition is given more formally as follows.

Definition 5. *We say a randomized algorithm $\mathcal{M}$ is δ-approximately $(\alpha, \varepsilon(\alpha))$-RDP with order $\alpha \geq 1$, if for all neighboring dataset X, X', there exist events E (depending on $\mathcal{M}(X)$) and E' (depending on $\mathcal{M}(X')$) such that $\Pr[E] \geq 1 - \delta$ and $\Pr[E'] \geq 1 - \delta$, and we have*

$$D_\alpha(\mathcal{M}(D)|E \,\|\, \mathcal{M}(D')|E') \leq \varepsilon.$$

If $\mathcal{M}$ is δ-approximate (α, ε)-RDP, we also shortly denote it as

$$D_\alpha^\delta(\mathcal{M}(x)\|\mathcal{M}(x')) \leq \varepsilon.$$

Some basic properties of approximate RDP are as follows (see, e.g., Appendix E, Papernot and Steinke, 2022):

- (ε, δ)-DP is equivalent to δ-approximate (∞, ε)-RDP.
- (ε, δ)-DP implies δ-approximate $(\alpha, \frac{1}{2}\varepsilon^2\alpha)$-RDP for all $\alpha \in (1, \infty)$.
- δ-approximate (α, ε)-RDP implies $(\hat{\varepsilon}, \hat{\delta})$-DP for

$$\hat{\delta} = \delta + \frac{\exp((\alpha - 1)(\hat{\varepsilon} - \varepsilon))}{\alpha} \cdot \left(1 - \frac{1}{\alpha}\right)^{\alpha - 1}. \tag{B.1}$$

- δ-approximate (α, ε)-Rényi differential privacy is closed under postprocessing.

- If $\mathcal{M}_1$ is δ_1-approximately (α, ε_1)-Rényi differentially private and $\mathcal{M}_2$ is δ_2-approximately (α, ε_2)-Rényi differentially private, then their composition is $(\delta_1 + \delta_2)$-approximately $(\alpha, \varepsilon_1 + \varepsilon_2)$-RDP.

The following is a tailored subsampling amplification result for δ-approximate RDP mechanisms, given by Wu et al. (2024). We need it for evaluating the privacy guarantees of the baseline method.

Theorem 6 (Privacy amplification by Poisson subsampling for approximate RDP, Wu et al. (2024))**.** *Let $\mathcal{M}$ be a mechanism satisfying δ-approximate $(\alpha, \varepsilon_M(\alpha))$-RDP. Let $\mathcal{M}_{\text{sub}}$ denote the mechanism that applies $\mathcal{M}$ to a Poisson subsample of the data with sampling probability γ. Then:*

$$\mathcal{M}_{\text{sub}} \textit{ satisfies } \delta'\textit{-approximate } (\alpha,\ \varepsilon_{\text{sub}}(\alpha))\textit{-RDP}$$

where $\delta' = \gamma\,\delta$ and $\varepsilon_{\text{sub}}(\alpha)$ equals the tightest possible amplification bound for an $\varepsilon_M(\alpha)$-RDP mechanism under Poisson sampling, with amplification rate adjusted to $\frac{\gamma(1-\delta)}{1-\gamma\delta}$.

C Baseline DP-ICL Methods

We next describe the baseline private aggregation methods by Wu et al. (2024) upon which our approach builds. They adopt the *Gaussian Report Noisy Max* (RNM), introduced by Zhu and Wang (2022), as one of the mechanisms for privately selecting class labels in classification tasks. For document question answering, where outputs are open-ended and higher dimensional, they operate in a lower-dimensional keyword space, using private mechanisms to identify salient content at the token level. The following two sections describe both methods in detail.

