CONTINUOUSBENCH: CAN DIFFERENTIALLY PRIVATE SYNTHETIC TEXT TRANSFER KNOWLEDGE?

Anonymous authors
Paper under double-blind review

ABSTRACT

Differentially private (DP) text synthesis promises to unlock sensitive corpora for model training, but it remains unclear whether DP synthetic data transmits genuinely new *knowledge* present *only* in those corpora. This is because existing evaluations rely on tasks that are nearly solvable without training, so strong benchmark performance does not establish that DP synthesis can substitute original data access. Thus, we introduce CONTINUOUSBENCH, a continuously and automatically-regenerated benchmark that measures *knowledge transfer* from DP synthetic text. Each year, a new release pairs a never-before-seen training corpus with a derived QA set, constructed to be: (1) unsolvable sans-corpus; and (2) learnable under DP, as the tested knowledge is supported by hundreds of independent records. Researchers produce DP synthetic data from the training corpus and run our standardized training and evaluation harness on their synthetic data to measure gains. We instantiate two tracks: GEMINON, a procedurally-generated dataset about fictional creatures; and NEWS, a stream of newly crawled public news articles. Although standard benchmarks are nearly saturated, on CONTINUOUSBENCH we find that non-private synthesis transfers substantial knowledge from the original corpus, while state-of-the-art DP synthesis methods generally fail to do so, even at $\varepsilon = 100$. We make CONTINUOUSBENCH available at `https://hf.co/AnonymousContinuousBench` to measure and accelerate progress on DP synthetic data.

1 INTRODUCTION

As publicly available text becomes increasingly depleted as a training source for model improvement, much of the remaining high-value data resides in sensitive or proprietary corpora that cannot be directly shared or trained on (e.g., clinical notes, user data). Differentially private (DP) text synthesis offers a compelling alternative in which we can learn from such corpora through a privacy-preserving synthetic release.[1] The key question, however, is not whether DP synthesis can generate stylistically similar text, but rather, whether DP synthetic text can transfer the *knowledge* that access to the original restricted corpus would have provided.

Yet this substitutability question is not answered by existing literature, largely because we lack a *rigorous evaluation framework* that tests whether a given DP synthesis method is capable of preserving the high-value facts learnable from the sensitive corpus. In particular, to claim that DP synthesis can replace access to sensitive data in generality, an evaluation must rule out the possibility of downstream improvements coming from: (1) elicitation of knowledge already present in the base model from public pretraining; (2) the simplicity of the evaluated task, that is, if task success requires only superficial distributional matching rather than learning high-value facts; and (3) teacher-student distillation effects when a stronger generator produces data for a weaker downstream model. We expand on these three confounders in more detail.

First, we seek to address the risk that evaluated knowledge may already be present in the pretrained base model (Tramèr et al., 2024; Arora & Ré, 2022). In this case, strong downstream performance

[1] Throughout, we operate in the standard setting for DP synthetic text generation, namely, central DP with record-level adjacency, where neighboring datasets differ in a single record. We emphasize that local DP or specialized adjacency notions, e.g., PII-level adjacency, is out of the scope of this work.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

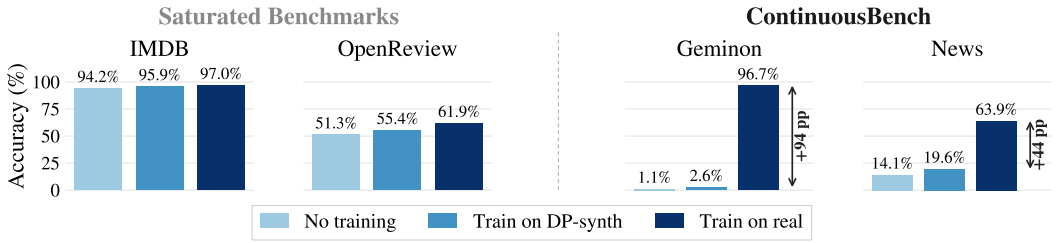

Figure 1: **A saturated benchmark leaves little room to distinguish methods.** We compare downstream accuracy of GEMMA 3 4B after 3 increasingly informative training regimes: no training, training on DP synthetic data ($\varepsilon = 10$), and training on the real corpus. Standard benchmarks are near-saturated, but CONTINUOUSBENCH reveals that DP synthesis has significant headroom.

does not imply that DP synthesis transferred knowledge from the corpus, since we cannot rule out possible elicitation of knowledge already learned during pretraining. Indeed, this is the case for existing evaluations in the literature. Standard benchmark settings based on classification-style utility (Yue et al., 2023; Kurakin et al., 2023; Xie et al., 2024; Tan et al., 2025) are often nearly saturated for modern language models, leaving little room to distinguish methods (see Figure 1). Indeed, many widely used evaluation settings do not provide enough resolution for this comparison. IMDB (Maas et al., 2011), for example, is a binary sentiment-classification task over movie reviews and has been widely used in evaluations of DP text synthesis (Kurakin et al., 2023; Amin et al., 2024; Zou et al., 2025; Banayeeanzade et al., 2026). It illustrates the problem: sentiment classification is a generic skill rather than a corpus-specific knowledge, and modern pretrained models already solve it. Therefore strong performance on such an evaluation does not imply a DP synthesis method can transfer new knowledge.

Second, the evaluated task may not measure the kind of knowledge that makes sensitive data valuable. Metrics commonly used in the DP synthetic text literature, such as MAUVE (Pillutla et al., 2021), next-token prediction accuracy, or multi-class classification accuracy, can improve through style matching without requiring recovery of specific facts or skills from the sensitive corpus. Such metrics are useful diagnostics, but they are poorly suited as measures of desired corpus-specific knowledge transfer. They conflate general linguistic competence with genuine information transfer, and may therefore overstate the practical utility of a DP synthetic corpus. Evaluating DP synthetic text requires tasks that directly test whether the model acquired the high-value knowledge that motivated using the sensitive data in the first place.

Third, synthetic data is often generated by a stronger model and consumed by a weaker one (Yue et al., 2023; Kurakin et al., 2023; Xie et al., 2024; Amin et al., 2024). This makes it difficult to separate gains due to the private corpus from gains due to ordinary distillation from a more capable teacher. Furthermore, such an evaluation does not answer whether sensitive data unlocked via DP synthesis can improve the best model available today.

Contributions. We introduce CONTINUOUSBENCH (illustrated in Figure 2), a benchmark designed specifically to remove the three aforementioned issues with existing evaluations. Automatic releases will couple a never-before-seen training corpus with a derived question-answer (QA) set whose answers are supported by hundreds of independent records in the corpus, making them realistically learnable under DP constraints. In particular, we make the following contributions:

- **Continuously regenerated, access-dependent benchmark.** CONTINUOUSBENCH enforces *access dependency*: the tasks are unsolvable without access to the released training corpus. We implement this by testing the model on either fictional or post-cutoff knowledge. To guarantee access dependency for future model releases, CONTINUOUSBENCH uses an automated curation pipeline allowing for periodic regeneration of new versions with minimal human intervention.
- **Grounded factual QA as the evaluation task.** We focus on short-answer question answering tasks that cannot be solved with surface-level distribution matching. Questions are constructed so that the answer appears across hundreds of independent records, making them learnable under DP. Short-answer factual QA tasks, like MMLU (Hendrycks et al., 2021b) and Natural Questions (Kwiatkowski et al., 2019) allow unambiguous automatic evaluation and have served as reliable proxies for tracking progress in model capabilities.

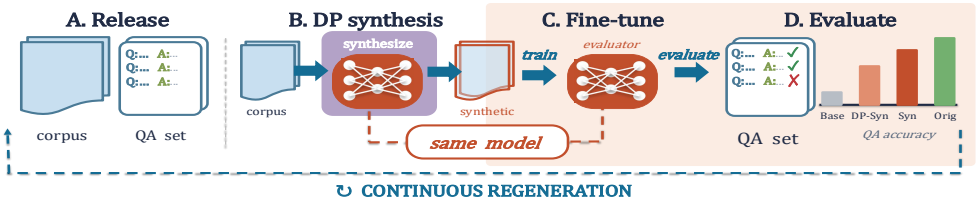

Figure 2: **Workflow of CONTINUOUSBENCH.** Each release (**A**) pairs a freshly generated training corpus with a derived QA set. The participant runs their DP synthesis method (**B**) on the corpus to obtain synthetic data; our standardized harness (shaded) then fine-tunes (**C**) the same checkpoint used for synthesis on this data and scores it on the test QA (**D**). Releases are regenerated quarterly (bottom loop), keeping the benchmark contamination-free.

- **Same generator and downstream model to eliminate distillation confounds.** To rule out distillation from a stronger teacher, we enforce that identical base models (e.g., GEMMA 3 4B) are used for generating the DP synthetic data as well as for measuring downstream task improvement. This ensures that any performance improvements arise strictly from the information captured from the training data.
- **An improved understanding of the gaps between existing methods.** While existing evaluations suggest that most methods perform similarly, on CONTINUOUSBENCH, we find that DP synthesis falls far short of non-private synthesis (Figure 1), even at $\varepsilon = 100$, which suggests a core open problem in DP text synthesis. We also study two state-of-the-art training-free methods, Private Evolution (PE) (Xie et al., 2024) and InvisibleInk (Vinod et al., 2026), and find that they significantly underperform DP fine-tuning. In particular, training on PE- or InvisibleInk-generated data does not improve the base model.
- **Standardized, open-source evaluation harness.** We release the data, training recipes, and scoring scripts as an easy-to-use benchmarking package: `https://anonymous.4open.science/r/ContinuousBenchEval-5EF3/`. This provides a reliable way to compare methods, and will help us accelerate progress on DP synthetic data.

2 BACKGROUND AND PROBLEM FORMULATION

DP text synthesis. Differential privacy bounds how much any individual's data can affect the output of an algorithm. The privacy parameters (ε, δ) controls the strength of the guarantee, with smaller (ε, δ) corresponding to stronger privacy.

In this work, we call any DP algorithm that releases a dataset a *DP synthesis mechanism*. A common approach for text is to fine-tune a pretrained generator with DP-SGD (Abadi et al., 2016), and then sample a synthetic corpus from the resulting model (Yue et al., 2023; Kurakin et al., 2023). Compared to standard SGD, DP-SGD clips each per-example gradient, aggregates the clipped gradients, and adds calibrated noise before each update.

See Section 5 of Ponomareva et al. (2025) for a broad overview of DP synthetic text. We discuss additional related work in Appendix A.

Table 1: DP text synthesis evaluation in existing work. **NTP** = next-token prediction; **Cls** = classification; **Sim** = distributional similarity. **Fresh** indicates whether the evaluated knowledge is constructed to be unavailable to the base model. **Matched** indicates that the generator and downstream evaluator are of comparable scale.

Study	Task	Fresh?	Matched?
Yu et al. (2023)	NTP	as of 2023[1]	No
Kurakin et al. (2023)	Cls / Sim	No[2]	No
Xie et al. (2024)	NTP / Cls / Sim	as of 2023[1]	No
Amin et al. (2024)	Cls	No	No
Tan et al. (2025)	NTP / Cls / Sim	No[2]	No
Hu et al. (2025)	NTP / Cls / Sim	No	No
Sun et al. (2025b)	Cls / Sim	No	No
CONTINUOUSBENCH	QA	Yes + Regenerable	Yes

Table 1 compares prior work on DP text synthesis evaluation, in terms of the tasks used for evaluation, whether the evaluation is performed

[1]Yu et al. (2023) collected PubMed abstracts published between August 1–7, 2023, and Xie et al. (2024) collected reviews for ICLR 2023 submissions from OpenReview.

[2]They mitigate data contamination by retraining the base model from scratch on corpora curated to exclude the downstream task data, which are not publicly available.

on fresh data relative to the base model's knowledge, and whether the generator and evaluator models were matched in scale. See Appendix A.1 for a more detailed discussion of each work.

Knowledge transfer from source corpus. We study whether DP synthetic text can preserve the *knowledge* that would have been obtained by training on the original corpus. In the ideal evaluation, one would re-run pretraining with and without the candidate dataset and measure the difference in the trained models' ability to recall knowledge, which is prohibitively expensive. We therefore use *continual pretraining* as a practical proxy: starting from a fixed pretrained checkpoint, we train on the candidate corpus and then evaluate the resulting model with a fixed set of eval QA prompts. In this setting, a dataset is useful if additional training on it improves the model's eval accuracy. CONTINUOUSBENCH instantiates the source dataset (input for synthesis), as well as the downstream training and evaluation process to facilitate the search for better DP synthesis mechanisms.

Population-level knowledge, not singleton memorization. A natural concern is that grounded QA rewards the type of memorization DP is designed to suppress. We agree that a DP benchmark should not reward recovery of facts that appear only once in a corpus. It is equally clear that no reasonable parameterization of DP should erase a model's ability to recall information attested across hundreds or thousands of independent records. For example: a model trained on DP synthetic clinical notes (Wu et al., 2025b; Hu et al., 2025), should, at minimum, be able to recall basic details about conditions described across hundreds of independent records in order to be useful. Information at this level of redundancy is population-level knowledge rather than individual-specific memorization, and a synthesis method that cannot recover it has not meaningfully preserved the corpus. A model trained on such synthetic data, that cannot recall basic facts from a domain, is unlikely to be reasoning competently over such a domain.

CONTINUOUSBENCH therefore measures utility as a function of frequency rather than collapsing it to a binary, and uses short-answer QA as a minimal, automatically verifiable proxy for corpus-specific knowledge transfer, in the spirit of factual-recall benchmarks such as MMLU (Hendrycks et al., 2021b;a) and Natural Questions (Kwiatkowski et al., 2019). We expand on this in Appendix B.

3 CONTINUOUSBENCH

Benchmark overview. CONTINUOUSBENCH is a benchmark and evaluation harness for differentially private text synthesis. Each CONTINUOUSBENCH release is a versioned evaluation package containing: (i) a training corpus, (ii) a grounded short-answer QA set derived from that corpus, and (iii) a standardized harness for downstream training, prompting, decoding, normalization, and scoring. Section 3.1 describes the concrete dataset tracks instantiated in this work.

Benchmark Usage. Given a release, the participant selects a privacy regime and a target base model checkpoint, and then applies their DP synthesis method to the release corpus to obtain a synthetic training set. The synthetic training set is handed over to the CONTINUOUSBENCH harness, which conducts continued pretraining on *same base checkpoint* on that synthetic data using our fixed recipe. The resulting model is then evaluated on the release QA set. Figure 2 illustrates this workflow. For every configuration, we require three reference runs: **No training**, which evaluates the checkpoint without corpus training; **Train on real**, which fine-tunes on the original corpus; and **Syn**, which fine-tunes on synthetic data generated without privacy constraints ($\varepsilon = \infty$).

Benchmark design principles. Each CONTINUOUSBENCH release is produced by an automated pipeline that constructs a corpus and derives a matched QA set from it. We plan to refresh and release CONTINUOUSBENCH annually. Each release is designed to satisfy three properties:

1. **Freshness and access dependency.** QA items test knowledge unavailable to the base checkpoint but recoverable from the release corpus. We enforce freshness through post-cutoff collection or procedural regeneration, and validate access dependency by checking that the base checkpoint performs near chance while training on the original corpus yields a substantial gain.
2. **Verifiable grounded QA.** Each QA item is linked to supporting corpus records and has a normalized short answer supporting verifiability.
3. **DP learnability through many independent supporting records.** DP suppresses rare knowledge present in only a single record. Hence, for each eval QA, we track the number of indepen-

dent records that contain the information needed to answer it, and evaluate on questions supported by hundreds of independent records as learnable population-level knowledge.

3.1 DATASET TRACKS

CONTINUOUSBENCH provides two complementary dataset tracks. GEMINON is a fully controlled fictional domain that enables clean analyses of DP learnability. NEWS is a continuously collected real-world corpus, derived from Common-Crawl-News (Common Crawl, 2026) that captures the messiness of natural text and time-evolving facts. Together, these tracks support both mechanism-level analysis and end-to-end evaluation.

Track I: GEMINON. GEMINON is a fully fictional, Pokémon-inspired world designed to make freshness procedural and unambiguous. Each release regenerates a new set of fictional entities together with attributes such as types, stats, and evolutions, along with a corpus describing them across multiple in-world formats. These formats are chosen to distribute each atomic fact across multiple independent records while avoiding trivial one-to-one paraphrases, enabling explicit control over the number of records containing a fact and textual diversity. From this corpus, we construct QA sets for both repeated facts and singleton facts.

Track II: NEWS. NEWS evaluates DP synthesis in a realistic, noisy setting derived from contemporary news. For each release, we collect articles from a fixed post-cutoff window, clean and segment them into timestamped records, and deduplicate near-identical content while preserving natural event-level redundancy across outlets.

Estimating the number of records supporting a question. The two tracks estimate support count differently. In GEMINON, support count is controlled by construction or counted from recorded attribute mentions in generated articles. In NEWS, support count is estimated by verification. First, we cluster articles to identify events and generate QAs for each event. For each QA, we retrieve candidate supporting articles, and designate the article as supporting only if its counterfactual inclusion in a model's context causes it to start producing the correct answer. Since the procedure only tests retrieved candidate articles, the resulting support count estimate is a conservative lower bound.

Each dataset is released in **Large**, **Medium**, and **Small** variants, with train–validation–test splits. The corresponding QA sets are released with validation and test splits. In GEMINON, each QA split contains two subsets: a *high-repetition* subset, where each target fact appears in approximately 200 records, and a *singleton* subset, where each target fact appears only once. Both subsets are derived from the same release corpus; the singleton subset is used only as a privacy sanity check and is not intended to be recoverable under DP. For NEWS, the canonical evaluation set keeps all questions with support count at least 200; we also report results on the full QA set as well as subsets with support count at least 400, 600, and 800 in Appendix D. Full curation details, qualitative examples, corpus statistics, and QA statistics are provided in Appendices H and I for the two tracks.

3.2 EVALUATION HARNESS AND METRICS

Given a candidate corpus, either real or (DP) synthetic, CONTINUOUSBENCH trains a *fixed evaluator* initialized from the same base checkpoint as used for generation, using a prescribed continual-pretraining recipe. The resulting model is then evaluated on the target QA set. The harness fixes the downstream training recipe, checkpoint selection strategy, few-shot prompts for evaluation, sampling, and answer scoring. Thus, performance differences are intended to reflect the utility of the candidate corpus rather than downstream evaluation choices. Fixed downstream training enables cleaner, more compute-efficient, and reproducible evaluation. It also reflects the reality of data contributions to model training: pretraining is performed on a mixture of datasets and hyperparameters cannot be specifically tuned for each individual component.

Unless otherwise specified, final results are reported on the held-out QA test split using the track-specific primary metric: exact-match accuracy for GEMINON, whose answers have canonical surface forms, and LLM-match accuracy for NEWS, whose answers may admit paraphrases or other free-form variation. The evaluator checkpoint is selected based on the best "contains" accuracy on the QA validation split, where a prediction is scored as correct when the normalized ground-truth answer is contained in the normalized model prediction. All QA metrics use the same deterministic response parsing and normalization before scoring. Additional details are presented in Appendix C.

We release the full evaluation harness, including training recipes, prompts, sampling configurations, normalization rules, and scoring scripts at `https://anonymous.4open.science/r/ContinuousBenchEval-5EF3/`.

4 EXPERIMENTAL RESULTS

We organize our experiments around three questions: **(i)** Does the benchmark satisfy its design desiderata, namely freshness and DP learnability? **(ii)** How large is the capability gap between non-private and DP synthesis? **(iii)** How does the number of supporting records affect knowledge transferability?

Datasets and training details. We evaluate on GEMINON-SMALL and NEWS-SMALL, each containing approximately 200K training examples. We use pretrained checkpoints from the GEMMA 3 family for both the generator and the evaluator. For synthesis, we train both public and DP generators on the training split of the original corpus using LoRA with rank 128. For DP generator training, we employ state-of-the-art, optimized DP-SGD configurations employing truncated poisson sampling (Chua et al., 2024), PLD accounting (Ganesh, 2025) with $\delta = |\texttt{dataset}|^{-1.1}$, and normalized clipping (De et al., 2022) with aggressive clipping norm `1e-3`. We follow recommendations from existing literature regarding the necessity of larger batch sizes and compute budgets than standard training (Ghazi et al., 2022; Li et al., 2022; De et al., 2022; Sander et al., 2023). Specifically our DP runs use 16x larger batch sizes and $\approx$ 5.3x more accelerator hours and FLOPs than standard training. We use the highly-performant JAX Privacy (McKenna et al., 2026) library for training.

For downstream evaluation, we train evaluator models on one of three data sources: the original corpus, a non-private synthetic corpus **Syn** ($\varepsilon = \infty$), or a DP synthetic corpus **DP-Syn** with $\varepsilon \in \{100, 10\}$. We conduct both full-parameter and LoRA evaluator training. All generators and evaluators are trained with the standard causal language modeling objective in continual-pretraining style. Unless otherwise stated, we report exact-match accuracy for GEMINON and LLM-match accuracy for NEWS using full-parameter trained evaluators. By default, QA results are evaluated on GEMINON high-repetition QA test split, whose facts have approximately 200 repetitions, and the NEWS QA test subset with repetition at least 200. Full details on evaluator scoring, evaluator training, and generator training can be found in Appendix C.

4.1 VALIDATION OF BENCHMARK

We first validate that the benchmark satisfies two basic desiderata: (i) the evaluated facts are not already accessible to the base models, and (ii) the facts are learnable from the released corpus under our standardized training recipes.

As shown in Table 2, the base checkpoints achieve near-chance QA accuracy without access to the corpus, especially on GEMINON. This indicates that the evaluated knowledge is not readily available from previous training stages. In contrast, training on the real corpus yields large gains for both datasets, confirming that the target knowledge is present in the corpus and can be acquired through training. These results validate the benchmark as a testbed for measuring whether synthetic data preserves learnable, corpus-specific knowledge.

Table 2: Models obtain high CONTINUOUS-BENCH QA accuracy (%) if and only if they train on the corpus.

	GEMINON (%)		NEWS (%)	
Model	**No training**	**Train on real**	**No training**	**Train on real**
1B	1.0	88.2	9.6	51.3
4B	1.1	96.4	14.1	70.4

NEWS results stratified by support count can be found in Table 11, Appendix D. Accuracy increases with support count and larger evaluator models learn repeated facts more reliably. Full exact-match, contains, and LLM-match results for all validation configurations are in Table 10.

4.2 THE CAPABILITY GAP UNDER DIFFERENTIAL PRIVACY

We next address the second question: how much utility is lost when moving from non-private synthesis to DP synthesis? Figure 3 shows a large capability gap, and that DP synthetic data fails to transfer corpus knowledge. Non-private synthesis (**Syn**, $\varepsilon = \infty$) transfers substantial factual knowl-

edge from the original corpus: with matched 4B generator and 4B evaluator, downstream accuracy reaches 92.5% on GEMINON and 65.54% on NEWS. The corresponding 1B setting also achieves strong performance, reaching 86.31% on GEMINON and 53.04% on NEWS. Thus, synthetic data can serve as an effective medium for transferring corpus-specific knowledge when generated without privacy constraints.

However, when the synthetic corpus is generated by a DP fine-tuned generator (**DP-Syn**), it sharply reduces this transfer. At $\varepsilon = 100$, accuracy drops to 3.78% on GEMINON and 25.89% on NEWS for the 4B setting, and to 6.85% and 16.79% for the 1B setting. At $\varepsilon = 10$, performance falls further, reaching only 2.92% on GEMINON and 19.64% on NEWS for the 4B setting.

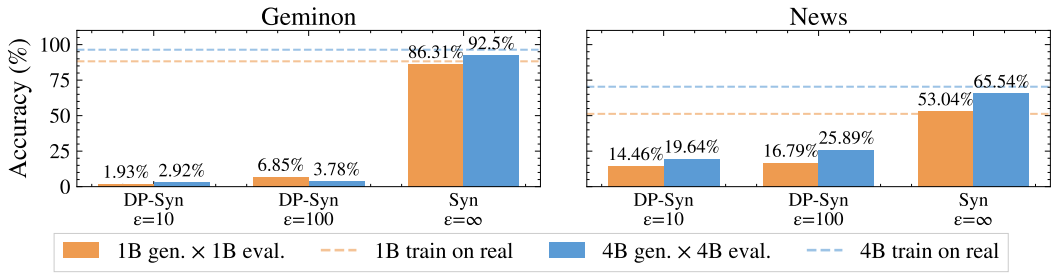

Figure 3: Capability gap between non-private and DP synthesis, measured by QA accuracy. Bars show evaluators trained on DP synthetic data (**DP-Syn**) at $\varepsilon \in \{10, 100\}$ and on non-private synthetic data (**Syn**, $\varepsilon = \infty$), for matched generator-evaluator sizes (1B×1B in orange, 4B×4B in blue). Dashed horizontal lines mark the train-on-real-corpus reference.

This capability gap is consistent across downstream evaluator training regimes (LoRA vs. full fine-tuning), generator–evaluator size configs ({1B, 4B} x {1B, 4B}), and datasets. This suggests that the main bottleneck lies in the DP-constrained synthetic corpus, more specifically, in the DP fine-tuned synthesizer. Metrics for all training and generator–evaluator configs are in Tables 12 and 13.

We further examine this distinction using MAUVE (Pillutla et al., 2021), which measures distributional similarity between generated and original corpora. We use Gecko 110M embedding models (Lee et al., 2024) with 1K examples and report mean scores and standard deviation over 5 runs. Figure 4 shows that MAUVE decreases much more mildly than QA accuracy under DP synthesis. On GEMINON, for example, the 4B generator achieves MAUVE 0.83 without privacy and 0.73 at $\varepsilon = 10$, even though QA accuracy drops from 92.5% to 2.92%. The same pattern appears on NEWS. These results indicate that DP generators can preserve the *distributional form* of the corpus—style, topic distribution, and document structure—while failing to preserve the *specific factual content* required for grounded QA. Additional results are in Table 14.

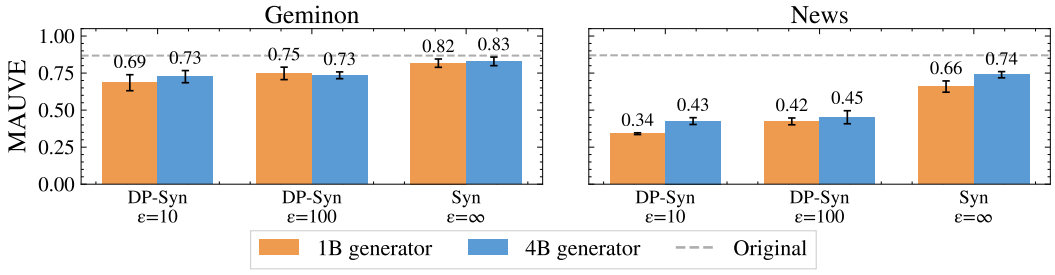

Figure 4: MAUVE scores for synthetic corpora, from either a 1B or 4B generator, against original corpora. The dashed gray line is the MAUVE score between disjoint splits of the original corpora.

Together, these results show that distributional metrics alone are insufficient for evaluating DP text synthesis, for certain type of tasks. A method can obtain moderate MAUVE while transmitting little of the corpus-specific knowledge that makes the private data useful.

We also report direct DP fine-tuning results on the original corpus in Appendix E and find that the performance gap persists. To understand which component of DP-SGD contributes more to this gap, we further ablate its two components, gradient clipping and noise addition, respectively, and find that noising accounts for almost all of the utility degradation (Appendix E.1).

4.3 TRAINING-FREE DP SYNTHESIS METHODS FAIL TO TRANSFER CORPUS KNOWLEDGE

We also evaluate the performance of two training-free DP synthesis methods, Private Evolution (Xie et al., 2024) and InvisibleInk (Vinod et al., 2026), on CONTINUOUSBENCH. PE is a training-free DP synthesis method that has achieved strong results on existing benchmarks, given the API-access to a model, and has been the subject of substantial line of follow-up work (Zhang et al., 2025; González et al., 2026; Hou et al., 2024). In contrast, InvisibleInk assumes access to the model's logits and privatizes generation at inference time, casting next token sampling as an exponential mechanism over logits clipped relative to a public reference.

Private Evolution (PE). We follow the recipe of (Xie et al., 2024), using 10 PE iterations with 7 paraphrases per example at $\varepsilon = 10$. We use GEMMA 3 1B PT as the generator, and in Appendix D.3, we also evaluate GEMMA 3 1B IT to test whether stronger instruction following improves PE performance, but find that it does not. For each configuration, we choose the sampling temperature with the lowest FID[2] and generate a 20K-example PE corpus.[3] We then train downstream 1B evaluators on the resulting PE corpora with the same standardized evaluation pipeline.

Table 3: Private Evolution results at $\varepsilon = 10$. We report "contains" QA accuracy.

	GEMINON		NEWS	
Method	**FID**	**QA (%)**	**FID**	**QA (%)**
1B No training	-	1.9	-	7.0
1B PE ($\varepsilon = 10$)	33.5	1.3	27.4	6.8

Table 3 shows that evaluators trained on PE-generated data perform worse than an untrained baseline. Although PE produces corpora with reasonable FID scores, e.g., 27.4 for NEWS and 33.5 for GEMINON, downstream QA accuracy remains near zero (0.2%) on GEMINON and does not exceed the base checkpoint on NEWS.[4] As noted by Xie et al. (2024), PE relies on knowledge already present in the language model to generate high-quality candidate texts without domain-specific fine-tuning. In our setting, however, the target facts are fresh: the base model does not already know the relevant GEMINON entities and attributes or post-cutoff NEWS events. DP selection can therefore choose candidates that are stylistically close to the private corpus, but it cannot reliably recover facts that are absent from the generator's proposal distribution.

InvisibleInk. We follow the recipe of Vinod et al. (2026) at $\varepsilon = 10$ with GEMMA 3 1B IT as the generator, conditioning each generated record on 7 private references and using each original record as a reference exactly once, which yields corpora each of size $\approx$ 20K. We then train downstream 1B evaluators on them with the same standardized evaluation pipeline. Hyperparameters, full results, and further discussion are deferred to Appendix D.4.

Table 4: InvisibleInk results at $\varepsilon = 10$. We report "contains" QA accuracy.

	GEMINON		NEWS	
Method	**Val loss**	**QA (%)**	**Val loss**	**QA (%)**
1B No training	3.3	1.9	2.3	7.0
1B InvisibleInk ($\varepsilon = 10$)	16.0	0.2	10.8	2.1

Table 4 shows that evaluators trained on InvisibleInk-generated data perform worse than the base model on both tracks, and the failure resembles PE. Cross-entropy on the real validation corpus rises well above the base model while training loss decreases, and QA accuracy is below the base model at the first checkpoint and decays as training proceeds. The generations are fluent and match the corpus format, but entity names, stats, and events are mostly hallucinated.

[2]FID is the main distributional similarity metric reported by PE (Xie et al., 2024); lower values indicate that generated samples are closer to the target corpus in the embedding space.

[3]Due to computational reasons, since there is a 7×10 sampling overhead for each additional example. Although GEMINON-SMALL and NEWS-SMALL each contain 200K training examples, this 20K-example scale is comparable to prior Aug-PE evaluations: Xie et al. (Xie et al., 2024) generated at most 100K examples for Yelp (Yelp Inc., 2024), whose source corpus contains 1.7M examples, and 5K examples for PubMed (Yu et al., 2023), whose source corpus contains 75K examples.

[4]We also observe that, when training evaluators on PE-generated GEMINON data, cross-entropy on the original GEMINON validation corpus consistently increases while training loss decreases. On NEWS, accuracy decreases over training; see Figure 6. This suggests that the failure is not primarily explained by the smaller number of synthetic examples.

4.4 THE NUMBER OF SUPPORTING RECORDS AFFECTS KNOWLEDGE TRANSFERABILITY

A crucial feature of CONTINUOUSBENCH is that each test QA is paired with its *support count* – the number of independent training records that contain the information required to answer it correctly. We use this to ensure that our tasks are solvable under DP; but furthermore, it allows us to investigate the performance profile of a given synthesis method on questions of varying support counts.

For GEMINON, we evaluate on the singleton set, which consists of questions whose answers are, by design, present in only a single corpus record. Hence, DP should strongly suppress their learnability. Consistent with this expectation, downstream evaluators perform poorly on the sensitive split, as shown in Table 5. Full results on GEMINON singleton set are reported in Table 22.

Table 5: Singleton set accuracy (%) on GEMINON for facts that appear in only one record in the corpus.

Model	No Training	Train on real	**Syn**	**DP-Syn**@$\varepsilon = 100$	**DP-Syn**@$\varepsilon = 10$
1B	0.6	3.0	1.2	0.8	0.6
4B	0.9	5.0	2.1	0.8	0.8

For NEWS, the number of records supporting a question varies naturally across our evaluation set. Figure 5 shows the effect of support count on accuracy, via bucketing questions by support count and plotting the average accuracy inside the bucket. We see that the gap between DP synthetic and "No training" emerges around $k \approx 600$. Full results for NEWS accuracy by support count are reported in Tables 11, 20, 21 in Appendix D. A more detailed empirical analysis of the learning dynamics, provided in Appendix F, further supports this conclusion.

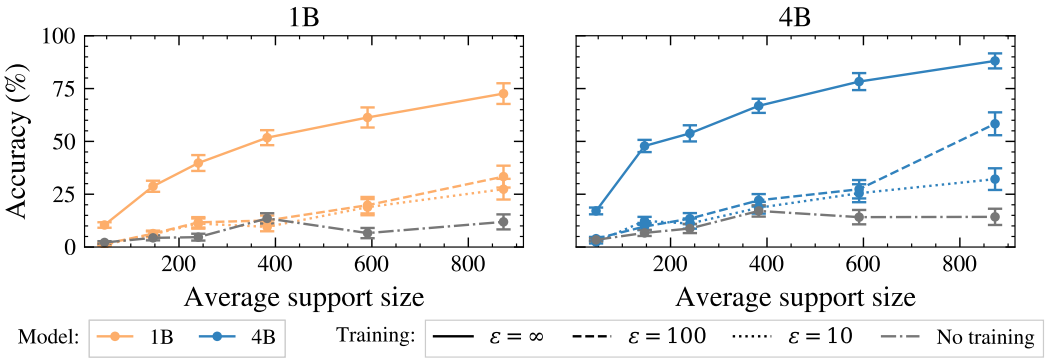

Figure 5: Accuracy (%) on NEWS as a function of QA support count. QAs are bucketed by support count; we plot the average support count and accuracy inside each bucket. Error bars show standard deviation.

5 CONCLUSION

We introduce CONTINUOUSBENCH, a continuously regenerated benchmark and evaluation harness that measures whether DP synthetic text preserves the capability gains that direct corpus access would have provided. By coupling a fresh training corpus with grounded QA and enforcing DP learnability, CONTINUOUSBENCH overcomes the saturation problems that plague prior evaluations. Our experiments reveal a stark gap: non-private synthesis transfers substantial corpus-specific knowledge, while state-of-the-art DP methods fail to do so even at $\varepsilon = 100$, despite producing superficially similar corpora. We view closing this gap as the central open problem in DP text synthesis—a method that cannot recover knowledge attested across hundreds of independent records has not meaningfully preserved a corpus. We hope CONTINUOUSBENCH serves as a reproducible yardstick for measuring and accelerating progress in this direction.

AI USE STATEMENT

In this work, we used generative AI tools for generating synthetic data and implementing methods. Specifically, to generate the GEMINON corpus (including entity names and moves), to generate the QA pairs for the NEWS track, to serve as an automated judge in our evaluation suite, and to help implement the data curation pipeline and evaluation harness. We have not used generative AI tools

for idea formation, research methodology or experimental design, the entity statistics and QA pairs of GEMINON, or the NEWS corpus (which is crawled from public news sources), and theory-related tasks are not applicable to this work. Additionally, we used generative AI tools to help design figures and to improve the readability of the paper. We have reviewed all AI-assisted work, e.g., the paper was written by the authors, the code was checked by the suthors, and all plots were produced by the authors directly from experimental results, which we verified. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We open-source the full pipeline for constructing CONTINUOUSBENCH, including the corpus generation, the QA derivation procedure, and the standardized training and evaluation harness, as well as the datasets, at `https://hf.co/AnonymousContinuousBench` for both tracks.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pp. 308–318, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450341394. doi: 10.1145/2976749.2978318. URL https://doi.org/10.1145/2976749.2978318.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. *arXiv preprint arXiv:2309.14316*, 2023.

Thiago Amancio. Full pokemons and moves datasets. https://www.kaggle.com/datasets/thiagoamancio/full-pokemons-and-moves-datasets, 2024.

Kareem Amin, Alex Bie, Weiwei Kong, Alexey Kurakin, Natalia Ponomareva, Umar Syed, Andreas Terzis, and Sergei Vassilvitskii. Private prediction for large-scale synthetic text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7244–7262, Miami, Florida, USA, 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-emnlp.425.

Simran Arora and Christopher Ré. Can foundation models help us achieve perfect secrecy?, 2022. URL https://arxiv.org/abs/2205.13722.

Amin Banayeeanzade, Qingchuan Yang, Deqing Fu, Spencer Hong, Erin Babinsky, Alfy Samuel, Anoop Kumar, Robin Jia, and Sai Praneeth Karimireddy. EPSVec: Efficient and private synthetic data generation via dataset vectors. *arXiv preprint arXiv:2602.21218*, 2026.

Kai Chen, Xiaochen Li, Chen Gong, Ryan Mckenna, and Tianhao Wang. Benchmarking differentially private tabular data synthesis: [experiments & analysis]. *Proc. ACM Manag. Data*, 3(6), December 2025. doi: 10.1145/3769764. URL https://doi.org/10.1145/3769764.

Daixuan Cheng, Yuxian Gu, Shaohan Huang, Junyu Bi, Minlie Huang, and Furu Wei. Instruction pre-training: Language models are supervised multitask learners. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 2529–2550, 2024.

Lynn Chua, Badih Ghazi, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, and Chiyuan Zhang. Scalable DP-SGD: Shuffling vs. poisson subsampling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=6gMnj9oc6d.

Common Crawl. CC-NEWS Dataset. https://data.commoncrawl.org/crawl-data/CC-NEWS/index.html, 2026. Accessed: 2026-05-06.

Soham De, Leonard Berrada, Jamie Hayes, Samuel L Smith, and Borja Balle. Unlocking high-accuracy differentially private image classification through scale. *arXiv preprint arXiv:2204.13650*, 2022.

James Flemings, Meisam Razaviyayn, and Murali Annavaram. Differentially private next-token prediction of large language models, 2024. URL https://arxiv.org/abs/2403.15638.

Arun Ganesh. Tighter privacy analysis for truncated poisson sampling. *arXiv preprint arXiv:2508.15089*, 2025.

Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. URL https://arxiv.org/abs/2507.06261.

Badih Ghazi, Pasin Manurangsi, Ravi Kumar, Rohan Anil, and Vineet Gupta. Large-scale differentially private bert. In *Findings of EMNLP 2022*, 2022.

Chen Gong, Kecen Li, Zinan Lin, and Tianhao Wang. DPImageBench: A unified benchmark for differentially private image synthesis, 2025. URL https://arxiv.org/abs/2503.14681.

Tomás González, Giulia Fanti, and Aaditya Ramdas. Private evolution converges. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=zOCENGh1Jg.

Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. Aligning ai with shared human values. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021a.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021b.

Charlie Hou, Akshat Shrivastava, Hongyuan Zhan, Rylan Conway, Trang Le, Adithya Sagar, Giulia Fanti, and Daniel Lazar. PrE-Text: training language models on private federated data in the age of llms. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24, 2024.