C.1 RNM-Gaussian Mechanism for Text Classification

The RNM-Gaussian mechanism M_σ adds independently sampled Gaussian noise to each bin of the voting histogram $h \in \mathbb{R}^k$ over class labels, where the histogram has global sensitivity $\Delta = \sqrt{2}$ (since a change in one example–query pair affects at most two bins). Specifically:

$$\widetilde{h}_i = h_i + \mathcal{N}(0, \sigma^2), \quad \text{for all } i = 1, \ldots, k,$$

and the privatized response is obtained via:

$$\mathcal{M}(h) = \arg\max_i \widetilde{h}_i.$$

When making T private predictions this way, setting $\sigma = \sqrt{2T \ln(1.25/\delta)}/\varepsilon$ ensures that the sequence of outputs satisfies (ε, δ)-differential privacy (Dwork and Roth, 2014). More accurate privacy bounds can be obtained via RDP or by using so called privacy profiles (Balle and Wang, 2018). In this work, we use RDP for privacy accounting.

C.2 Keyword Space Aggregation (KSA) for Document Question Answering

In the document question answering task, the output A of the LLM consists of natural language tokens (e.g., answers or summaries), rather than a fixed class label. To enable private aggregation in this higher-dimensional output space, we adopt the Keyword Space Aggregation (KSA) method. This approach reduces the complexity of the aggregation by projecting responses into a lower-dimensional token space and performing differentially private selection over salient tokens.

Given a query content q, a retrieval function $\mathcal{R}$ obtains M disjoint subsets of the private dataset X and construct M in-context prompts. I.e., the retrieved set of batches $\mathcal{R}(X) = \{B_i\}_{i=1}^M$, where each disjoint batch B_i contains a number of data points. For each prompt, the output is sampled from the language model:

$$O_i(q) := \mathrm{LM}(q\ +\ B_i),$$

where $O_i(q)$ is the natural language answer generated by the model for the i-th prompt. These outputs are then tokenized to form a frequency histogram $h \in \mathbb{R}^D$ over the vocabulary $\mathcal{V}$ of size D, where each count h_t corresponds to the number of outputs in which token t appears:

$$h_t = \left|\{\, i : \text{token } t \in O_i(q)\,\}\right|, \quad t = 1, \ldots, D.$$

To privately identify the most relevant semantic content, a differentially private mechanism is applied to select the top-k tokens from the histogram h. Depending on the vocabulary size D, they consider the following approach.

Propose-Test-Release (PTR). When D is large or unbounded, a PTR mechanism first privately tests whether the frequency gap $h_{(K)} - h_{(K+1)}$ exceeds a threshold. If the test passes, the top-K tokens are released exactly. To determine K privately, one can also perform a noisy argmax:

$$\arg\max_k \left(h_{(k)} - h_{(k+1)}\right),$$

where $h_{(k)}$ denotes the k-th largest entry in h.

The selected keywords $\{t_1, \ldots, t_K\}$ are then incorporated into a follow-up prompt that guides the language model to generate a coherent final answer. For example, we use a structured template such as:

```
Using the following keywords, answer the question
concisely: t_1, t_2, ..., t_K.
```

An alternatively, exponential mechanism (EM) can also be used to release top keywords, however we found the privacy-utility trade-offs of the PTR-based method superior in our experiments compared to EM-based method.

This procedure ensures that the final output reflects the aggregated knowledge across demonstrations, while differential privacy is guaranteed through token-level mechanisms. The KSA method thus enables private, scalable, and semantically meaningful aggregation in the document QA setting.

C.3 Privacy Amplification via Subsampling

Wu et al. (2024) use Poisson subsampling as the retrieval method $\mathcal{R}$ to select demonstration, and to amplify privacy guarantees. For each query q, for the retrieved set of batches $\mathcal{R}(X)$, each example $x_i \in X$ is included independently with probability γ, and partition the sampled set into M disjoint batches for in-context prompting. This reduces the likelihood of any individual contributing to the final output, leading to improved privacy bounds. In particular, if the aggregation mechanism is (ε, δ)-DP, then the overall mechanism with subsampling satisfies approximately $(\gamma\varepsilon, \gamma\delta)$-DP, under standard amplification results. The accurate privacy accounting can be carried out either using subsampling results for RDP (Appendix Thm. 6).

RNM-Gaussian with subsampling: After subsampling and constructing the class histogram h, Gaussian noise is added as before: $\widetilde{h}_i = h_i + \mathcal{N}(0, \sigma^2)$, with the final output $\arg\max_i \widetilde{h}_i$ satisfying improved privacy due to subsampling.