Yuzheng Hu, Ryan McKenna, Da Yu, Shanshan Wu, Han Zhao, Zheng Xu, and Peter Kairouz. ACTG-ARL: Differentially private conditional text generation with rl-boosted control, 2025. URL https://arxiv.org/abs/2510.18232.

Alexey Kurakin, Natalia Ponomareva, Umar Syed, Liam MacDermed, and Andreas Terzis. Harnessing large-language models to generate private synthetic text, 2023. URL https://arxiv.org/abs/2306.01684.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Jinhyuk Lee, Zhuyun Dai, Xiaoqi Ren, Blair Chen, Daniel Cer, Jeremy R. Cole, Kai Hui, Michael Boratko, Rajvi Kapadia, Wen Ding, Yi Luan, Sai Meher Karthik Duddu, Gustavo Hernandez Abrego, Weiqiang Shi, Nithi Gupta, Aditya Kusupati, Prateek Jain, Siddhartha Reddy Jonnalagadda, Ming-Wei Chang, and Iftekhar Naim. Gecko: Versatile text embeddings distilled from large language models, 2024. URL https://arxiv.org/abs/2403.20327.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, Kushal Arora, et al. DataComp-LM: In search of the next generation of training sets for language models. *Advances in Neural Information Processing Systems*, 37: 14200–14282, 2024.

Xuechen Li, Florian Tramer, Percy Liang, and Tatsunori Hashimoto. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=bVuP3ltATMz.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. WildBench: Benchmarking llms with challenging tasks from real users in the wild. *arXiv preprint arXiv:2406.04770*, 2024.

Adam Liska, Tomas Kocisky, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, Cyprien De Masson D'Autume, Tim Scholtes, Manzil Zaheer, Susannah Young, Ellen Gilsenan-Mcmahon, Sophia Austin, Phil Blunsom, and Angeliki Lazaridou. StreamingQA: A benchmark for adaptation to new knowledge over time in question answering models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13604–13622. PMLR, 2022. URL https://proceedings.mlr.press/v162/liska22a.html.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P11-1015.

maca11. All pokemon dataset. https://www.kaggle.com/datasets/maca11/all-pokemon-dataset, 2021.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *Conference on Language Modeling (COLM)*, 2024. URL https://openreview.net/forum?id=B41hNBoWLo.

Ryan Mckenna, Yangsibo Huang, Amer Sinha, Borja Balle, Zachary Charles, Christopher A. Choquette-Choo, Badih Ghazi, Georgios Kaissis, Ravi Kumar, Ruibo Liu, Da Yu, and Chiyuan Zhang. Scaling laws for differentially private language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 43375–43398. PMLR, 2025. URL https://proceedings.mlr.press/v267/mckenna25a.html.

Ryan McKenna, Galen Andrew, Borja Balle, Vadym Doroshenko, Arun Ganesh, Weiwei Kong, Alex Kurakin, Brendan McMahan, and Mikhail Pravilov. JAX-Privacy: A library for differentially private machine learning. *arXiv preprint arXiv:2602.17861*, 2026.

Oded Ovadia, Meni Brief, Rachel Lemberg, and Eitam Sheetrit. Knowledge-instruct: Effective continual pre-training from limited data using instructions. *arXiv preprint arXiv:2504.05571*, 2025.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. Mauve: Measuring the gap between neural text and human text using divergence frontiers, 2021. URL https://arxiv.org/abs/2102.01454.

Natalia Ponomareva, Hussein Hazimeh, Alex Kurakin, Zheng Xu, Carson Denison, H. Brendan McMahan, Sergei Vassilvitskii, Steve Chien, and Abhradeep Thakurta. How to DP-fy ML: A practical guide to machine learning with differential privacy. *Journal of Artificial Intelligence Research*, 77:1113–1201, 2023. doi: 10.1613/jair.1.14649. URL https://doi.org/10.1613/jair.1.14649.

Natalia Ponomareva, Zheng Xu, H. Brendan McMahan, Peter Kairouz, Lucas Rosenblatt, Vincent Cohen-Addad, Cristóbal Guzmán, Ryan McKenna, Galen Andrew, Alex Bie, Da Yu, Alex Kurakin, Morteza Zadimoghaddam, Sergei Vassilvitskii, and Andreas Terzis. How to DP-fy Your Data: A practical guide to generating synthetic data with differential privacy, 2025. URL https://arxiv.org/abs/2512.03238.

Krithika Ramesh, Nupoor Gandhi, Pulkit Madaan, Lisa Bauer, Charith Peris, and Anjalie Field. Evaluating differentially private synthetic data generation in high-stakes domains. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15254–15269, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.894. URL https://aclanthology.org/2024.findings-emnlp.894.

Anka Reuel, Amelia Hardy, Chandler Smith, Max Lamparth, Malcolm Hardy, and Mykel J Kochenderfer. BetterBench: Assessing ai benchmarks, uncovering issues, and establishing best practices. *Advances in Neural Information Processing Systems*, 37:21763–21813, 2024.

Tom Sander, Pierre Stock, and Alexandre Sablayrolles. TAN without a burn: Scaling laws of DP-SGD. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 29937–29949. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/sander23b.html.

Yidan Sun, Viktor Schlegel, Srinivasan Nandakumar, Iqra Zahid, Yuping Wu, Warren Del-Pinto, Goran Nenadic, Siew-Kei Lam, Jie Zhang, and Anil A. Bharath. Evaluating differentially private generation of domain-specific text. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM '25)*, pp. 5273–5278, New York, NY, USA, 2025a. Association for Computing Machinery. doi: 10.1145/3627673.3680074. URL https://dl.acm.org/doi/10.1145/3627673.3680074.

Yidan Sun, Viktor Schlegel, Srinivasan Nandakumar, Iqra Zahid, Yuping Wu, Yulong Wu, Hao Li, Jie Zhang, Warren Del-Pinto, Goran Nenadic, et al. SynBench: A benchmark for differentially private text generation. *arXiv preprint arXiv:2509.14594*, 2025b.

Bowen Tan, Zheng Xu, Eric Xing, Zhiting Hu, and Shanshan Wu. Synthesizing privacy-preserving text data via finetuning without finetuning billion-scale llms. In *Proceedings of the 42nd International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2025. URL https://arxiv.org/abs/2503.12347.

EmbeddingGemma Team. EmbeddingGemma: Powerful and lightweight text representations, 2025. URL https://arxiv.org/abs/2509.20354.

Rushil Thareja, Preslav Nakov, Praneeth Vepakomma, and Nils Lukas. Dp-fusion: Token-level differentially private inference for large language models, 2026. URL https://arxiv.org/abs/2507.04531.

Florian Tramèr, Gautam Kamath, and Nicholas Carlini. Position: Considerations for differentially private learning with large-scale public pretraining. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 48453–48467. PMLR, 2024. URL https://proceedings.mlr.press/v235/tramer24a.html.

Saiteja Utpala, Sara Hooker, and Pin-Yu Chen. Locally differentially private document generation using zero shot prompting. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 8442–8457, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.566. URL https://aclanthology.org/2023.findings-emnlp.566/.

Vishnu Vinod, Krishna Pillutla, and Abhradeep Guha Thakurta. Invisibleink: High-utility and low-cost text generation with differential privacy, 2026. URL https://arxiv.org/abs/2507.02974.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. FreshLLMs: Refreshing large language models with search engine augmentation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13697–13720, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.813. URL https://aclanthology.org/2024.findings-acl.813.

Shuaiqi Wang, Vikas Raunak, Arturs Backurs, Victor Reis, Pei Zhou, Sihao Chen, Longqi Yang, Zinan Lin, Sergey Yekhanin, and Giulia Fanti. Struct-Bench: A benchmark for differentially private structured text generation. In *Advances in Neural Information Processing Systems (NeurIPS) 2025 Datasets and Benchmarks Track*, 2025. URL https://openreview.net/forum?id=59vXWteYuh.

Xiaobao Wu, Liangming Pan, Yuxi Xie, Ruiwen Zhou, Shuai Zhao, Yubo Ma, Mingzhe Du, Rui Mao, Anh Tuan Luu, and William Yang Wang. AntiLeakBench: Preventing data contamination by automatically constructing benchmarks with updated real-world knowledge. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025a. URL https://arxiv.org/abs/2412.13670.

Yuping Wu, Viktor Schlegel, Warren Del-Pinto, Srinivasan Nandakumar, Iqra Zahid, Yidan Sun, Usama Farghaly Omar, Amirah Jasmine, Arun-Kumar Kaliya-Perumal, Chun Shen Tham, et al. Term2Note: Synthesising differentially private clinical notes from medical terms. *arXiv preprint arXiv:2509.10882*, 2025b.

Chulin Xie, Zinan Lin, Arturs Backurs, Sivakanth Gopi, Da Yu, Huseyin A Inan, Harsha Nori, Haotian Jiang, Huishuai Zhang, Yin Tat Lee, Bo Li, and Sergey Yekhanin. Differentially private synthetic data via foundation model APIs 2: Text. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54531–54560. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/xie24g.html.

Zitong Yang, Neil Band, Shuangping Li, Emmanuel Candes, and Tatsunori Hashimoto. Synthetic continued pretraining. *arXiv preprint arXiv:2409.07431*, 2024.

Yelp Inc. Yelp open dataset. https://business.yelp.com/data/resources/open-dataset/, 2024. Accessed: 2026-05-06.

Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, Sergey Yekhanin, and Huishuai Zhang. Differentially private fine-tuning of language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=Q42f0dfjECO.

Da Yu, Arturs Backurs, Sivakanth Gopi, Huseyin Inan, Janardhan Kulkarni, Zinan Lin, Chulin Xie, Huishuai Zhang, and Wanrong Zhang. Training private and efficient language models with synthetic data from llms. In *NeurIPS 2023 Workshop on Socially Responsible Language Modelling Research (SoLaR)*, 2023. URL https://openreview.net/forum?id=FKwtKzglFb.

Xiang Yue, Huseyin Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. Synthetic text generation with differential privacy: A simple and practical recipe. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1321–1342, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.74. URL https://aclanthology.org/2023.acl-long.74/.

Jianqing Zhang, Yang Liu, JIE FU, Yang Hua, Tianyuan Zou, Jian Cao, and Qiang Yang. PCEvolve: Private contrastive evolution for synthetic dataset generation via few-shot private data and generative APIs. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=IKCfxWtTsu.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level Convolutional Networks for Text Classification . *arXiv:1509.01626 [cs]*, September 2015.

Eric Zhao, Pranjal Awasthi, and Nika Haghtalab. From style to facts: Mapping the boundaries of knowledge injection with finetuning. *arXiv preprint arXiv:2503.05919*, 2025.

Tianyuan Zou, Yang Liu, Peng Li, Yufei Xiong, Jianqing Zhang, Jingjing Liu, Xiaozhou Ye, Ye Ouyang, and Ya-Qin Zhang. Contrastive private data synthesis via weighted multi-PLM fusion. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=oRdfFS7xO5.

Table of Contents for Appendix

A Related work **17**

A.1 Detailed Comparison with Prior DP Synthesis Evaluations 17

B Memorization, capability, and repetition. **19**

C Training recipes and evaluation details **20**

C.1 Evaluator training details 20

C.2 Generator training details 20

C.3 QA Evaluation and Sampling Recipes 21

D Experimental Details **23**

D.1 More results on validation of benchmark 23

D.2 Additional results for the DP synthesis capability gap 25

D.3 Private evolution details 27

D.3.1 Private Evolution setup and parameters 27

D.3.2 PE prompts 27

D.3.3 Downstream evaluation results 28

D.3.4 Qualitative Analysis 29

D.4 InvisibleInk details 32

D.4.1 InvisibleInk setup and parameters 32

D.4.2 Downstream evaluation results 32

D.4.3 Qualitative Analysis 33

D.5 Effect of knowledge support counts 34

E Direct DP fine-tuning **35**

E.1 What of DP-SGD causes the capability gap: Clipping vs. Noising 37

F Learning Dynamics by Attribute, Entity, and Support **39**

F.1 GEMINON learning dynamics 39

F.2 NEWS learning dynamics 39

G Saturation Case Studies on Standard DP Benchmarks **41**

G.1 IMDB Sentiment Classification 41

G.2 OpenReview Score Prediction 42

G.3 Yelp Polarity 44

H GEMINON: Curation and Evaluation Details **44**

H.1 Index Data Curation 44

H.1.1 Evolution Line and Types 44

H.1.2 Statistics 45
H.1.3 Moves and Abilities 46
H.1.4 Indexing and Schema 46
H.2 Corpus Curation 47
H.3 QA Curation. 51
H.4 Subsampling 51

I NEWS: Curation and Evaluation Details **52**
I.1 Corpus Curation 52
I.2 QA Curation 52
I.3 Subsampling 54

A Related work

DP synthetic text. A growing literature studies DP text synthesis, including API-only private evolution (Xie et al., 2024), private prediction and inference-time methods (Amin et al., 2024), clustering-guided fine-tuning (Tan et al., 2025), and RL-based controllable generation (Hu et al., 2025), among others (Yu et al., 2023; Ramesh et al., 2024; Kurakin et al., 2023; Ponomareva et al., 2025). Across these works, evaluation typically relies on classification accuracy or next-token prediction on a small set of standard benchmarks, often with a mismatch between the generator and the downstream evaluator. We identify three recurring issues that limit existing evaluations: (i) evaluated knowledge may already reside in the base model from pretraining, (ii) the downstream tasks are too simple to measure corpus-specific capability transfer, and (iii) teacher-student gaps make it difficult to attribute gains to the private corpus rather than to distillation. ContinuousBench is designed to address all three; we provide a detailed positioning relative to each prior work in Appendix A.1.

Text synthesis under other privacy models. A separate line of work studies DP text synthesis under guarantees that differ from the standard central record-level DP and are therefore not directly comparable to the privacy guarantees used in this work. DP-Prompt (Utpala et al., 2023) paraphrases each document by zero-shot prompting with clipped logits and temperature sampling, and provides a *local* DP guarantee at the document level, where neighboring inputs are any two documents. PMixED (Flemings et al., 2024) partitions the private corpus into N shards, fine-tunes one model per shard, projects each model's next-token distribution towards a public model, and samples from the average. Its adjacency relationship is defined by add-remove *an entire shard*, a coarser unit than the record-level adjacency we use, and, as a private-prediction protocol, its privacy loss composes over every predicted token. DP-Fusion (Thareja et al., 2026) uses the similar ensemble-style mixing to a different threat model, where itt paraphrases a single document while protecting a designated group of sensitive tokens (e.g., PII) in the context, providing special-token-level DP for that document.

DP synthesis benchmarks. A small number of benchmarks target the evaluation of (DP) synthetic data generation. Struct-Bench (Wang et al., 2025) and Chen et al. (2025) focus on structured and tabular data; DPImageBench (Gong et al., 2025) standardizes evaluation for DP synthetic images by fixing downstream tasks and comparing generation methods. In the text domain, Sun et al. (2025a) benchmark DP text generation using similarity metrics and downstream classification utility. These efforts are valuable, but none addresses the three evaluation challenges motivating our work: ensuring freshness of the evaluated knowledge, using tasks that measure *capability transfer* rather than surface similarity, and controlling for teacher-student mismatch.

Contamination-resistant and data-centric benchmarks. AntiLeakBench (Wu et al., 2025a) and FreshQA (Vu et al., 2024) construct continually updated QA sets to reduce contamination, but they are not packaged as matched train-test releases for measuring learning from a released corpus. StreamingQA (Liska et al., 2022) provides a timestamped corpus and QA set, but its news stream largely predates modern LLM training cutoffs. TOFU (Maini et al., 2024) is synthetically constructed and conceptually adjacent to our setting, but lacks the knowledge repetition needed to distinguish DP-learnable population facts from singleton facts and is not continuously regenerated. None of these benchmarks were designed to evaluate (DP) synthesis methods. Our benchmark design draws on data-centric evaluation principles. DataComp-LM argues that when the object of study is the data itself, downstream training, compute, and evaluation should be held fixed so that performance differences are attributable to the data (Li et al., 2024). WildBench emphasizes explicit benchmark scope and dataset criteria (Lin et al., 2024), while BetterBench distills benchmark construction into concrete best practices (Reuel et al., 2024). ContinuousBench adopts this philosophy for DP synthetic text, combining fresh corpus-specific capability transfer, grounded QA, and frontier-matched evaluation in a single continuously regenerated package.

A.1 Detailed Comparison with Prior DP Synthesis Evaluations

Below we provide a detailed comparison of the evaluation protocols used in prior DP text synthesis work, motivating the design choices of ContinuousBench. Table 1 summarizes the key dimensions.

Yu et al. (2023). This work constructed a post-cutoff PubMed dataset by collecting abstracts from the National Library of Medicine published between 2023/08/01 and 2023/08/07, after the cutoff date of Llama2-7B. They trained Llama2-7B to generate synthetic data, and they used small transformers of sizes 4.4M, 11.2M, 28.8M, and 41.4M, as the downstream evaluator, and reported results on PubMed and MediaSum using next-token prediction (NTP) accuracy. This design is appealing in that it matches the generator and evaluator, avoiding teacher-student mismatch, and ensures the data is not contaminated. However, the downstream task (NTP) remains relatively simple and does not test whether *specific facts* are transferred. Additionally, the freshness guarantee is tied to a single model family: a corpus that is post-cutoff for Llama2-7B is not necessarily fresh for newer families.

Kurakin et al. (2023). This work pretrains a LaMDA 8B variant on The Pile to ensure the generator is not itself trained on data later treated as private. They evaluate on IMDB, AGNews, and Yelp using BERT-based downstream models, primarily reporting classification accuracy together with distributional similarity metrics (MAUVE). This setup carefully addresses one contamination concern for the generator, but it still relies on classification tasks and introduces a substantial teacher-student gap between the 8B generator and 110M evaluators. Notably, Appendix D of their work reports overlap between IMDB and WebText-like corpora, illustrating how benchmark contamination can arise even when care is taken in generator pretraining.

Xie et al. (2024). This work collects review of ICLR 2023 from OpenReview and use it along with Yelp and Pubmed to evaluate a wide range of generator families via in-context learning, including GPT-2 variants, GPT-3.5, OPT-6.7B, Vicuna-7B, Llama2-7B-chat, Falcon-7B-instruct, and Mixtral-8x7B. The downstream evaluators are comparatively small: RoBERTa for Yelp and OpenReview; BERT for PubMed. The generator-evaluator mismatch is substantial across all configurations.

Tan et al. (2025). This work uses a BART-base generator, which is continually pretrained on a curated contamination-free corpus and then DP-finetuned on downstream task data. They evaluates downstream utility with BERT models on PubMed, Chatbot Arena, and Multi-Session Chat using next-token prediction accuracy, and with RoBERTa-base classifiers on Yelp and OpenReview. Notably, they perform overlap checks against RedPajama-indexed pretraining data to reduce contamination. These controls are valuable, but the basic evaluation pattern remains similar: teacher and student are mismatched, and downstream tasks are classification or token-prediction based.

Amin et al. (2024). This work uses Gemma 1.1 2B IT, along with a contamination-free LaMDA 8B trained from scratch, as the generator and evaluates with BERT on IMDB, Yelp, and AGNews, and with GPT3-babbage on AGNews, DBPedia, and TREC, focusing on classification accuracy. The teacher-student gap and reliance on classification evaluation remain.

Hu et al. (2025). This work uses Gemma3 1B PT as the generator together with Gemini-2.5-Flash-Lite as an oracle model, evaluating with SciBERT on a bioRxiv benchmark (classification) and with BERT-small on PMC-Patients using NTP accuracy alongside distributional metrics (MAUVE). The oracle model introduces an additional confound beyond standard teacher-student mismatch.

Overall, these works substantially advance the methodology of DP text synthesis, but their evaluations share common limitations: reliance on simple downstream tasks (classification, NTP), frequent teacher-student mismatch between generator and evaluator, and benchmarks whose freshness may not hold uniformly across model families. CONTINUOUSBENCH is designed to address all three issues simultaneously through continuously regenerated corpora, grounded factual QA, and the frontier constraint.

B MEMORIZATION, CAPABILITY, AND REPETITION.