KSA with subsampling: In the QA setting, we apply the same subsampling step before generating outputs $O_i(q)$ and aggregating tokens into the histogram h. The top-K selection (via PTR) is then performed on the reduced set, benefiting from the same privacy amplification.

Our main contribution is to replace the existing $\mathcal{R}$ of random subsampling with a kNN-based retrieval of the most relevant examples from the database. While this approach sacrifices the privacy amplification benefits of subsampling, it significantly improves the quality of the generated outputs. As a result, we can tolerate higher noise levels in the aggregation step, ultimately yielding a better overall privacy–utility trade-off.

The combination of kNNs and individual RDP has also been used in (Zhu et al., 2023) for private classification with kNN search. However, we consider a completely different and a much broader task of ICL. The method (Zhu et al., 2023) cannot be applied to generative tasks such as question answering. Despite kNN's popularity in non-private ICL/RAG pipelines, no prior work on DP-ICL has considered employing it.

D Propose-Test-Release

In this Section, we give background details on the propose-test-release (PTR) which is also part of the baseline method Wu et al. (2024) and which forms also the basis of our DP-KSA-kNN method.

The main idea of DP-KSA implemented with PTR paradigm is that, for the task of releasing the top-k indices of a voting histogram, if $H(k) - H(k+1) > 2$, then the top-k indices are exactly the same for all neighboring datasets. Thus, we can release them without additional noise in that case. To

ensure this, a DP test of the gap $H(k) - H(k+1)$ has to be carried out. This whole PTR procedure is depicted in Algorithm 2.

Algorithm 2 TopKwithPTR

Require: k – number of top tokens to release; H – histogram of token counts; δ – failure probability

$d_k \leftarrow H(k) - H(k+1)$
$\hat{d}_k \leftarrow \max(2, d_k) + \mathcal{N}(0, 4\sigma^2) - \Phi^{-1}(1-\delta; 0, 2\sigma)$
if $\hat{d}_k > 2$ **then**
    return exact top-k tokens
else
    return Terminate (or fallback to zero-shot learning)
end if

The utility can be further optimized, by selecting k that maximizes the gap $H(k) - H(k+1)$ in a privacy-preserving way using the exponential mechanism. This is depicted in Algorithm 3.

Algorithm 3 FindBestK

Require: H – histogram of token counts
for $k = 1$ to $N-1$ **do**
    $d_k \leftarrow H(k) - H(k+1)$
end for
return $\arg\max_k \left(d_k + r(k) + \text{Gumbel}(4/\varepsilon)\right)$

For Algorithm 2, we have the following privacy guarantee given in (Thm. 11, Wu et al., 2024).

Theorem 7 (TopKwithPTR Privacy Guarantee)**.** *Let H be the histogram of a set of i.i.d. samples from a bounded-support distribution. Let $H(k)$ denote the k-th largest value in H, and suppose we want to release the top-k indices of H only if $H(k) - H(k+1) > 2$. Define*

$$\hat{d}_k := \max(2, d_k) + \mathcal{N}(0, 4\sigma^2) - \Phi^{-1}(1-\delta; 0, 2\sigma),$$

where $\Phi^{-1}(\cdot; 0, 2\sigma)$ is the inverse CDF of a Gaussian distribution with mean 0 and standard deviation 2σ. Then the mechanism that releases the top-k indices if $\hat{d}_k > 2$ satisfies δ-approximate $\left(\alpha, \frac{\alpha}{2\sigma^2}\right)$-RDP for all $\alpha \geq 1$.

The following privacy guarantee is a standard result for the exponential mechanism, based on the "Gumbel max trick" which means that the implementation of the exponential mechanism is equivalent to running report noisy max with properly scaled additive Gumbel distributed noise (Remark 3.1, Dwork and Roth, 2014). In practice, when using FindBestK for DP-KSA, we set $k_{\max} = 30$ and $k_{\min} = 15$.