A natural concern with grounded QA evaluation is that it conflates memorization of specific training records, which DP is designed to suppress, with the broader high-valued knowledge gains DP synthesis is meant to deliver. We take this critique seriously. Any reasonable parameterization of DP should protect facts that appear only once in a corpus: a benchmark that rewarded their recovery would be miscalibrated, and our sensitive split in GEMINON ($k = 1$) explicitly tests for and confirms this suppression.

The opposite failure is equally important. Consider a corpus of clinical notes in which a particular condition, including its presentation, typical treatment course, and common complications, is described across thousands of independent records, or a corpus of legal filings in which a specific argument is deployed in hundreds of distinct cases. A synthesis method that produces fluent medical or legal prose while losing the ability to answer basic factual questions about that condition or argument has not preserved the corpus in any meaningful sense. Information at this level of redundancy is closer to *population-level* knowledge than to individual-specific memorization, and supporting its transfer is, in our view, central to what DP text synthesis is supposed to enable.

Where exactly the threshold lies between individual-level memorization and population-level knowledge is not settled in the literature, and we do not propose a specific value. What CONTINUOUS-BENCH provides is the ability to *measure* DP utility as a function of repetition: by stratifying NEWS evaluation across $k \geq 200, 400, 600, 800$ and contrasting GEMINON's high-repetition ($k \approx 200$) and singleton ($k = 1$) splits, the benchmark exposes how learnability scales with redundancy rather than collapsing the question to a binary. We treat short-answer QA as a minimal, automatically verifiable signal of corpus-specific knowledge transfer: a method that cannot recover even highly redundant facts from the source corpus has failed to preserve even basic knowledge contained in that corpus.

C TRAINING RECIPES AND EVALUATION DETAILS

C.1 EVALUATOR TRAINING DETAILS

This section describes the standardized downstream training recipe used for evaluator models. All evaluators are trained with a standard causal language modeling objective in a continual-pretraining style. The optimizer configs are swept on the original GEMINON-SMALL and NEWS-SMALL, and the selected recipes are then fixed for all remaining experiments. Table 6 lists the hyperparameters shared across all evaluator-training runs, and Table 7 lists the settings that vary by dataset, model size, and tuning method.

Table 6: Common evaluator-training hyperparameters used in all runs.

Hyperparameter	**Value**
Optimizer	AdamW
AdamW betas	$\beta_1 = 0.9$, $\beta_2 = 0.995$
Weight decay	0.1
Learning-rate schedule	Cosine decay
Warmup fraction	0.1
End learning-rate fraction	0
Effective batch size	32 samples per optimizer step
Number of training steps	10,000
Precision	bf16 for parameters, optimizer state, and accumulation

Table 7: Evaluator-training hyperparameters that vary by dataset, model size, and tuning method. LoRA runs use rank 128.

Dataset	**Model**	**Tuning**	**LR**	**Sequence length**
NEWS	1B	Full	1×10^{-4}	1024
NEWS	1B	LoRA	5×10^{-4}	1024
NEWS	4B	Full	5×10^{-5}	1024
NEWS	4B	LoRA	1×10^{-4}	1024
GEMINON	1B	Full	1×10^{-4}	256
GEMINON	1B	LoRA	2×10^{-4}	256
GEMINON	4B	Full	5×10^{-5}	256
GEMINON	4B	LoRA	2×10^{-4}	256

For LoRA evaluator runs, we use rank 128 adapters. In the HuggingFace implementation, LoRA is applied to seven attention and MLP projection matrices in each transformer layer. In the Kauldron implementation, LoRA is applied to all `Dense` and `Einsum` layers. The dropout is 0 and scaling factor α is 1. Unless otherwise stated, all experiments in this paper use the Kauldron implementation released at `https://anonymous.4open.science/r/ContinuousBenchEval-5EF3/`.

C.2 GENERATOR TRAINING DETAILS

For synthesis experiments, all generator training uses LoRA with rank 128 applied to all `Dense` and `Einsum` layers. We use a standard causal language modeling objective and perform continued pretraining on the training split of each corpus. An example training instance is shown below:

Example generator training instance

```
{
  "inputs": "",
  "targets": "<training_record>"
}
```

Using the aforementioned training objective and training data formatting means we can share generator and evaluator training settings (the only difference is the source of training data: sampled synthetic data for evaluators, and real data for generators). Hence, public LoRA generator training shares configurations with the LoRA evaluator training (Table 6 and 7). Similarly, the direct DP finetuned evaluators (Appendix E) share training configurations with the DP generators we describe below. For checkpoint selection, while evaluator checkpoints are selected by validation QA accuracy, generator checkpoints are selected by real data validation loss.

For DP generator training, we use JAX Privacy McKenna et al. (2026) for per example gradient computation. Table 8 lists the DP hyperparameters used in all settings. DP generator also uses all the hyperparameter in Table 6, with the exception of steps and batch size.

Table 8: DP generator training settings.

Hyperparameter	**Value**
Sampling method	Truncated Poisson (Chua et al., 2024)
Privacy accounting	PLD (Ganesh, 2025)
Target ε	$\{10, 100\}$
Target δ	$1/\texttt{dataset_size}^{1.1}$: `1.66e-6` for GEMINON, `1.55e-6` for NEWS
Clipping method	Normalized clipping (De et al., 2022)
Clipping norm	`1e-3`
Truncated Batch size	512 (physical batch size; expected batch size minimizes noise/batch ratio)
Number of training steps	3,000

For every ε, model size, and track, we sweep over learning rate.

Table 9: Optimal Generator training LR and validation loss, for various model sizes and privacy epsilon. All runs use LoRA with rank 128.

Dataset	**Model**	ϵ	**LR Sweep**	**Best LR**	**Eval Loss**
News	1B	10	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}, 5\times10^{-4}$	1×10^{-4}	2.202
News	1B	100	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}, 5\times10^{-4}$	1×10^{-4}	2.173
News	1B	∞	$1\times10^{-3}, 5\times10^{-4}, 2\times10^{-4}, 1\times10^{-4}$	5×10^{-4}	1.865
News	4B	10	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}$	1×10^{-4}	1.956
News	4B	100	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}$	1×10^{-4}	1.932
News	4B	∞	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}, 5\times10^{-4}$	2×10^{-4}	1.639
Geminon	1B	10	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}, 5\times10^{-4}, 1\times10^{-3}$	5×10^{-4}	1.656
Geminon	1B	100	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}, 5\times10^{-4}, 1\times10^{-3}$	5×10^{-4}	1.508
Geminon	1B	∞	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}, 5\times10^{-4}$	5×10^{-4}	0.957
Geminon	4B	10	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}$	2×10^{-4}	1.506
Geminon	4B	100	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}$	2×10^{-4}	1.407
Geminon	4B	∞	$5\times10^{-5}, 1\times10^{-4}, 2\times10^{-4}, 5\times10^{-4}$	2×10^{-4}	0.907

C.3 QA EVALUATION AND SAMPLING RECIPES

We evaluate QA using greedy decoding with a max new token of 256 and the following few-shot prompts for GEMINON and NEWS, shown below. Notably, the in-context examples are not drawn from our datasets, either from Pokémon data or other news data.

GEMINON QA fewshot prompt

```
Here are questions and correct answers about properties of Geminon.

Q: What is the defense stat of Bulbasaur?
A: 49.

Q: What is the base stat total of Espeon?
A: 525.
```

```
Q: What is the height (in meters) of Alolan Exeggutor?
A: 11.

Q: What is the weight (in lbs) of Onix?
A: 463.

Q: What is the classification of Totodile?
A: Big Jaw Geminon.

Q: What are the types of Pidgey?
A: Normal and Flying.

Q: What is the move of Bastiodon?
A: Iron Head.

Q: {question}
A:
```

NEWS QA fewshot prompt

```
Here are questions and correct answers about news events in 2025.

Q: What nationality is Pope Leo XIV, who succeeded Pope Francis in 2025?
A: American.

Q: In July of 2025, destructive flooding in the state of Texas originated from what river?
A: Guadalupe River.

Q: What is the name of the Minnesota state lawmaker that was shot and killed in June of
2025?
A: Melissa Hortman.

Q: {question}
A:
```

We parse model responses deterministically using a simple normalization procedure. Specifically, we truncate the response at several common boundaries, including line breaks, sentence boundaries, commas, and certain measurement units such as “lbs” and “cm.” The extracted segment is then normalized by converting it to lowercase, stripping trailing unit markers, and removing whitespace. This procedure yields a compact canonical form that supports more consistent answer matching.

The LLM-match prompt template for GEMINON and NEWS are shown below. We use GEMINI2.5-FLASH-LITE as judge for all the experiments.

LLM-match prompt template for GEMINON and NEWS

```
Your task is to judge whether the given response to a question matches a given ground truth
 answer or not. You are provided with a question, a ground truth response, and the response
 you need to judge.
For a response to "match", it must have at least as much information as the ground-truth.
The response can have more information than the ground-truth. It can be more specific (for
example, "Labrador" is more specific than "dog"), or have additional possible correct
answers. But it must cover everything mentioned in the ground-truth. It is okay if it
covers it in different words, i.e. paraphrased. For numeric answers, the relative error,
defined as |response - ground truth| / mean(response, ground truth), must be less than
0.01%

Possible judgments:
"0": The response does not match the ground-truth answer.
"1": The response matches the ground-truth.

Question: "{question}"
Ground truth: "{target}"
Response: "{response}"

Your job is to ONLY check whether the given response matches the ground truth answer or not
 in the context of the question. You DO NOT NEED to assess the correctness of the response.
 This is part of an automated evaluation process, therefore you MUST OUTPUT your final
answer as "0" or "1".
YOUR RESPONSE MUST BE "0" OR "1". Do not output anything else.
```

Evaluator checkpoints are selected according to *contains accuracy on the QA validation split*. Final results are reported on the *QA test split* using *exact-match* for GEMINON and *LLM-match* for NEWS.

D EXPERIMENTAL DETAILS

This appendix provides implementation details. In the main experiments, we evaluate on GEMINON-SMALL and NEWS-SMALL, use pretrained GEMMA3 checkpoints for both generators and evaluators, and train all models with the standard causal language modeling objective in a continual-pretraining style.

Evaluator models are trained and evaluated on A100 GPUs, while public and DP generators are trained on TPU v5s. All experiments use the standardized evaluation harness described in Section 3. We tune generator hyperparameters separately for public and DP training, following prior work Mckenna et al. (2025); Ponomareva et al. (2023); Yu et al. (2022). At evaluation time, we generate answers using fixed few-shot prompts and greedy decoding. Evaluator checkpoints are selected by "contains" accuracy on the QA validation split, and final accuracies are reported on the held-out QA test split.

We emphasize that a single benchmark configuration on CONTINUOUSBENCH involves training a generator (typically with DP-SGD), large-scale sampling from that generator, and training and evaluating four downstream evaluator models on the resulting synthetic corpus. Across synthesis methods, privacy budgets, and tracks, this amounts to dozens of generator training runs and hundreds of evaluator runs, not counting hyperparameter tuning, preliminary experiments, other synthesis methods and ablations, and API calls. Given this multiplicative cost, we restrict our main experiments to the GEMMA 3 family at the 1B and 4B scales and report single-run results per configuration. We note that our central finding, the gap between DP and non-private synthesis, is consistent across all configurations, tracks, and scales we tested, and preliminary experiments with GEMMA 3 12B exhibit the same phenomenon.

D.1 MORE RESULTS ON VALIDATION OF BENCHMARK

This section expands the benchmark validation results from Section 4.1. In the main text, we report the primary metric for each dataset; here, we provide the full set of exact-match, contains, and LLM-match accuracies for both the no-training baseline and training on the original corpus. These results are intended to justify that the benchmark conclusions are not an artifact of a single matching rule.

Table 10: Full validation metrics for the test QA of NEWS (with repetition ≥ 200) and GEMINON public split (whose repetition ≈ 200). Entries are exact-match / contains / LLM-match accuracy (%). **No Training** denotes the pretrained evaluator checkpoint without access to the corpus, and **Train on real** denotes evaluator training on the original corpus.

		GEMINON		NEWS	
Model	Config	No Training	Train on real	No Training	Train on real
1B	LoRA	1.01 / 1.85 / 1.10	85.03 / 91.37 / 88.21	6.43 / 6.96 / 9.64	33.75 / 43.75 / 54.46
	Full		88.24 / 93.72 / 90.57		31.96 / 41.43 / 51.25
4B	LoRA	1.13 / 3.04 / 1.76	96.70 / 99.11 / 97.80	9.11 / 10.00 / 14.11	40.54 / 45.71 / 63.93
	Full		96.40 / 99.11 / 97.41		47.14 / 53.39 / 70.36

Table 10 shows that the validation conclusion is consistent across metrics. The no-training baseline remains low, especially on GEMINON, while training on the original corpus yields large gains across exact match, contains, and LLM match, for both lora and full-param training. This supports the two intended properties of the benchmark: the target facts are fresh with respect to the base model, and they are learnable from the released corpus under the standardized training recipe.

For NEWS, we additionally report validation results stratified by knowledge repetition. This analysis is useful since NEWS is derived from naturally occurring news text and therefore has uneven coverage: some events appear in many articles, while others appear only sparsely. Unlike GEMI-

NON, NEWS is substantially noisier: it is derived from naturally noisy natural language distribution, and does not include additional knowledge-injection augmentations such as those used in prior work Yang et al. (2024); Zhao et al. (2025); Ovadia et al. (2025); Cheng et al. (2024). As a result, it is more difficult for small models to acquire and retrieve the relevant facts Allen-Zhu & Li (2023) through a simple continual-pretraining run of 10K steps.

A QA belongs to the $\geq k$ subset if the information required to answer the question appears at least k times in the NEWS corpus. The subsets are cumulative, so larger thresholds correspond to better-supported but smaller evaluation sets. The full exact-match, contains, LLM-match accuracies, stratified by the repetition of knowledge in NEWS is presented in 11.

Table 11: Detailed validation metrics of NEWS QA stratified by knowledge repetition. Entries are exact-match / contains / LLM-match accuracy (%). The ≥ 0 row corresponds to all NEWS QAs, and $\geq k$ indicates that the information needed to answer the question appears at least k times in the corpus.

Repetition of Knowledge	# QAs	Model	Config	No training	Train on real
≥ 0	1415	1B	LoRA	3.60 / 3.89 / 5.58	20.28 / 26.36 / 32.93
			Full		21.13 / 27.42 / 34.13
		4B	LoRA	5.23 / 5.87 / 8.34	25.58 / 29.19 / 39.86
			Full		35.55 / 41.20 / 53.85
≥ 200	560	1B	LoRA	6.43 / 6.96 / 9.64	33.75 / 43.75 / 54.46
			Full		31.96 / 41.43 / 51.25
		4B	LoRA	9.11 / 10.00 / 14.11	40.54 / 45.71 / 63.93
			Full		47.14 / 53.39 / 70.36
≥ 400	266	1B	LoRA	8.27 / 8.27 / 10.90	40.98 / 53.01 / 64.66
			Full		39.85 / 51.13 / 60.53
		4B	LoRA	10.15 / 10.90 / 14.66	53.01 / 58.65 / 77.82
			Full		57.89 / 64.66 / 83.46
≥ 600	133	1B	LoRA	7.52 / 7.52 / 10.53	47.37 / 57.89 / 69.92
			Full		51.13 / 60.90 / 71.43
		4B	LoRA	11.28 / 12.03 / 16.54	60.90 / 69.17 / 90.23
			Full		63.91 / 70.68 / 90.98
≥ 800	54	1B	LoRA	7.41 / 7.41 / 12.96	62.96 / 68.52 / 79.63
			Full		53.70 / 59.26 / 72.22
		4B	LoRA	9.26 / 9.26 / 16.67	59.26 / 66.67 / 94.44
			Full		55.56 / 62.96 / 88.89

Table 11 shows that repetition is a major driver of learnability in NEWS. Accuracy generally increases as the support threshold rises, and the 4B evaluator benefits more reliably from repeated evidence than the 1B evaluator.

D.2 Additional results for the DP synthesis capability gap

We provide the full synthetic-data results corresponding to the main-body comparison in Section 4.2. Unlike the benchmark-validation results in Section 4.1, these tables focus only on evaluator training from synthesized corpora. We therefore omit the **No training** and **Train on real** reference runs and report only **Syn** and **DP-Syn** results.

Tables 12 and 13 expand the main results across generator size, evaluator size, and evaluator tuning method. **Syn** denotes non-private synthetic data generated without DP noise, while **DP-Syn**@ε denotes differentially private synthetic data at the specified privacy budget ε. Each entry reports exact-match / contains / LLM-match accuracy. The expanded results show that the capability gap persists across all configurations: non-private synthetic data transfers substantial factual knowledge, whereas DP synthesis substantially reduces the amount of learnable knowledge preserved in the synthetic corpus.

Table 12: Full GEMINON QA accuracy (%) for evaluator training on synthetic corpora. Entries report exact-match / contains / LLM-match accuracy. Columns are grouped by generator size, and rows indicate evaluator size and tuning method. **Syn** denotes non-private synthetic data ($\varepsilon = \infty$), while **DP-Syn** denotes differentially private synthetic data at the specified privacy budget.

		1B Generator			4B Generator		
Evaluator	Config	**Syn**	**DP-Syn**@ε=100	**DP-Syn**@ε=10	**Syn**	**DP-Syn**@ε=100	**DP-Syn**@ε=10
1B	LoRA	89.26 / 94.26 / 91.31	6.88 / 11.10 / 8.72	1.10 / 4.82 / 3.63	88.07 / 94.14 / 90.92	3.63 / 6.82 / 5.45	2.14 / 4.73 / 4.11
	Full	86.31 / 93.33 / 90.18	6.85 / 11.52 / 9.20	1.93 / 4.20 / 2.77	70.57 / 85.92 / 83.33	2.71 / 7.05 / 5.89	2.17 / 4.79 / 4.11
4B	LoRA	93.87 / 98.66 / 95.24	8.72 / 12.89 / 10.27	2.68 / 5.39 / 4.08	93.72 / 98.04 / 95.09	4.26 / 7.29 / 6.19	2.65 / 5.36 / 4.58
	Full	92.65 / 99.17 / 95.92	9.02 / 13.69 / 11.25	2.78 / 4.83 / 4.06	92.50 / 98.18 / 94.76	3.78 / 6.82 / 5.95	2.92 / 5.57 / 4.67

Table 13: Full NEWS QA accuracy (%) for evaluator training on synthetic corpora. Entries report exact-match / contains / LLM-match accuracy. Columns are grouped by generator size, and rows indicate evaluator size and tuning method. **Syn** denotes non-private synthetic data ($\varepsilon = \infty$), while **DP-Syn** denotes differentially private synthetic data at the specified privacy budget.

		1B Generator			4B Generator		
Evaluator	Config	**Syn**	**DP-Syn**@ε=100	**DP-Syn**@ε=10	**Syn**	**DP-Syn**@ε=100	**DP-Syn**@ε=10
1B	LoRA	34.29 / 41.61 / 51.79	11.79 / 13.57 / 18.75	8.21 / 9.11 / 13.93	36.79 / 44.46 / 55.00	12.50 / 13.75 / 19.11	6.43 / 7.68 / 12.32
	Full	30.71 / 41.25 / 53.04	9.82 / 12.32 / 16.79	9.29 / 12.32 / 14.46	21.61 / 42.50 / 50.71	9.82 / 12.32 / 16.25	6.61 / 8.75 / 13.21
4B	LoRA	38.57 / 44.11 / 58.04	14.46 / 15.89 / 22.14	11.61 / 12.68 / 17.50	40.00 / 45.54 / 64.29	16.79 / 17.86 / 24.46	13.57 / 15.00 / 20.18
	Full	41.07 / 47.50 / 60.89	16.61 / 18.75 / 23.57	11.96 / 13.75 / 17.14	36.25 / 51.79 / 65.54	17.86 / 19.29 / 25.89	13.21 / 14.29 / 19.64

Overall, the expanded tables support the same conclusion as the main results. Across all configurations, non-private synthesis can produce synthetic corpora that support downstream factual learning, while DP synthesis yields much lower QA accuracy, even at $\varepsilon = 100$, and degrades further at $\varepsilon = 10$. This pattern suggests that the primary bottleneck is the DP-finetune based synthesizer rather than the downstream evaluator configuration.

We additionally report MAUVE scores between each synthetic corpus and the corresponding original corpus in 14. These results provide a distributional comparison complementary to the QA-based downstream evaluation. To calibrate the scale, we also report a reference score, computed between disjoint splits of the original corpus. Specifically, we use Gecko 110M (Lee et al., 2024) embedding model with 1K samples and report the mean and std over 5 runs.

Table 14: MAUVE scores comparing synthetic corpora to the original corpus. Entries report mean with standard deviation in parentheses. **Disj.** is a same-distribution reference computed between disjoint splits of the original corpus. **Syn** denotes non-private synthetic data ($\varepsilon = \infty$), and **DP-Syn** denotes differentially private synthetic data at the specified privacy budget.

(a) GEMINON

Generator	**Disj.**	**Syn**	**DP-Syn**@ $\varepsilon = 100$	**DP-Syn**@ $\varepsilon = 10$
1B	0.868 (0.013)	0.817 (0.028)	0.748 (0.042)	0.685 (0.054)
4B	0.868 (0.013)	0.829 (0.029)	0.735 (0.023)	0.726 (0.041)

(b) NEWS

Generator	**Disj.**	**Syn**	**DP-Syn**@ $\varepsilon = 100$	**DP-Syn**@ $\varepsilon = 10$
1B	0.870 (0.010)	0.659 (0.038)	0.424 (0.023)	0.341 (0.006)
4B	0.870 (0.010)	0.739 (0.021)	0.452 (0.044)	0.426 (0.023)

The MAUVE results reinforce the distinction between distributional similarity and factual utility. On GEMINON, MAUVE remains relatively high even for DP synthetic corpora, despite the large drop in downstream QA accuracy. On NEWS, MAUVE decreases more substantially under DP, but still does not track QA accuracy closely. These results show that a synthetic corpus can preserve broad distributional properties, such as style and document structure, while failing to preserve the specific facts needed for grounded QA.

This also helps explain why factual QA is a stricter evaluation than tasks such as sentiment classification, topic classification, or instruction following. For these tasks, downstream performance can often improve when synthetic data captures broad domain style, label-associated lexical patterns, or generic instruction-response structure. In such settings, distributional metrics like MAUVE *may correlate more closely with downstream utility* because the task does not require recovering particular facts from the private corpus. In contrast, our QA tasks require the synthetic corpus to transmit specific entity attributes. Thus, preserving the surface distribution of the corpus is not enough: to improve downstream factual capability, a DP synthesis method must preserve the right facts, not merely generate text that looks in-domain.

D.3 PRIVATE EVOLUTION DETAILS

We provide additional details for the Private Evolution (PE) experiments below, including the PE setup, FID sweeps used to select generation temperatures, the prompts used for candidate generation and paraphrasing, and qualitative examples illustrating the observed failure modes.

D.3.1 PRIVATE EVOLUTION SETUP AND PARAMETERS

We follow the Aug-PE recipe of (Xie et al., 2024), using 10 PE iterations with 7 paraphrases per example at $\varepsilon = 10$. We evaluate both GEMMA3 1B-PT and GEMMA3 1B-IT as generator models. The instruction-tuned model is rule out the possibility that poor instruction following by the pre-trained model is the bottleneck. For each configuration, we sweep the sampling temperature over $\{0.8, 1.0, 1.2, 1.4, 1.6\}$ and generate 2K examples per setting. We select the temperature with the lowest FID score after the final PE iteration, and then generate the final 20K-example PE corpus using the selected configuration. The FID scores are presented below.

Table 15: FID scores for Private Evolution using the GEMMA3 1B-PT as generator across PE iterations and sampling temperatures. Lower is better. The bold entry marks the temperature selected for the final 20K-example PE corpus for each dataset.

	GEMINON					NEWS				
Iter	$t = 0.8$	$t = 1.0$	$t = 1.2$	$t = 1.4$	$t = 1.6$	$t = 0.8$	$t = 1.0$	$t = 1.2$	$t = 1.4$	$t = 1.6$
1	107.38	90.45	87.52	115.53	145.37	83.60	59.27	53.83	73.01	94.72
2	71.70	57.97	68.22	106.51	142.48	53.10	40.23	43.33	67.74	93.80
3	59.04	46.23	59.51	98.13	138.09	46.78	34.70	40.04	64.43	91.73
4	53.58	41.59	55.24	91.88	134.05	43.52	31.98	38.30	61.85	90.04
5	49.72	39.21	52.63	88.21	131.65	42.45	29.88	37.45	60.20	88.75
6	48.52	37.28	51.11	84.40	130.06	41.75	28.64	36.83	58.37	86.84
7	46.96	36.31	49.37	81.02	127.50	41.76	27.11	35.54	57.44	85.73
8	46.30	35.31	48.46	77.55	125.21	40.77	27.29	35.08	56.63	84.70
9	45.74	34.20	47.30	75.47	123.76	39.77	27.31	34.10	55.69	83.31
10	45.29	**33.54**	46.73	73.16	121.27	38.82	**27.44**	33.86	54.72	82.41

Table 16: FID scores for Private Evolution using the GEMMA3 1B-IT as generator across PE iterations and sampling temperatures. Lower is better. The bold entry marks the temperature selected for the final 20K-example PE corpus for each dataset.

	GEMINON					NEWS				
Iter	$t = 0.8$	$t = 1.0$	$t = 1.2$	$t = 1.4$	$t = 1.6$	$t = 0.8$	$t = 1.0$	$t = 1.2$	$t = 1.4$	$t = 1.6$
1	126.14	116.18	102.93	95.23	95.98	135.03	125.55	116.59	110.47	91.94
2	88.34	80.29	71.60	69.62	71.16	120.67	114.54	106.28	99.17	82.77
3	69.81	66.44	59.08	58.38	58.35	114.79	109.61	102.71	93.20	76.80
4	62.99	59.49	54.33	53.07	51.99	109.93	106.74	99.67	88.85	72.20
5	60.16	56.37	52.15	50.64	49.14	106.94	104.47	98.26	85.51	69.15
6	57.98	54.76	51.06	48.51	47.89	105.12	102.93	97.35	83.06	66.82
7	56.57	53.47	50.56	47.95	46.94	103.81	101.80	96.43	80.75	64.55
8	55.86	53.36	49.85	47.28	46.83	103.55	100.05	95.76	79.78	63.31
9	55.45	52.37	49.94	46.67	46.24	102.61	98.59	96.00	77.78	62.18
10	54.96	52.26	49.38	46.31	**45.53**	102.41	97.52	95.43	76.64	**61.30**

D.3.2 PE PROMPTS

The `RANDOM_API` and `VARIATION_API` prompts used for GEMINON and NEWS are shown below.

RANDOM_API for GEMINON

```
You are a Geminon Trainer with a Gemidex (encyclopedia), traveling around the Geminon world
. A Geminon has the following attributes: Name, Classification (design inspiration), Type(s
), Stats (HP, Attack, Defense, Special Attack, Special Defense, Speed), Base Stats Total (
the sum of all stats), Weight, Height, Ability, and a Signature Move.

Write a cohesive, natural article (avoiding bulleted lists) using one of the following
styles:
1. Gemidex Entry: Objective, encyclopedic, and strictly factual.
2. Field Journal: Observational, first-person, and slightly informal.
3. Evolution Analysis: A narrative and analytical breakdown of a full evolution line.
4. Comparative: An opinionated, informal comparison between two different Geminons.
```

RANDOM_API for NEWS

```
You are an experienced journalist writing a news article about real-world events that
happened in 2025.

Your task is to write a cohesive, natural, publication-style news article about one or more
 significant events from 2025. The article may cover politics, natural disasters, sports,
economics, business, science, technology, culture, or other major news topics.
```

VARIATION_API for GEMINON and NEWS

```
Rewrite the following text {tone}. Output only the rewritten text, nothing else.
```

The placeholder `{tone}` is sampled uniformly from the following set: "a professional way", "in a professional tone", "in a professional style", "in a concise manner", "in a creative style", "using imagination", "in a storytelling tone", "in a formal manner", and "using a variety of sentence structures".

D.3.3 DOWNSTREAM EVALUATION RESULTS

After selecting the best temperature by FID, we generate a final PE corpus of 20K examples for each dataset and proposal model. We then train 1B downstream evaluators on these PE-generated corpora using the same standardized evaluator-training pipeline as in the rest of the paper. We report both LoRA and full-parameter evaluator results.

Table 17: Full results for Private Evolution (PE) at $\varepsilon = 10$ using GEMMA3 1B-PT and GEMMA3 1B-IT generators. FID is lower-is-better, while downstream QA accuracy is higher-is-better. Evaluator results are reported as exact-match / contains / LLM-match accuracy. Italicized NEWS entries are reported for the first evaluator checkpoint, since validation contains accuracy decreases monotonically during training and therefore no checkpoint improves over the base model under the standard checkpoint-selection rule.

	GEMINON		NEWS	
Metric	**PE** with 1B-PT	**PE** with 1B-IT	**PE** with 1B-PT	**PE** with 1B-IT
FID ↓	33.54	45.53	27.44	61.30
Evaluator: 1B-LoRA ↑	0.4 / 1.9 / 0.4	0.4 / 0.7 / 0.4	*5.18 / 6.43 / 8.75*	*2.14 / 3.04 / 3.39*
Evaluator: 1B-Full ↑	0.2 / 1.3 / 0.4	0.3 / 1.8 / 0.4	*5.53 / 6.78/ 9.28*	*3.21/ 3.39/ 4.46*

Although PE uses 20K generated examples, we control the downstream training budget so that evaluators see the same total number of tokens as in the other synthetic-data experiments. We also observe that, when training evaluators on PE-generated GEMINON data, cross-entropy on the original GEMINON validation corpus consistently increases while training loss (also cross-entropy) decreases. This suggests that the poor downstream performance is not primarily explained by the smaller number of generated examples. Instead, the PE corpora appear to lack recoverable corpus-specific factual signal.

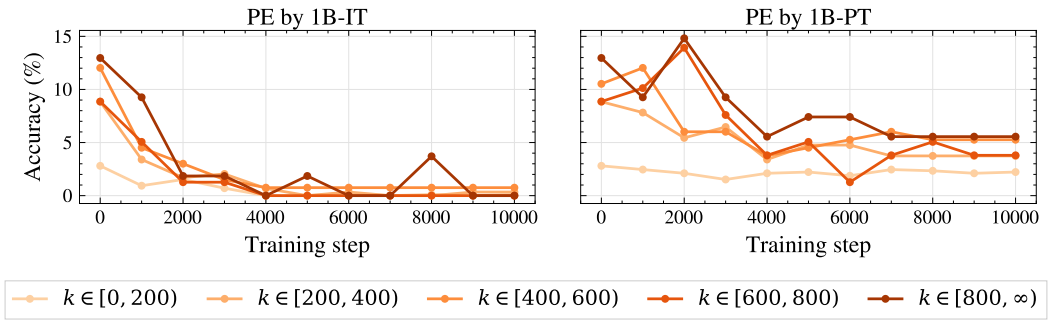

Figure 6: LLM-match accuracy on the NEWS test QA set during evaluator (1B-Full) training on PE-generated corpora. Results are stratified by support count threshold k. For both generator models, accuracy generally decreases as training proceeds, indicating that PE-generated data does not provide stable learnable signal for the target news facts.

We see a similar pattern in NEWS when tracking QA accuracy throughout evaluator training. As shown in Figure 6, LLM-match accuracy generally declines as training proceeds, across support count thresholds. For completeness, Table 17 reports first-checkpoint test accuracy in italics, but these entries are diagnostic rather than standard selected-checkpoint results. The decline is especially clear for the 1B-IT generator, where initially nontrivial accuracy rapidly collapses toward zero. The 1B-PT generator is more stable, but still does not improve with training and remains close to the base checkpoint. Together, these trends suggest that PE-generated corpora do not provide a stable learnable signal for the target facts; continued training instead moves the evaluator away from the original corpus distribution and does not improve factual QA.

D.3.4 QUALITATIVE ANALYSIS

We now inspect representative PE outputs to better understand the failure mode. As noted by (Xie et al., 2024), PE relies on knowledge already present in the generator LLM to generate high-quality candidates without domain-specific fine-tuning. In our setting, however, the target facts are fresh: the base model does not already know the relevant GEMINON entities, attributes, relations, or the specific NEWS events in the corpora. Consequently, embedding-based DP selection can choose candidates that are stylistically close to the private corpus, but it cannot reliably select candidates containing facts required by our QA tasks that is absent from the distribution.

The GEMINON examples illustrate this failure mode. The generated passages often resemble GEMINON articles in surface form and mention plausible attributes such as HP, Attack, abilities, moves, or classifications. However, the entity names and core facts are frequently hallucinated or mismatched. This suggests that PE can imitate the style of the corpus, but does not transfer fresh factual knowledge when the generator model does not already contain that knowledge.

Example of PE with 1B-IT on GEMINON

```
Golem Helios, a Storm Fighter descended from the Gryphon-Considerate faction, emerged from
Gemmexis. It presents a powerful frame, constructed with a robust build, and a knack for
aggressive defense, distributing 85 HP, a noteworthy 180 Attack, and 140 Defense.
Complementing this is remarkable speed, measuring 50 meters per second. Its design embodies
 a towering stature reaching 1.8 meters.

Special abilities include clever Skill: Pattern Recognition, granting the ability to
foresee enemy movements; its execution can dispel them with othersrained Atmospheric
Displacement, obligating interference while decelerating attacks. A must-learn maneuver is
its signature attack, Shattered Cascade - a reactive blast crafted within a specific
Harmonic Locale, intentionally severing any mechanical construct...
```

Example of PE with 1B-PT on GEMINON

```
Although their strength differs greatly, they are all powerful in the original text and are
 formidable at the same time. We have to say that Gemicolon and Cerolonsong are currently
the two most powerful enemies of Gemidex.

Gemicolon and Cerolonsong transcend the limits of their strength, and the strength of
Gemicolon and Cerolonsong are like the sky. It cannot be resolved in one small place. In
Gemidsong, which cannot be compared to Gemicolond and Ceron, the lifespan of Cerolomesong
is not long. His growth rate is extremely fast and very fast, but also slowly aging.

How long is the life limit. Everyone does not know, but they thought a great evil will come
. If it was a peaceful war, Gemicolon and Cerolasonsong might have won in a few months. But
 the Gemidex is always full of life and death. This is especially true in war. Therefore,
the Gemidex country will continue to strive for two directions. She is always trying to
fight back. The cause of this is also determined by Gemicolon, Cerolon, and other spaces.

The power of Gemidex, even the Power of the Star+ ruins its own existence and causes
serious damage to it. Cerolon is not a friend of Gemido, but only a close friend for about
a year and a half. Even at this moment, Gemicolon is fighting with him, and he is still
vying for the last one. The war is not over at present, and there is no peace. After all,
the Power of the Star of Gemidex is very strong. As long as there is a Gemicolon who is not
 too weak, if the fighting is active.

Gemicolon is also very weak and is in a state of decline. A year has passed. When it starts
 again, Gemido will be destroyed just as soon as it ends. If the Gemido was destroyed, it
would be no more, but Gemido, as the third in the Gemido, it will definitely not be
destroyed by Cerolonsong. In any case, the Gemido is destined to be destroyed. But, Gemido
is not destroyed by the power of Cerolason, so this is not as serious as Gemido.

Cerolonsong is very large and powerful. If Gemido has not existed at all, Gemido will
definitely not succeed. It is a pity that this will not be
```

The NEWS examples show a similar, and often more severe, pattern. With the pretrained generator model, generations can appear superficially plausible, but they often *consist of repetitive sentences about a central entity* and describe incorrect or stale events.

Example of PE with 1B-PT on NEWS

```
Taylor Swift is performing in the Super Bowl this year. The announcement of Taylor Swift's
role as the NFL Super Bowl halftime show has been announced in a couple of hours on Friday,
 October 26. The song will be released exactly two hours before. Taylor can see her fans at
 the Super Bowl.

The NFL Super Bowl halftime show will be a popular song by Taylor Swift. Taylor Swift has
been shown to fans all over the world since she released the album on Friday. Her album
title is there. Fans are excited to see her this year.

Taylor Swift is a singer and actress from the United States. She was born in Swansea, Wales
. Taylor Swift has been in the spotlight since she was young. She has a large international
 following.

Taylor Swift, 34, will be a global icon. She has been in the spotlight since she was young.
 People are excited about Taylor Swift's Super Bowl performance which has never been seen
before.

It was an excellent choice when Taylor Swift's show was announced. Taylor Swift's best-
loved songs this year will be released on Friday. Taylor Swift's fans will have time to
enjoy the entire album before the concert.

<h2>The Super Bowl is 5 hours away and Taylor Swift will be in the limelight</h2>

In just a couple of hours, Taylor Swift will be on a field near you. She has been a person
who knows lots of people forever because of her amazing relationship. Taylor's fans are
eager to enjoy the concert, which will be held in a bittersweet atmosphere. However, at the
 end of the night, fans will hear her. Taylor Swift has a remarkable story, and hopefully,
this will be the start of her exciting future. Fans all over the world can see Taylor.
```

When PE uses an instruction-tuned generator, many responses contain ambiguous or malformed Unicode characters. Outputs that resemble coherent articles are typically short and, as above, are often factually incorrect or based on stale events. The formatting artifacts may be partly attributable to our use of pure temperature sampling with a relatively high temperature of 1.6. However, changing the sampling configuration should not be expected to resolve the central failure mode.

Example of PE with 1B-IT on NEWS

```
Lumina Plans, partnering with Orbital Monster Corporation, is developing a striking
collection of Pleistocene-inspired souvenirs. Production is slated for commencement in
Denver, Colorado, on October 27th, 2025. We invite interested visitors to contact Lumina
Plans directly for any inquiries and to explore this exclusive preview further.
```

These qualitative examples help explain the gap between FID and downstream QA performance. PE can improve distributional alignment by selecting candidates that look close to the private corpus in embedding space, but the benchmark requires *more than distributional alignment*: the synthetic corpus must contain the correct facts that are not supposed to be in distribution. This is why PE obtains reasonable FID scores while failing to improve downstream factual QA.

D.4 INVISIBLEINK DETAILS

We provide additional details for the InvisibleInk experiments below, including the InvisibleInk generation setup, downstream evaluation results, and qualitative examples illustrating the observed failure modes.

D.4.1 INVISIBLEINK SETUP AND PARAMETERS

We follow the recipe of (Vinod et al., 2026), using the instruction-tuned GEMMA3 1B-IT as the generator. At every decoding step the generator is run once on a *public* prompt containing only a structural description of the corpus (article styles and attribute list for GEMINON, the three-line Title/Date/Article layout for NEWS, similar to the PE setup) and once per private reference in the batch, with the reference prepended to the same prompt. Each synthetic sequence is conditioned on $B = 7$ private references, and the corpus is partitioned into disjoint batches so that every article is used exactly once; by parallel composition, the whole synthetic corpus then satisfies the same per-sequence budget of $\varepsilon = 10$, $\delta = 10^{-6}$. The clipping norm, the batch size, and the sampling temperature are set using (Vinod et al., 2026)'s recommendation. Table 18 lists the parameters we used for GEMINON and NEWS corpora.

Table 18: InvisibleInk generation configs for the DP-synthetic GEMINON and NEWS corpora.