Theorem 8 (FindBestK Privacy Guarantee)**.** *Let $d_k := H(k) - H(k+1)$ for $k = 1, \ldots, N-1$, and define $r(k)$ to be a regularizer such that $r(k) = -\infty$ for $k > k_{\max}$ or $k < k_{\min}$, and $r(k) = 0$ otherwise. Then the mechanism*

$$\arg\max_k \left(d_k + r(k) + \textit{Gumbel}(4/\varepsilon)\right)$$

satisfies ε-differential privacy.

E Fully Adaptive δ-Approximate RDP Accounting (Proof of Thm 1)

Theorem 9 (Privacy Filter for δ-Approximate Rényi Differential Privacy)**.** *Let $K \in \mathbb{Z}_+$ define the maximum number of compositions and let $\{\mathcal{M}_i\}_{i=1}^{K}$ be an adaptively chosen sequence of randomized mechanisms, where each $\mathcal{M}_i$ is δ_i-approximate $(\alpha, \varepsilon_i(\alpha))$-RDP for some $\alpha \geq 1$. Let $\varepsilon_{\max}(\alpha) > 0$ and $\delta_{\max} \geq 0$ define the privacy budgets. Then, a* privacy filter *that halts when either*

$$\sum_{i=1}^{T+1} \varepsilon_i > \varepsilon_{\max}(\alpha) \quad \textit{or} \quad \sum_{i=1}^{T+1} \delta_i > \delta_{\max}$$

ensures that, the composed mechanism $\mathcal{M}^{(K)} = \big(\mathcal{M}_1, \ldots, \mathcal{M}_K\big)$ *is* $\delta_{\max}$*-approximate* $\varepsilon_{\max}(\alpha)$*-RDP.*

Proof. We use here the notation used in (Feldman and Zrnic, 2021). For $n \in [K]$ and for two neighboring datasets X and X', denote

$$y^{(n)} = (y_1, \ldots, y_n),$$
$$\mathcal{M}^{(n)}(X) = \big(\mathcal{M}_1(X, \varepsilon_1), \mathcal{M}_2(\mathcal{M}_1(X), X, \varepsilon_2), \ldots, \mathcal{M}_n(\mathcal{M}_1(X), \ldots, \mathcal{M}_{n-1}(X), X, \varepsilon_n)\big),$$
$$\mathrm{Loss}^{(n)}(y^{(n)}; X, X', \alpha) = \left(\frac{\mathbb{P}(\mathcal{M}^{(n)}(X) = y^{(n)})}{\mathbb{P}(\mathcal{M}^{(n)}(X') = y^{(n)})}\right)^{\alpha},$$

and

$$\mathrm{Loss}_n(y^{(n)}; X, X', \alpha) = \left(\frac{\mathbb{P}(\mathcal{M}_n(y^{(n-1)}, X, \varepsilon_n) = y_n)}{\mathbb{P}(\mathcal{M}_n(y^{(n-1)}, X', \varepsilon_n) = y_n)}\right)^{\alpha}.$$

Since ε_n depends only on $y^{(n-1)}$ (not directly on the dataset), by the Bayes rule we have that

$$\mathrm{Loss}^{(n)}(y^{(n)}; X, X', \alpha) = \mathrm{Loss}^{(n-1)}(y^{(n-1)}; X, X', \alpha) \cdot \mathrm{Loss}_n(y^{(n)}; X, X', \alpha).$$

We next analyze the δ-approximate RDP of the fully adaptive composition, using similar techniques as used in the proof of (Thm. 3.1, Feldman and Zrnic, 2021). In the RDP integrals below, $y^{(K)}$ is distributed according to $\mathcal{M}^{(K)}(X')$, and by "with probability at least 1-δ" we mean that the given RDP bound holds except with probability at most δ over the randomness of $\mathcal{M}^{(K)}(X')$.