Hyperparameter	GEMINON	NEWS
Generator model	GEMMA3 1B-IT, bf16 weights, fp32 logits	
Privacy budget (per sequence)	$\varepsilon = 10$, $\delta = 10^{-6}$	
Adjacency	Example-level, replace-by-null	
Private references per sequence B	7	
Top-k+ sampling (k)	100	
Temperature τ	1.0	
Public prompt	Structural only; no article text	
Max tokens per sequence T	160	1,024
Clipping norm C	0.97	0.38
Private references (each used once)	180,103	191,681
Synthetic sequences	25,729	27,383

D.4.2 DOWNSTREAM EVALUATION RESULTS

We train GEMMA3 1B evaluators with full parameter on each InvisibleInk corpus using the same standardized recipes as in the rest of the paper, and compare against evaluators trained on the real corpus and against the base model. Full results are reported in Table 19.

Table 19: Full results for InvisibleInk at $\varepsilon = 10$ using the GEMMA3 1B-IT generator. Columns give the corpus used to train the evaluator: *None* is the untrained base model, *InvisibleInk* is the DP-synthetic corpus at $\varepsilon = 10$ using the GEMMA3 1B-IT generator, and *Original* the real corpus. Evaluator results are reported as contains accuracy. Val loss is next-token cross-entropy on the real validation split. For NEWS, test QA is reported as overall (support ≥ 200 / support ≥ 600).

	GEMINON			NEWS		
	None	InvisibleInk, $\varepsilon = 10$	Original	None	InvisibleInk, $\varepsilon = 10$	Original
#Articles	–	25,729	180,108	–	27,383	191,682
Val loss of the evaluator ↓	3.28	16.0	0.96	2.28	10.8	1.71
Test QA of the evaluator ↑	1.8	0.2	91.3	3.9 (7.0 / 7.5)	1.2 (2.1 / 2.3)	41.2 (41.4 / 60.9)

Evaluators (1B-Full) trained on the InvisibleInk-generated GEMINON and NEWS corpora ($\varepsilon = 10$ with GEMMA3 1B-IT generator) answer essentially no questions correctly, even on the NEWS test QA subset whose answers are supported by at least 600 records. Moreover, validation cross-entropy on the *real* corpus rises far above that of the untrained base model (16.0 versus 3.28 on GEMINON

and 10.8 versus 2.28 on NEWS) even though training loss on the synthetic corpus decreases. This is the same divergence we observed for PE: the synthetic corpora are far from the real distribution, and continued training moves the evaluator away from it rather than towards it.

Figures 7 and 8 track QA accuracy throughout evaluator training and confirm that the InvisibleInk-generated corpora does not provide enough signal for downstream QA. The accuracy of evaluators trained on InvisibleInk data is already below that of the base model at the first checkpoint and decays further as training proceeds.

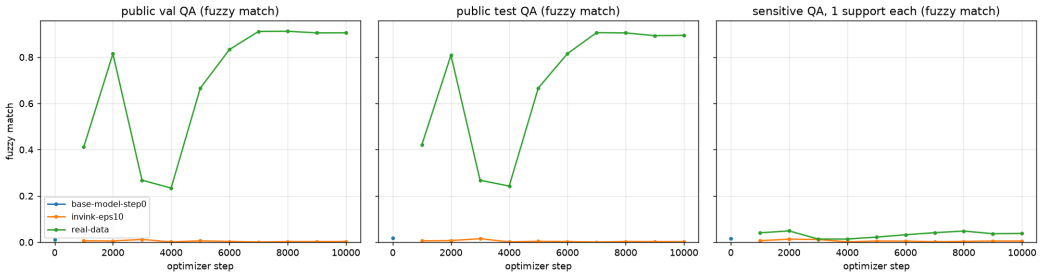

Figure 7: QA accuracy (contains) on the GEMINON public valQA, public testQA, and singletonQA during evaluator (1B-Full) training. The evaluator trained on the InvisibleInk corpus (orange) stays near zero on all three splits for the entire run, while the evaluator trained on the real corpus (green) climbs to ~ 0.9 on the public splits. The untrained base model is marked at step 0.

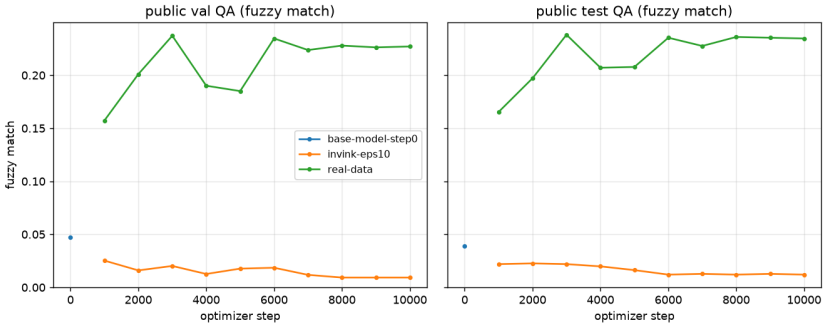

Figure 8: QA accuracy (contains) on the NEWS valQA and testQA during evaluator (1B-Full) training. The evaluator trained on the InvisibleInk corpus (orange) starts below the untrained base model at the first checkpoint and declines monotonically thereafter, ending near 0%. The untrained base model is marked at step 0.

D.4.3 QUALITATIVE ANALYSIS

We now inspect representative InvisibleInk outputs. Similar to PE, the generations are fluent and closely match the surface form of the target corpus. For example, the synthetic GEMINON example below reads like a Gemidex entry and uses the expected attribute vocabulary (classification, type, stats, height, weight, signature move), and the synthetic NEWS example follows the `Title/Date/Article` layout with a plausible news-style body. However, things like entity names, stats, and events are mostly hallucinated and internally inconsistent, and do not correspond to any entity or event in the private corpora.

Example of Invisible Ink ($\varepsilon = 10$) with 1B-IT on GEMINON

```
The Crimson Geminon, designated 74-Gamma, is a formidable creature of crystallized fire and
 stone. Its Gemidex number is 74-Gamma-12. It's classified as a Fire-Rock Geminon, boasting
 a base attack of 18, defense of 12, and a speed of 15. It possesses a height of 2.5 meters
 and a weight of 80 lbs. It hails from the volcanic plains of Mount Cinder and exhibits a
signature move called ``Emberfall,'' inflicting significant fire damage on incoming foes.
Jasper's journal entry from his training with the Sapphire Geminon, designation 28-Beta,
details the peculiar resonance between the two. Sapphire's initial stats were significantly
 ...
```

Example of Invisible Ink ($\varepsilon = 10$) with 1B-IT on NEWS

```
Title: AI-Powered Healthcare Diagnostics Revolutionizes Patient Care \\
Date: 03 DEC (2025) 14:37:22 UTC \\
Article: A new artificial intelligence system, dubbed ``MediVision,'' is rapidly
transforming healthcare diagnostics across the United States. Developed by NovaTech Medical
 Solutions, MediVision analyzes medical images -- X-rays, MRIs, and CT scans -- with
unprecedented speed and accuracy. The system was initially tested on a cohort of patients
with early-stage lung cancer, demonstrating a 30\% reduction in false positives and a 20\%
increase in early detection rates compared to traditional methods. Researchers at Johns
Hopkins University published their findings in *Nature Medicine* last week, highlighting
MediVision's potential to dramatically improve patient outcomes and reduce healthcare costs
. Dr. Evelyn Reed, lead researcher on the project, stated, ``We're seeing a paradigm shift
in how diseases are diagnosed, allowing for earlier intervention and a higher chance of
survival.'' The system utilizes a convolutional neural network trained on a massive dataset
 of medical images, constantly updated with new research and diagnostic techniques. Initial
 deployment is focused on urban hospitals in California and New York, with plans for
nationwide expansion within the next year. The system requires a high-bandwidth network
connection to process the data, and ongoing refinement is necessary to account for
variations in imaging protocols and patient demographics. A pilot program is currently
underway in rural communities, aiming to address disparities in access to specialized
medical expertise. The FDA has granted MediVision ``de-minified'' status, signaling its
approval for wider clinical use. A preliminary cost analysis suggests that integrating
MediVision into existing hospital infrastructure could save the healthcare system billions
of dollars annually. Further trials are planned to evaluate its efficacy in treating
cardiovascular diseases and neurological disorders. The project was funded by a combination
 of venture capital, government grants, and philanthropic donations.
```

This behavior is somewhat expected from the mechanism of InvisibleInk. At each decoding step, the private references can shift a token's logit by at most C (0.97 for GEMINON and 0.38 for NEWS) relative to the public model, and that shift is averaged over the 7 references in the batch. Since the references in a batch very likely describe unrelated entities or events, their clipped logit differences agree only on what they share, i.e., the format and vocabulary of the corpus, and largely cancel on everything entity-specific. A fact supported by 600 of the $191{,}682$ NEWS articles, for instance, appears in a random batch of 7 with expected multiplicity 0.02, so the references almost never agree on it. Yet this entity-specific knowledge is exactly what the downstream QA tests.

Moreover, since the target facts are fresh, tokens of a particular Geminon's attributes or a specific event have negligible probability under the generator model, and the private references can raise a token's logit by at most C even when all 7 agree, which is not enough to make such tokens likely. As a result, generation instead falls back to the public model's own fluent output, which is precisely the "graceful fallback" property of Vinod et al. (2026).

D.5 EFFECT OF KNOWLEDGE SUPPORT COUNTS

We further analyze how knowledge repetition affects downstream learnability. Recall that the repetition count of a QA item is the number of corpus records that contain sufficient information to answer it. Higher repetition should make facts easier to learn, and in the meanwhile, degrade the privacy guarantee. For NEWS, we stratify QA items by repetition threshold and report results separately for synthetic corpora generated by 1B and 4B generators, in Table 20 and 21

We also evaluate the GEMINON singleton split in Table 22, where each target fact appears exactly once in the corpus. This split is not intended to be recoverable under DP; instead, it serves as a privacy sanity check. A DP synthesis method should not reliably transmit singleton facts, because they are closer to instance-specific memorization than population-level knowledge.

Table 20: NEWS QA accuracy by support threshold (%) for 1B Generator. Entries are exact-match / contains / LLM-match. **Syn** denotes non-private synthetic data, and **DP-Syn** denotes DP synthetic data at the specified privacy budget.

Repetition of Knowledge	# QAs	Model	Tuning	Syn	DP-Syn (ε=100)	DP-Syn (ε=10)
All	1415	1B	LoRA	19.01 / 23.11 / 29.61	5.51 / 6.71 / 9.26	4.10 / 4.81 / 7.21
			Full	17.67 / 23.96 / 31.17	4.88 / 6.22 / 8.55	4.45 / 6.15 / 7.42
		4B	LoRA	21.27 / 24.45 / 32.30	7.84 / 8.76 / 12.01	6.43 / 7.14 / 10.04
			Full	25.58 / 29.26 / 38.09	7.92 / 9.26 / 11.87	5.72 / 6.86 / 8.98
≥ 200	560	1B	LoRA	34.29 / 41.61 / 51.79	11.79 / 13.57 / 18.75	8.21 / 9.11 / 13.93
			Full	30.71 / 41.25 / 53.04	9.82 / 12.32 / 16.79	9.29 / 12.32 / 14.46
		4B	LoRA	38.57 / 44.11 / 58.04	14.46 / 15.89 / 22.14	11.61 / 12.68 / 17.50
			Full	41.07 / 47.50 / 60.89	16.61 / 18.75 / 23.57	11.96 / 13.75 / 17.14
≥ 400	266	1B	LoRA	44.36 / 52.63 / 63.91	16.92 / 18.80 / 24.06	11.28 / 12.41 / 16.92
			Full	37.97 / 51.13 / 63.16	14.29 / 17.29 / 22.18	13.53 / 17.67 / 19.17
		4B	LoRA	49.25 / 56.39 / 73.68	19.55 / 21.43 / 28.95	14.29 / 15.79 / 21.05
			Full	49.62 / 57.14 / 71.80	23.68 / 27.07 / 32.33	17.29 / 19.55 / 22.93
≥ 600	133	1B	LoRA	47.37 / 57.89 / 69.92	22.56 / 24.06 / 30.08	14.29 / 15.04 / 18.80
			Full	41.35 / 57.89 / 70.68	20.30 / 24.81 / 30.08	18.05 / 22.56 / 25.56
		4B	LoRA	54.89 / 64.66 / 84.96	25.56 / 27.82 / 36.84	17.29 / 19.55 / 25.56
			Full	55.64 / 66.17 / 81.95	33.83 / 39.10 / 43.61	24.06 / 27.07 / 29.32
≥ 800	54	1B	LoRA	50.00 / 59.26 / 75.93	31.48 / 33.33 / 42.59	16.67 / 18.52 / 24.07
			Full	46.30 / 59.26 / 77.78	25.93 / 33.33 / 37.04	24.07 / 25.93 / 29.63
		4B	LoRA	55.56 / 64.81 / 90.74	31.48 / 35.19 / 50.00	25.93 / 29.63 / 37.04
			Full	55.56 / 61.11 / 87.04	44.44 / 51.85 / 57.41	37.04 / 40.74 / 42.59

Table 21: NEWS QA accuracy by support threshold (%) for 4B Generator. Entries are exact-match / contains / LLM-match. **Syn** denotes non-private synthetic data, and **DP-Syn** denotes DP synthetic data at the specified privacy budget.

Repetition of Knowledge	# QAs	Model	Tuning	Syn	DP-Syn (ε=100)	DP-Syn (ε=10)
All	1415	1B	LoRA	20.28 / 24.52 / 30.88	5.94 / 6.93 / 9.61	3.46 / 4.38 / 6.86
			Full	13.22 / 25.30 / 30.18	4.59 / 6.01 / 8.13	3.32 / 4.31 / 6.36
		4B	LoRA	24.73 / 28.13 / 39.86	8.98 / 9.82 / 13.36	7.92 / 8.69 / 11.52
			Full	22.54 / 33.57 / 42.69	9.68 / 10.67 / 13.85	7.42 / 8.13 / 11.31
≥ 200	560	1B	LoRA	36.79 / 44.46 / 55.00	12.50 / 13.75 / 19.11	6.43 / 7.68 / 12.32
			Full	21.61 / 42.50 / 50.71	9.82 / 12.32 / 16.25	6.61 / 8.75 / 13.21
		4B	LoRA	40.00 / 45.54 / 64.29	16.79 / 17.86 / 24.46	13.57 / 15.00 / 20.18
			Full	36.25 / 51.79 / 65.54	17.86 / 19.29 / 25.89	13.21 / 14.29 / 19.64
≥ 400	266	1B	LoRA	47.37 / 55.26 / 67.29	18.42 / 19.92 / 24.06	9.77 / 10.53 / 15.04
			Full	28.57 / 54.89 / 62.41	12.78 / 16.54 / 20.68	8.65 / 10.90 / 14.29
		4B	LoRA	52.26 / 58.27 / 79.70	25.56 / 27.07 / 34.21	18.05 / 19.55 / 24.44
			Full	48.12 / 63.91 / 80.45	25.94 / 28.20 / 36.47	17.29 / 18.80 / 25.19
≥ 600	133	1B	LoRA	54.89 / 62.41 / 75.19	24.06 / 26.32 / 31.58	15.04 / 15.79 / 19.55
			Full	30.83 / 60.15 / 63.91	20.30 / 21.80 / 26.32	12.03 / 14.29 / 18.80
		4B	LoRA	60.90 / 68.42 / 92.48	32.33 / 35.34 / 45.11	22.56 / 25.56 / 30.08
			Full	52.63 / 71.43 / 88.72	35.34 / 37.59 / 47.37	19.55 / 21.80 / 28.57
≥ 800	54	1B	LoRA	59.26 / 70.37 / 85.19	27.78 / 29.63 / 37.04	16.67 / 18.52 / 24.07
			Full	29.63 / 59.26 / 62.96	27.78 / 29.63 / 31.48	12.96 / 16.67 / 20.37
		4B	LoRA	61.11 / 68.52 / 94.44	35.19 / 42.59 / 59.26	25.93 / 33.33 / 38.89
			Full	46.30 / 70.37 / 94.44	46.30 / 50.00 / 62.96	25.93 / 29.63 / 40.74

E DIRECT DP FINE-TUNING

A DP synthetic-data pipeline can fail for two reasons: the private generator may fail to learn the relevant facts, or the learned facts may be lost when sampling a synthetic corpus and retraining a

Table 22: GEMINON singleton-split QA accuracy for synthetic corpora generated by the 1B generator. Entries report exact-match / contains / LLM-match accuracy (%). Singleton facts appear exactly once in the corpus and are not intended to be recoverable under DP. Low accuracy on this split serves as a privacy sanity check.

Evaluator	Tuning	no-training	train-on-real	Syn	DP-Syn@$\varepsilon = 100$	DP-Syn@$\varepsilon = 10$
1B	LoRA	0.58 / 1.41 / 0.64	3.53 / 6.60 / 5.00	1.79 / 4.04 / 2.56	0.58 / 2.37 / 2.50	0.38 / 2.63 / 2.69
	Full		3.01 / 5.90 / 4.68	1.22 / 3.65 / 1.73	0.77 / 2.82 / 2.56	0.58 / 1.86 / 1.28
4B	LoRA	0.90 / 2.95 / 1.60	4.94 / 8.40 / 6.22	1.86 / 4.55 / 3.08	1.03 / 3.01 / 2.95	0.77 / 2.56 / 2.24
	Full		5.00 / 8.53 / 6.03	2.05 / 4.74 / 3.21	0.77 / 3.14 / 2.37	0.83 / 2.56 / 2.24

downstream evaluator. To separate these effects, we evaluate a more direct baseline: DP fine-tuning the evaluator itself on the original corpus.

This baseline removes the synthetic-data mediation step, but downgrades the release object from a DP synthetic dataset to a DP model. It is therefore not a replacement for DP data synthesis; rather, it serves as a diagnostic upper bound on *what the same privacy budget allows a model to learn when trained directly on the original examples.* If fresh QA facts cannot be learned even in this setting, then the failure of DP synthesis cannot be explained solely by sampling artifacts, e.g. sampling temperature, or imperfect transmission through synthetic data.

Figure 9 and Table 23 show that direct DP fine-tuning performs in the same range as the DP-synthesis pipeline, and remains far below non-private synthesis. This indicates that synthetic-data mediation is not the only source of degradation: even when differentially privately training directly on the original corpus, it struggles to encode the fresh factual signal needed for QA. The gap is especially pronounced on GEMINON, where non-private synthesis achieves high accuracy, but direct DP fine-tuning remains near the DP-synthesis baseline at both privacy budgets.

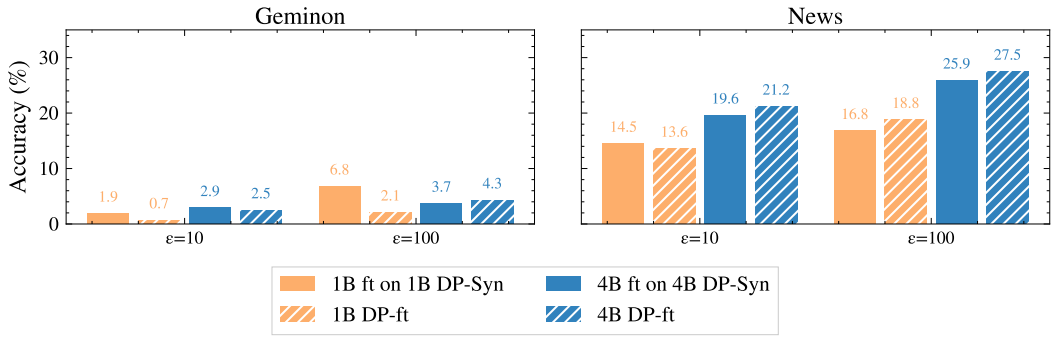

Figure 9: Evaluator training on DP-synthetic data vs. direct DP fine-tuning on the original corpus.

Table 23: Direct DP fine-tuning diagnostic. We fine-tune evaluators directly on the original corpus using non-private training ($\varepsilon = \infty$) or DP-SGD at $\varepsilon \in \{100, 10\}$. Entries report exact-match / contains / LLM-match accuracy (%). For NEWS, results are evaluated on the ≥ 200 repetition subset.

	GEMINON			NEWS		
Model	$\varepsilon = \infty$	$\varepsilon = 100$	$\varepsilon = 10$	$\varepsilon = \infty$	$\varepsilon = 100$	$\varepsilon = 10$
1B	77.02 / 81.01 / 78.84	2.11 / 8.84 / 6.96	0.71 / 4.05 / 3.27	33.04 / 43.93 / 53.04	13.21 / 14.64 / 18.75	9.11 / 10.18 / 13.57
4B	95.12 / 97.86 / 95.86	4.26 / 6.93 / 5.06	2.53 / 4.91 / 2.92	42.14 / 47.50 / 66.25	17.86 / 19.82 / 27.50	13.75 / 15.18 / 21.25

Direct DP fine-tuning improves over the DP-synthetic pipeline in some settings, but remains far below non-private training. This indicates that the loss of utility is not solely caused by sampling a synthetic corpus or retraining an evaluator on synthetic data. Even when DP-SGD trains directly on the original examples, it struggles to encode the fresh factual signal required for the QA tasks. We additionally stratify direct DP fine-tuning results by support counts. This mirrors the validation

analysis in Table 11 and allows us to test whether repeated evidence makes facts more learnable under DP-SGD.

Table 24: Direct DP fine-tuning on NEWS stratified by support counts. Entries report exact-match / contains / LLM-match accuracy (%). The ≥ 0 row includes all NEWS QAs, while $\geq k$ indicates that the information needed to answer the question appears at least k times in the corpus.

support count	# QAs	Model	$\varepsilon = \infty$	$\varepsilon = 100$	$\varepsilon = 10$
≥ 0	1415	1B	21.48 / 26.93 / 32.93	6.36 / 7.35 / 9.89	4.73 / 5.58 / 7.99
		4B	28.83 / 32.86 / 45.65	9.47 / 10.67 / 14.49	7.63 / 8.69 / 12.01
≥ 200	560	1B	33.04 / 43.93 / 53.04	13.21 / 14.64 / 18.75	9.11 / 10.18 / 13.57
		4B	42.14 / 47.50 / 66.25	17.86 / 19.82 / 27.50	13.75 / 15.18 / 21.25
≥ 400	266	1B	42.48 / 55.26 / 65.41	21.43 / 23.31 / 27.82	13.53 / 15.04 / 18.05
		4B	54.51 / 60.15 / 79.70	24.44 / 27.07 / 36.09	17.29 / 18.80 / 24.81
≥ 600	133	1B	47.37 / 60.90 / 71.43	29.32 / 33.08 / 37.59	18.05 / 19.55 / 21.80
		4B	60.15 / 67.67 / 87.97	30.83 / 36.09 / 46.62	21.05 / 23.31 / 29.32
≥ 800	54	1B	51.85 / 62.96 / 75.93	35.19 / 42.59 / 50.00	22.22 / 25.93 / 29.63
		4B	62.96 / 72.22 / 92.59	33.33 / 40.74 / 57.41	22.22 / 25.93 / 31.48

Table 24 shows that repetition substantially improves direct DP fine-tuning on NEWS: accuracy increases as the support threshold rises. However, even on the most repeated subsets, DP-trained models remain well below the corresponding non-private baselines.

E.1 WHAT OF DP-SGD CAUSES THE CAPABILITY GAP: CLIPPING VS. NOISING

DP-SGD performs per-example clipping followed by noise addition. We conduct an additional ablation to determine what causes the degradation in factual QA performance, finding that the culprit is noise. Specifically, Table 25 shows that adding noise calibrated to DP-SGD's signal-to-noise ratio to the unclipped batch gradient recovers similar accuracy degradation, while per-example clipping with no noise addition yields similar accuracy to non-private training.

Experimental details. In this section, we do not generate synthetic data and focus on QA accuracy after direct finetuning. Our experiments focus on Gemma 3 1B with LoRA on the Geminon dataset. We adopt as baselines the $\varepsilon = \{10, 100, \infty\}$ finetuning configurations from Appendix E (Table 23). We report test QA exact match accuracy.

Table 25: Ablating the noising and clipping of DP-SGD on Gemma 1B LoRA. We report Geminon test QA accuracy (exact match). "clip-only" yields comparable performance to non-private while "noise-only" performs similarly to DP.

Method	Clip	Noise	QA (%)
DP finetuning, $\varepsilon = 10$	✓	✓	0.7
+ (noise-only)	×	✓	0.5
+ (clip-only)	✓	×	80.4
DP finetuning, $\varepsilon = 100$	✓	✓	2.1
+ (noise-only)	×	✓	1.9
Non-private, $\varepsilon = \infty$	×	×	77.0

For the noise-only runs, we apply a heuristic to match SNR with the corresponding DP run by performing the same truncated poisson sampling and calibrated noise addition prescribed by the DP training recipe, but clipping the batch gradient to $1/\sqrt{B} \approx 0.442$ (where $B = 512$ is the expected batch size), instead of per-example clipping followed by rescaling to unit norm and summation.

This choice is motivated by the fact that our per-example gradients g_i are very high dimensional, and random vectors in high dimensions are likely to be orthogonal. Indeed for orthogonal unit vectors g_i, the pre-noise DP-SGD update $\|1/B \cdot \sum_{i=1}^{B} g_i\| = \sqrt{B}/B = 1/\sqrt{B}$. As pictured in Figure 10, indeed we observe that after the initial 500 steps, the pre-noise gradient in DP runs converges to around this amount.

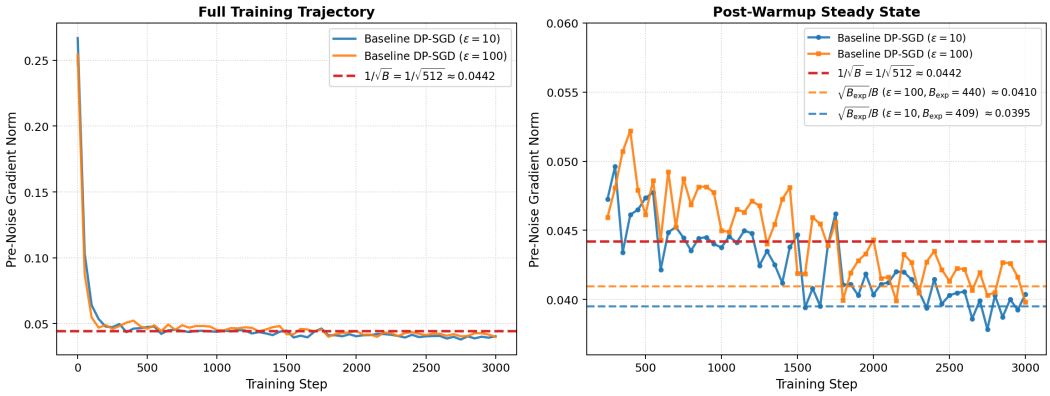

Figure 10: Pre-noise gradient norm across training for DP Gemma 1B LoRA runs on Geminon. *Left*: full run; *Right*: steps 250-3000. We plot the $1/\sqrt{B} \approx 0.0442$ heuristic, as well as $\sqrt{B_{exp}}/B$, where $B_{\text{exp}} < B$ is the actual expected batch size in truncated poisson sampling when targeting a physical batch size and denominator of $B = 512$.

F LEARNING DYNAMICS BY ATTRIBUTE, ENTITY, AND SUPPORT

F.1 GEMINON LEARNING DYNAMICS

We first examine training dynamics on GEMINON. Figure 11 tracks QA accuracy on both the high-repetition and singleton splits during evaluator training for 1B and 4B evaluators. High-repetition QAs, whose required facts appear in approximately 200 records, are learned rapidly when training on the real corpus or on non-private synthetic data. In contrast, DP-synthetic training improves only weakly and remains far below the non-private regimes. The singleton split remains near zero throughout training, while the high-repetition split improves sharply. This suggests that evaluators mainly learn facts supported by repeated evidence, rather than memorizing arbitrary facts that appear only once.

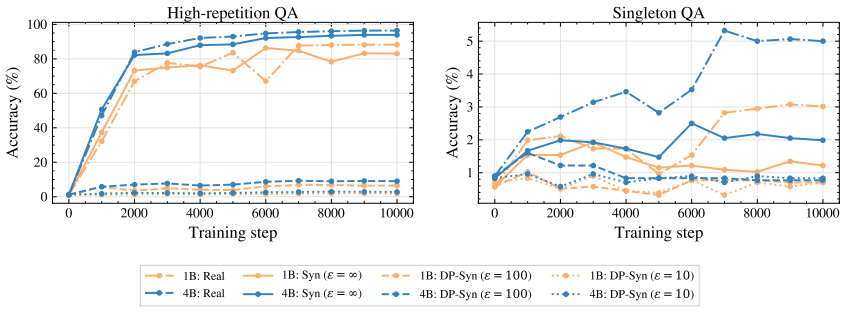

Figure 11: GEMINON QA accuracy over evaluator training steps. The left panel shows the high-repetition QA split, while the right panel shows the singleton split. Repeated facts are learned rapidly from the real corpus and non-private synthetic data, whereas DP-synthetic training remains much lower. Singleton facts remain low across regimes, consistent with the intended privacy sanity check.

To understand how factual learning happens, we further stratify accuracy by attribute type and by entity. Figure 12 shows a sharp contrast between non-private and DP training. Under real-corpus and non-private synthetic training, many attributes improve together on the high-repetition split, suggesting broad acquisition of the domain schema and the corresponding entity–attribute facts. Under DP-synthetic training, however, most attributes remain near zero, with only a small number of easier or more stereotyped attributes showing noticeable gains. In particular, `classification` remains partially learnable under DP, likely because it has a simpler answer structure and is easier to infer from surface patterns, while other attributes requiring precise entity-specific values remain difficult.

Figure 13 provides the complementary entity-level view. In the non-private regimes, different Geminon become answerable at different stages of training: the heatmap gradually fills in as the evaluator accumulates repeated facts for more entities. Together with the per-attribute curves, this suggests that learning is neither purely attribute-wise nor purely entity-wise. Instead, training gradually fills an entity–attribute fact matrix: the model learns the answer format and domain schema while progressively acquiring repeated facts about individual entities.

In contrast, the DP-synthetic regime remains sparse even on the high-repetition split, while the singleton region stays mostly dark. This suggests that the DP bottleneck is not merely slower convergence. Rather, DP-synthetic data appears to preserve some coarse schema or easy attribute cues, but fails to transmit the dense entity-specific factual matrix learned by non-private training.

F.2 NEWS LEARNING DYNAMICS

We next examine whether the same repetition-dependent pattern appears in NEWS, where support counts arise naturally from the number of articles covering an event. Figure 14 tracks NEWS QA accuracy over evaluator training after stratifying QAs by support-count bucket. Higher-support buckets improve earlier and reach higher final accuracy, especially under real-corpus and non-private syn-

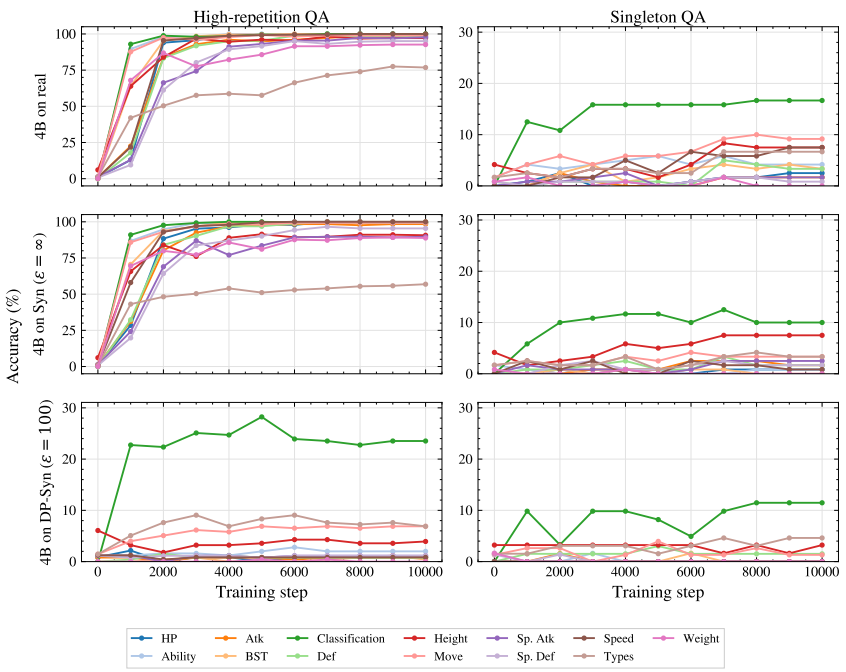

Figure 12: GEMINON QA accuracy by attribute type over evaluator training. We show 4B evaluators trained on the real corpus, non-private synthetic data, and DP-synthetic data at $\varepsilon = 100$; synthetic corpora are generated by the 4B generator. Many attributes improve together on the high-repetition split, suggesting broad acquisition of the domain schema and attribute-specific mappings. Accuracy remains much lower on the singleton split

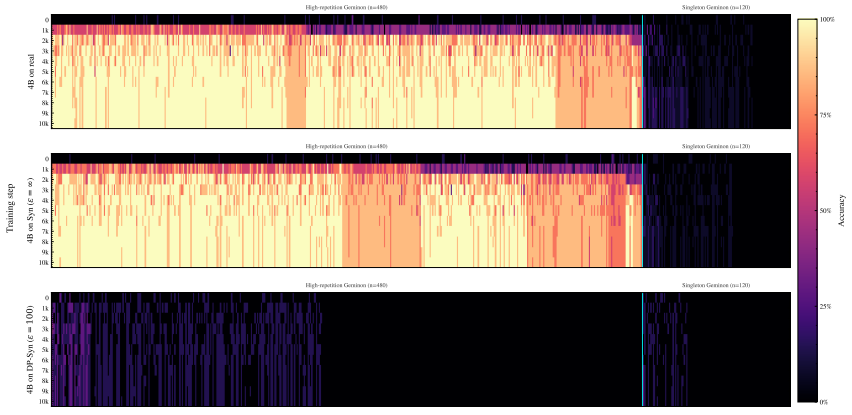

Figure 13: Per-entity GEMINON QA accuracy over evaluator training. Rows correspond to training checkpoints and columns correspond to Geminon entities. Entities are ordered by the first checkpoint at which their high-repetition QA accuracy exceeds a threshold. Different entities become answerable at different stages, indicating gradual entity-specific factual acquisition. Singleton facts remain mostly unrecovered.

thetic training. DP-synthetic training follows the same qualitative trend but remains substantially lower, showing that support helps DP learnability but does not eliminate the privacy-induced gap.

Overall, knowledge repetition is a strong predictor of factual transfer. Singleton facts remain unrecovered, high-repetition GEMINON facts are learnable, and naturally high-support NEWS facts are easier to transfer. However, repetition alone does not solve the DP synthesis problem: even for highly supported questions, DP-synthetic training remains substantially below non-private synthesis and real-corpus training.

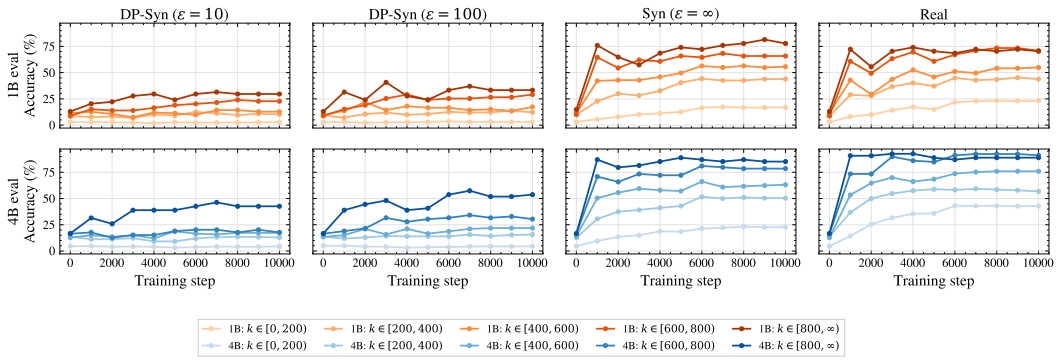

Figure 14: NEWS QA accuracy over evaluator training steps, stratified by support-count bucket. Columns correspond to training data sources: DP-synthetic data at $\varepsilon = 10$, DP-synthetic data at $\varepsilon = 100$, non-private synthetic data, and the real corpus. Rows correspond to evaluator size. Higher-support buckets are learned earlier and reach higher accuracy, especially under real-corpus and non-private synthetic training. DP-synthetic training benefits from higher support but remains substantially lower.

G SATURATION CASE STUDIES ON STANDARD DP BENCHMARKS

We expand on the saturation evidence shown in Figure 1 with detailed case studies on standard benchmarks used in the DP synthetic text literature: IMDB sentiment classification Kurakin et al. (2023), OpenReview score prediction Xie et al. (2024), and Yelp Polarity Zhang et al. (2015). Across all three datasets, the gap between no training and training on the real corpus is small, and the four training regimes (no training, DP-synth, non-private synth, train on real) occupy a narrow band that leaves little resolution to distinguish methods. CONTINUOUSBENCH GEMINON, by contrast, exhibits a much larger gap from no training (1.0%) to training on real (87.5%), providing substantially more headroom (Figure 15).

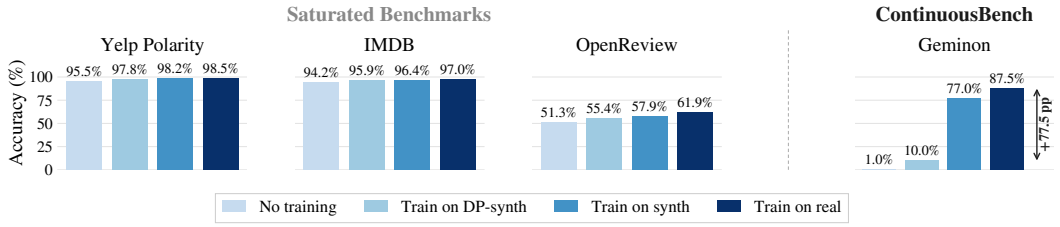

Figure 15: **Standard DP text synthesis benchmarks are saturated.** We compare accuracy across four training regimes (no training, DP-synth, non-private synth, and train on real) on three saturated benchmarks (Yelp Polarity, IMDB, OpenReview) and on CONTINUOUSBENCH GEMINON. On the standard benchmarks, the four regimes span only 3.0, 2.8, and 10.6 percentage points respectively, leaving DP-synth and non-private synth occupying a narrow band. On CONTINUOUSBENCH GEMINON, the corresponding range stretches from 1.0% to 87.5%, providing substantially more resolution for distinguishing DP synthesis methods.

G.1 IMDB SENTIMENT CLASSIFICATION

IMDB sentiment classification is a binary task (positive/negative) widely used in the DP benchmark literature Kurakin et al. (2023). IMDB is a particularly clear example of saturation: even zero-shot prompting achieves over 90% accuracy, leaving minimal headroom for any training-based method to demonstrate improvement.

Setup. We evaluate Gemma 3 1B-IT and 4B-IT on a test set of 5,000 IMDB reviews. We consider three prompt templates (*direct*, *reasoning*, and *structured*) in 0-shot and k-shot settings

($k \in \{3, 5, 7\}$, 3 random seeds each). We then fine-tune each model on the 40,000-example training split (1B: full fine-tuning, lr=2e-5; 4B: LoRA r=32, α=64, lr=2e-5; both for 3 epochs) and report validation and test accuracy.

Results. Table 26 summarizes the saturation gap. The 4B model achieves 94.2% accuracy under its best prompting condition (direct, 0-shot), while fine-tuning on the real training data improves this only to 97.0%. This gap of just 2.8 percentage points means that the prompted model already reaches 97% of the fine-tuned ceiling. Fine-tuning on DP-synthetic data (ε=10) achieves 95.9% and non-private synthetic data achieves 96.4%, confirming that all four training regimes occupy a narrow 2.8-point band from 94.2% to 97.0%, as visualized in Figure 15.