Straightforward calculation then shows that

$$\begin{aligned}
&\mathbb{E}_{y^{(K)} \mid \sum_{i=1}^K \varepsilon_i \le \varepsilon_{\max} \text{ and } \sum_{i=1}^K \delta_i \le \delta_{\max}} \left(\frac{\mathbb{P}(\mathcal{M}^{(K)}(X) = y^{(K)})}{\mathbb{P}(\mathcal{M}^{(K)}(X') = y^{(K)})}\right)^{\alpha} \\
=&\mathbb{E}_{y^{(K)}} \left[\mathrm{Loss}^{(K)}(y^{(K)}; X, X', \alpha) \middle| \sum_{i=1}^K \varepsilon_i \le \varepsilon_{\max} \text{ with probability at least } 1 - \sum_{i=1}^K \delta_i \ge 1 - \delta_{\max}\right] \\
=&\mathbb{E}_{y^{(K)}} \left[\mathrm{Loss}^{(K)}(y^{(K)}; X, X', \alpha) \middle| \varepsilon_K \le \varepsilon_{\max} - \sum_{i=1}^{K-1} \varepsilon_i \text{ with probability at least } 1 - \delta_K \ge 1 - \delta_{\max} + \sum_{i=1}^{K-1} \delta_i\right] \\
=&\mathbb{E}_{y^{(K-1)}} \mathbb{E}_{y_K} \Bigg[\mathrm{Loss}^{(K-1)}(y^{(K-1)}; X, X', \alpha) \cdot \mathrm{Loss}_K(y^{(K)}; X, X', \alpha) \Bigg| \\
&\qquad \varepsilon_K \le \varepsilon_{\max} - \sum_{i=1}^{K-1} \varepsilon_i \text{ with probability at least } 1 - \delta_K \ge 1 - \delta_{\max} + \sum_{i=1}^{K-1} \delta_i\Bigg]
\end{aligned}$$

This implies that

$$\begin{aligned}
&\mathbb{E}_{y^{(K)} \mid \sum_{i=1}^K \varepsilon_i \le \varepsilon_{\max} \text{ and } \sum_{i=1}^K \delta_i \le \delta_{\max}} \left(\frac{\mathbb{P}(\mathcal{M}^{(K)}(X) = y^{(K)})}{\mathbb{P}(\mathcal{M}^{(K)}(X') = y^{(K)})}\right)^{\alpha} \\
\le&\mathbb{E}_{y^{(K-1)}} \left[\mathrm{Loss}^{(K-1)}(y^{(K-1)}; X, X', \alpha)\right] e^{(\alpha-1)\left(\varepsilon_{\max} - \sum_{i=1}^{K-1} \varepsilon_i\right)} \text{ with probability at least } 1 - \delta_{\max} + \sum_{i=1}^{K-1} \delta_i
\end{aligned}$$

Continuing, and using the fact that

$$\begin{aligned}
&\mathbb{E}_{y^{(K-1)}} \left[\mathrm{Loss}^{(K-1)}(y^{(K-1)}; X, X', \alpha)\right] \\
&= \mathbb{E}_{y^{(K-2)}} \mathbb{E}_{y_{K-1}} \left[\mathrm{Loss}^{(K-2)}(y^{(K-1)}; X, X', \alpha) \mathrm{Loss}_{K-1}(y^{(K-1)}; X, X', \alpha)\right],
\end{aligned}$$

we have that

$$\begin{aligned}
&\mathbb{E}_{y^{(K)} \mid \sum_{i=1}^K \varepsilon_i \le \varepsilon_{\max} \text{ and } \sum_{i=1}^K \delta_i \le \delta_{\max}} \left(\frac{\mathbb{P}(\mathcal{M}^{(K)}(X) = y^{(K)})}{\mathbb{P}(\mathcal{M}^{(K)}(X') = y^{(K)})}\right)^{\alpha} \\
\le&\mathbb{E}_{y^{(K-2)}} \left[\mathrm{Loss}^{(K-2)}(y^{(K-1)}; X, X', \alpha) \mathrm{Loss}_{K-1}(y^{(K-1)}; X, X', \alpha)\right] e^{(\alpha-1)\left(\varepsilon_{\max} - \sum_{i=1}^{K-2} \varepsilon_i\right)} \\
&\qquad \text{with probability at least} \quad 1 - \delta_{\max} + \sum_{i=1}^{K-2} \delta_i
\end{aligned}$$

since

$$\mathbb{E}_{y_{K-1}}\left[\text{Loss}_{K-1}(y^{(K-1)}; X, X', \alpha)\right] \leq e^{(\alpha-1)\varepsilon_{K-1}}$$

with probability at least $1 - \delta_{K-1}$.

Next, continuing integration mechanism by mechanism, we inductively see that with probability at least $1 - \delta_{\max}$, we have that

$$\mathbb{E}_{y^{(K)}}\left[\left(\frac{\mathbb{P}(\mathcal{M}^{(K)}(X) = y^{(K)})}{\mathbb{P}(\mathcal{M}^{(K)}(X') = y^{(K)})}\right)^{\alpha}\right] \leq e^{(\alpha-1)\varepsilon_{\max}}.$$

The conditions $\sum_{i=1}^{K+1} \varepsilon_i \leq \varepsilon_{\max}$ and $\sum_{i=1}^{K+1} \delta_i \leq \delta_{\max}$ hold by construction of the filter. □

F Example LLM Prompt: Template Used for 4-Shot Text Classification

```
Instruction: Classify each article into one of the following categories
separated by comma: class1, class2, .., class_k.
Article: {demo text 1}, Class: {class1} \n
Article: {demo text 2}, Class: {class1} \n
Article: {demo text 3}, Class: {class2} \n
Article: {query text 3}, Class:
```

G Example LLM Prompt: Template Used for 4-Shot QA

For question answering task,we used the following prompt template. The bracketed terms (e.g., `{demo text1}`) indicate placeholders for specific data.

```
Read the text: {demo text1}
Answer the question with at most 4 words: {demo question1}
Do not provide a Yes/No answer: {demo answer1}

Read the text: {demo text2}
Answer the question with at most 4 words: {demo question2}
Do not provide a Yes/No answer: {demo Answer2}

Read the text: {demo text3}
Answer the question with at most 4 words: {demo question3}
Do not provide a Yes/No answer: {demo Answer3}

Read the text: {demo text4}
Answer the question with at most 4 words: {demo question4}
Do not provide a Yes/No answer: {demo Answer4}

Read the text: {query text}
Answer the question with at most 4 words: {query question}
Do not provide a Yes/No answer:
```

H Description of Evaluation Metrics

Metric	Description
ANLS (Average Normalized Levenshtein Similarity)	Based on the Levenshtein (edit) distance, which measures the minimum number of single-character edits needed to convert one string into another. More lenient than usual ROUGE metrics and allows partial credit for semantically correct approximate answers. This was a key metric in the NeurIPS 2023 competition that introduced the federated DocVQA dataset (Tobaben et al., 2024).
Exact Match	The fraction of test queries with final LM responses exactly matching the ground truth answer. Considered important in QA tasks as most answers tend to be at most 3 words.
ROUGE-1	Measures the overlap of unigrams (individual words) between the LM response and the ground truth answer. Counts the number of words in the prediction that also appear in the ground truth.
ROUGE-2	Measures the overlap of bigrams between the LM response and the ground truth answer. Captures more contextual similarity than ROUGE-1.
ROUGE-L	Captures sentence-level structure similarity by finding the longest sequence of words appearing in both LM response and ground truth in the same order.
BLEU (Bilingual Evaluation Understudy)	Computes the proportion of n-grams (1 to 4) in the LM response that appear in the ground truth. The final score is the geometric mean of n-gram precision multiplied by a penalty term for overly concise answers.

Table 2: Description of evaluation metrics used in QA tasks.

I Distributions of Number of Tokens for Q&A tasks

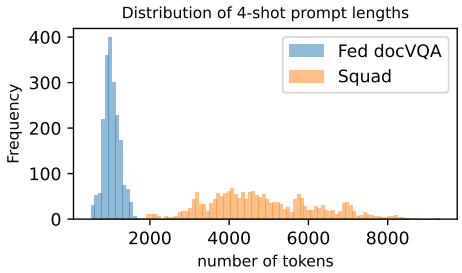

Figure 3: The distributions of number of tokens in the 4 shot prompts (created using demonstration and test examples) when # shards= 20 for two datasets. The prompts for the fed docVQA dataset are longer due to verbose nature of the images, hence many more ocr extracted tokens.

J Further Experimental Results for Q&A tasks

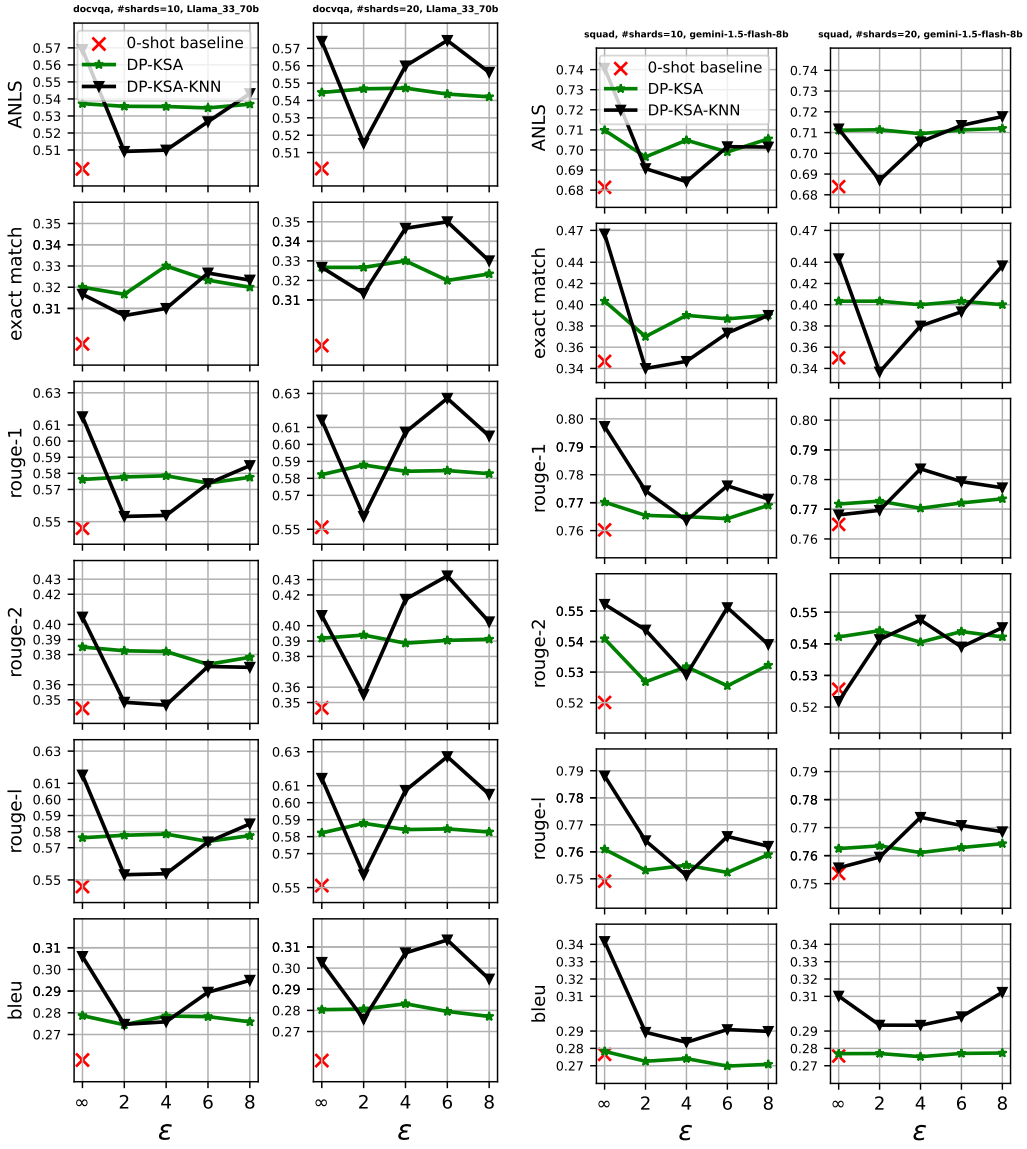

Figure 4: A comparison of DP-KSA and DP-KSA-kNN for average Q&A task metrics. Left: docVQA dataset using Llama 3.3-70B-It. Right: SQUAD dataset using Gemini-1.5-flash-8B.