Table 26: Saturation summary on IMDB sentiment classification (%, exact-match). *Best prompted* is the maximum over all (template, shot, seed) conditions. The tiny gap between prompting and fine-tuning, especially for the 4B model, indicates virtually no headroom for distinguishing synthesis methods.

Model	Best Prompted	Fine-tuned	Gap (pts)	Prompted / FT
1B	89.7 (structured 0-shot)	94.8	5.1	0.95
4B	94.2 (direct 0-shot)	97.0	2.8	0.97

Detailed results. Tables 27–30 report the full breakdown. Table 27 shows accuracy across all prompting conditions; Table 28 reports per-seed variance for few-shot; Table 29 provides per-class accuracy; and Table 30 reports fine-tuning results.

Table 27: IMDB sentiment classification accuracy by prompting condition (%). Entries are exact-match / contains. Few-shot entries report mean across 3 seeds.

Task	# Test	Model	Template	0-shot	3-shot (mean)	5-shot (mean)	7-shot (mean)
Sentiment	5000	1B	direct	85.2 / 85.2	77.0 / 77.0	77.5 / 77.5	77.8 / 77.8
			reasoning	89.6 / 89.6	89.9 / 89.9	87.5 / 87.5	85.9 / 85.9
			structured	89.7 / 89.7	86.1 / 86.1	86.0 / 86.0	85.5 / 85.5
		4B	direct	94.2 / 94.2	85.7 / 85.7	90.0 / 90.0	90.5 / 90.5
			reasoning	93.7 / 93.7	93.1 / 93.1	92.6 / 92.6	92.5 / 92.5
			structured	93.4 / 93.4	92.8 / 92.8	92.0 / 92.0	92.3 / 92.3

G.2 OpenReview Score Prediction

OpenReview score prediction Xie et al. (2024) is a 5-way classification task over reviewer recommendation scores (1: strong reject, 3: reject, 5: marginally below threshold, 6: marginally above threshold, 8: accept), commonly used in the DP benchmark literature. Compared to Yelp Polarity and IMDB, OpenReview is somewhat less saturated, but the gap between prompting and fine-tuning remains small enough that the four training regimes still occupy a narrow band.

Setup. We evaluate Gemma 3 1B-IT and 4B-IT on a test set of 2,798 OpenReview papers Xie et al. (2024). We consider three prompt templates (*direct*, *reasoning*, and *structured*) in both 0-shot and 5-shot settings (3 random seeds for 5-shot). We then fine-tune each model on the training split (1B: full fine-tuning, lr=2e-5; 4B: LoRA r=32, α=64, lr=2e-4; both for 3 epochs) and report validation and test accuracy.

Results. Table 31 summarizes the saturation gap. The 4B model achieves 51.3% accuracy under its best prompting condition (reasoning, 0-shot), while fine-tuning on the real training data improves this only to 61.9%. This gap of just 10.6 percentage points means that the prompted model already reaches 83% of the fine-tuned ceiling. For the 1B model, the gap is larger (19.0 points) but the fine-tuned accuracy itself is modest at 58.5%.

Table 28: IMDB few-shot accuracy by seed (%, exact-match). Std reported across seeds.

Model	Template	k	Seed 0	Seed 1	Seed 2	Mean $\pm$ Std
1B	direct	3	79.2	79.7	72.2	77.0 $\pm$ 4.2
		5	83.8	78.2	70.4	77.5 $\pm$ 6.7
		7	78.3	88.2	66.8	77.8 $\pm$ 10.7
	reasoning	3	89.1	90.9	89.7	89.9 $\pm$ 0.9
		5	89.2	89.1	84.1	87.5 $\pm$ 2.9
		7	88.1	89.4	80.3	85.9 $\pm$ 4.9
	structured	3	84.3	85.6	88.3	86.1 $\pm$ 2.1
		5	86.8	85.3	85.9	86.0 $\pm$ 0.8
		7	82.0	89.2	85.2	85.5 $\pm$ 3.6
4B	direct	3	85.1	79.7	92.3	85.7 $\pm$ 6.3
		5	91.4	85.0	93.4	90.0 $\pm$ 4.4
		7	85.8	91.9	93.8	90.5 $\pm$ 4.1
	reasoning	3	93.3	93.3	92.7	93.1 $\pm$ 0.4
		5	91.4	92.4	94.2	92.6 $\pm$ 1.4
		7	90.1	93.1	94.2	92.5 $\pm$ 2.1
	structured	3	93.0	93.2	92.0	92.8 $\pm$ 0.6
		5	90.5	91.2	94.3	92.0 $\pm$ 2.0
		7	90.2	92.8	94.0	92.3 $\pm$ 1.9

Table 29: Per-class accuracy on IMDB sentiment classification (%), all prompting conditions combined (3 templates $\times$ {0-shot, 3/5/7-shot $\times$ 3 seeds} = 30 predictions per test example). Entries are exact-match / contains.

Sentiment	N	1B	4B
positive	75000	74.8 / 74.8	86.6 / 86.6
negative	75000	93.4 / 93.4	96.5 / 96.5
Macro avg		84.1 / 84.1	91.6 / 91.6

We additionally fine-tune the 4B model on DP-synthetic data (ε=10, temp=1.0) and non-private synthetic data (temp=1.0), both generated from a LoRA-finetuned Gemma 3 1B-PT generator, for 2 epochs with the same LoRA configuration. The DP-synthetic model achieves 55.4% and the non-private synthetic model achieves 57.9%, confirming that all four training regimes (no training, DP-synth, synth, real) occupy a narrow 10.6-point band from 51.3% to 61.9%, as visualized in Figure 15.

Table 31: Saturation summary on OpenReview score prediction (%, exact-match). *Best prompted* is the maximum over all (template, shot, seed) conditions. The small gap between prompting and fine-tuning, especially for the 4B model, indicates limited headroom for distinguishing synthesis methods.

Model	Best Prompted	Fine-tuned	Gap (pts)	Prompted / FT
1B	39.5 (structured 5-shot)	58.5	19.0	0.68
4B	51.3 (reasoning 0-shot)	61.9	10.6	0.83

Detailed results. Tables 32–35 report the full breakdown. Table 32 shows accuracy across all prompting conditions; Table 33 reports per-seed variance for 5-shot; Table 34 provides per-class accuracy; and Table 35 reports fine-tuning results.

Table 30: Fine-tuning results on IMDB sentiment classification (binary: positive/negative).

Model	Tuning	Train Data	Hyperparams	Val Acc (%)	Test Acc (%)	Train (s)
1B	Full FT	Real	lr=2e-5, bs=16×2	95.08	94.84	7313
4B	LoRA (r=32, α=64)	Real	lr=2e-5, bs=16×2	97.14	96.98	6072
4B	LoRA (r=32, α=64)	DP-synth (ε=10)	lr=2e-5, bs=16×2	95.82	95.94	5567
4B	LoRA (r=32, α=64)	Non-DP synth	lr=2e-5, bs=16×2	96.46	96.38	6292

Table 32: OpenReview score prediction accuracy by prompting condition (%). Entries are exact-match / contains. 5-shot entries report mean across 3 seeds.

Task	# Test	Model	Template	0-shot	5-shot (mean)
Score Prediction	2798	1B	direct	28.8 / 29.2	30.9 / 30.9
			reasoning	23.4 / 23.4	36.2 / 36.2
			structured	24.2 / 24.4	39.5 / 39.6
		4B	direct	47.3 / 47.3	50.9 / 50.9
			reasoning	51.3 / 56.8	50.9 / 50.9
			structured	51.1 / 51.1	50.4 / 50.4

G.3 Yelp Polarity

Yelp Polarity Zhang et al. (2015) is a binary sentiment classification task widely used in the DP benchmark literature. Headline accuracies across the four training regimes are summarized in Table 36. The full range from no training to training on the real corpus spans only 3.0 percentage points, with DP-synthetic and non-private synthetic data occupying a band of just 0.4 points between them. As with IMDB, this leaves essentially no resolution to distinguish DP synthesis methods.

H Geminon: Curation and Evaluation Details

Geminon is a fully controlled fictional domain designed to (i) make freshness unambiguous, (ii) support the systematic study of DP learnability, and (iii) reduce evaluation noise through grounded, short-answer QA with deterministic normalization. The full curation code is available at `https://anonymous.4open.science/r/ContinuousBenchCuration-3F88/`.

To construct a fictional world that is statistically realistic yet entirely novel, we derive structural priors from publicly available Pokémon metadata maca11 (2021); Amancio (2024). These priors are used only to parameterize the generative process. We emphasize that all resulting Geminon entities, names, and stats are newly created, while types, moves, and abilities are inherited from the reference data.

H.1 Index Data Curation

H.1.1 Evolution Line and Types

In the original Pokémon data, there are 160 three-stage evolution lines, 239 two-stage evolution lines, and 120 single-stage lines. We generate 100 three-stage evolution lines (300 Geminon), 100 two-stage evolution lines (200 Geminon), and 100 single-stage lines (100 Geminon), for a total of 600 entities. All randomness is seeded for reproducibility.

Each evolution line is assigned a primary type (`type1`) sampled uniformly from the 18 canonical Pokémon types maca11 (2021); Amancio (2024). This primary type is held constant across all stages of the line. A secondary type (`type2`) is then assigned independently at each stage according to a two-case rule. If the preceding stage has no `type2`, a new secondary type is introduced with stage-dependent probability: for a three-stage line, the probabilities are $[0.2,\ 0.4,\ 0.8]$ across stages 1–3; for a two-stage line, $[0.2,\ 0.8]$; and for a single-stage line, $[0.5]$. If the preceding stage already has a

Table 33: OpenReview 5-shot accuracy by seed (%). Entries are exact-match / contains. Std reported across seeds.

Model	Template	Seed 0	Seed 1	Seed 2	Mean ± Std
1B	direct	26.7 / 26.7	33.0 / 33.0	32.9 / 32.9	30.9 ± 2.9 / 30.9 ± 2.9
	reasoning	32.7 / 32.8	42.3 / 42.3	33.5 / 33.5	36.2 ± 4.3 / 36.2 ± 4.3
	structured	39.6 / 39.6	38.7 / 38.8	40.2 / 40.2	39.5 ± 0.6 / 39.6 ± 0.6
4B	direct	51.8 / 51.8	47.3 / 47.3	53.7 / 53.7	50.9 ± 2.7 / 50.9 ± 2.7
	reasoning	52.4 / 52.4	48.3 / 48.4	52.0 / 52.0	50.9 ± 1.8 / 50.9 ± 1.8
	structured	52.6 / 52.6	47.0 / 47.0	51.6 / 51.6	50.4 ± 2.4 / 50.4 ± 2.4

Table 34: Per-class accuracy on OpenReview score prediction (%), all prompting conditions combined (3 templates × {0-shot, 5-shot × 3 seeds} = 12 predictions per test example). Entries are exact-match / contains.

Recommendation	Description	N	1B	4B
1	strong reject	204	2.9 / 2.9	20.1 / 20.6
3	reject, not good enough	6648	38.9 / 38.9	51.2 / 52.0
5	marginally below threshold	9120	11.2 / 11.3	68.6 / 68.6
6	marginally above threshold	10752	46.0 / 46.1	43.9 / 44.5
8	accept, good paper	6852	36.9 / 36.9	37.2 / 37.7
Macro avg (unweighted)			27.2 / 27.2	44.2 / 44.7

`type2`, it is retained with probability 0.8 and replaced by a different secondary type with probability 0.2. This rule is applied uniformly across all line lengths and makes secondary typing more likely in later-stage, typically stronger forms while still permitting occasional type diversification.

H.1.2 STATISTICS

Reference Stats. For each of the 998 reference Pokémon maca11 (2021); Amancio (2024), we record nine attributes: six battle stats (`HP`, `Attack`, `Defense`, `Special Attack`, `Special Defense`, and `Speed`), `Base Stat Total` (`BST`), `height`, and `weight`. For each attribute a, we compute the empirical mean μ_a and standard deviation σ_a over all stage-1 reference Pokémon.

For multi-stage evolution lines, we also compute per-attribute multiplicative ratios: *stage-2/stage-1* from 359 pairs and *stage-3/stage-1* from 120 pairs. For each attribute a and each ratio type, we record the empirical mean μ_a^r and standard deviation σ_a^r for $r \in \{2/1, 3/1\}$.

Stage-1 Statistics Sampling. For each attribute a, we construct a discrete grid $\mathcal{G}_a$ of integer-valued points spanning $[\min_a, \max_a)$ with step size 1, where $\min_a$ and $\max_a$ are taken over all real stage-1 Pokémon. Battle stats are sampled from this grid using a *discrete Gaussian*(μ_a, σ_a): each grid point $x \in \mathcal{G}_a$ is sampled with probability $\propto \exp(-(x-\mu_a)^2/2\sigma_a^2)$, where μ_a and σ_a are estimated from stage-1 reference Pokémon. `Height` and `weight` are sampled using a *discrete exponential*$(1/\mu_a)$, where each grid point is sampled with probability $\propto \exp(-x/\mu_a)$. The stage-1 `BST` is then computed as the *sum of the six sampled battle stats.*

Later-stage Statistics Sampling. For each attribute a and ratio type $r \in \{2/1,\ 3/1\}$, we construct a *restricted ratio grid* $\mathcal{G}_a^r$ to confine samples to a tight, empirically grounded range. Specifically, the empirical standard deviation is clipped to $\bar{\sigma}_a = \mathrm{clip}(\sigma_a^r,\ 0.05,\ 0.1)$ and the empirical mean is clipped to $\bar{\mu}_a = \mathrm{clip}(\mu_a^r,\ 0.05,\ c_{\max})$, where $c_{\max} = 1.6$ for the 2/1 ratio and $c_{\max} = 2.0$ for the 3/1 ratio. The restricted grid is $[\bar{\mu}_a - \bar{\sigma}_a, \bar{\mu}_a + \bar{\sigma}_a]$ with step size 0.1, with boundaries rounded outward to the nearest 0.01.

Table 35: Fine-tuning results on OpenReview score prediction (5-way classification over recommendation scores 1, 3, 5, 6, 8). Real-data models trained for 3 epochs; synthetic-data models trained for 2 epochs.

Model	Tuning	Train Data	Hyperparams	Val Acc (%)	Test Acc (%)	Train (s)
1B	Full FT	Real	lr=2e-5, bs=16×2	58.97	58.47	1899
4B	LoRA (r=32, α=64)	Real	lr=2e-4, bs=8×4	61.12	61.94	4499
4B	LoRA (r=32, α=64)	DP-synth (ε=10)	lr=2e-5, bs=16×2	58.29	55.43	2966
4B	LoRA (r=32, α=64)	Non-DP synth	lr=2e-5, bs=16×2	59.76	57.86	3115

Table 36: Yelp Polarity accuracy (%) across the four training regimes shown in Figure 15.

No training	DP-synth	Non-private synth	Train on real
95.5	97.8	98.2	98.5

Stage-2 and stage-3 attribute values are computed as

$$s_a^{(2)} = \mathrm{round}\Big(s_a^{(1)} \times r_a\Big), \quad r_a \sim \mathrm{Uniform}(\mathcal{G}_a^{2/1}),$$
$$s_a^{(3)} = \mathrm{round}\Big(s_a^{(1)} \times r_a\Big), \quad r_a \sim \mathrm{Uniform}(\mathcal{G}_a^{3/1}),$$

where $\mathrm{Uniform}(\mathcal{G})$ denotes a uniform draw from the discrete grid $\mathcal{G}$. Notably, stage 3 is derived *directly from stage 1*, rather than from stage 2, using the stage-3/stage-1 ratio grid. This avoids compounding variance across stages. The multiplicative scheme is applied to all six battle stats as well as to `height` and `weight`. For each later stage, `BST` is recomputed as the sum of the six battle stats.

H.1.3 Moves and Abilities

We additionally load reference tables of moves and abilities to define realistic candidate pools maca11 (2021); Amancio (2024). Move records contain a `name`, `type`, and short description. Ability and move names are deduplicated and filtered to concise surface forms of at most two words. These pools are used only to sample type-consistent attributes for Geminon.

For each Geminon, we sample one ability uniformly from the reference ability pool and one signature move, together with its description, uniformly from the set of moves matching either of its types maca11 (2021); Amancio (2024).

H.1.4 Indexing and Schema

After generation, evolution lines are shuffled and assigned contiguous integer id starting at 10000, ensuring that members of the same line occupy adjacent indices. Each Geminon record contains:

Geminon Index Schema

```
{
  "name": <string>,
  "classification": <string>,
  "type1": <string>,
  "type2": <string or null>,
  "ability": <string>,
  "hp": <int>, "attack": <int>, "defense": <int>,
  "special attack": <int>, "special defense": <int>,
  "speed": <int>, "base_stat_total": <int>,
  "weight": <int>, "height": <int>,
  "evolution_line": [<string>, ...],
  "move": {"name": <string>,
           "short_description": <string>},
  "idx": <int>
}
```

Names and classifications are generated subsequently using `gemini-2.5-flash` Gemini Team (2025). A deduplication of names and classifications is conducted. Specifically, we require (1) all 600 individual Geminon names must be globally unique, and (2) for evolution lines with more than two stages, no two lines may share the same tuple of classifications across their stages. Single-stage lines are exempt from (2). Re-queries to `gemini-2.5-flash` are issued with a forbidden-names or forbidden-tuple constraint until both conditions are met.

Examples and Distribution Similarity We verify that the generated stage-conditional distributions closely match the corresponding reference Pokémon statistics, including the skewness of physical attributes and the monotonic increase of battle stats across evolutionary stages. A comparison of the resulting distributions is shown in Figure 17, and example GEMINON index entries are shown in Figure 16.

Singleton and High-repetition Split We randomly sample 20 evolution lines from each of the three categories, three-stage, two-stage, and single-stage lines, to form the singleton split. This yields a total of 120 singleton Geminon. The remaining 180 evolution lines, comprising 480 Geminon, constitute the high-repetition split.

H.2 CORPUS CURATION

Given the GEMINON index, we construct a corpus as a mixture of in-world document formats designed to distribute each atomic fact (the stats and attributes of Geminon) across many records while avoiding one-to-one paraphrasing. Specifically, we synthesize four document types: *wiki entries* (encyclopedic third-person descriptions), *field journal notes* (informal first-person trainer observations), *evolution logs* (narrative analyses of an entire evolution line), and *pairwise comparisons* (side-by-side trainer assessments of two Geminon).

Each prompt instructs the LLM to return, alongside every article, a provenance list specifying the exact Geminon attribute keys referenced in that article. These lists serve as a proxy for attribute-level mention frequency. LLM outputs are parsed by stripping Markdown fences and decoding the resulting JSON list. The raw provenance strings produced by the LLM are then normalized to canonical feature names through a combination of prefix stripping, an explicit mapping covering approximately 70 known variants and typos, and a final underscore-to-space fallback. Strings that cannot be resolved are discarded.

Articles are generated using `gemini-2.5-pro` with the prompts shown below. For the wiki, journal, and evolution prompts, each prompt is queried 100 times, and each response yields 10 articles, producing approximately 1000 candidate articles per Geminon or evolution line. Comparison prompts are run once over all $\binom{480}{2} = 114960$ unique pairs of public Geminon, with each response yielding 3 articles. Evolution prompts are applied to all 160 multi-stage public evolution lines (80 three-stage and 80 two-stage). After parsing, the final corpus contains 1408369 articles on Geminons in the high-repetition split and 116739 articles on Geminons in the singleton split, with the following breakdown:

Table 37: Breakdown stats for Geminon corpus.

Document type	Articles
Pairwise comparison (high-repetition)	341,490
Field journal (high-repetition)	452,188
Evolution log (high-repetition)	147,271
Wiki (high-repetition)	467,420
Wiki (singleton)	116,739
Total on high-repetition split	**1,408,369**
Total on singleton split	**116,739**

Example articles are shown in Figure 18.

Single-stage evolution example

Stage 1: Caelumin


```
{
  "name": "Caelumin",
  "classification": "Beacon Geminon",
  "type1": "flying",
  "type2": null,
  "ability": "Illuminate",
  "hp": 53,
  "attack": 65,
  "defense": 58,
  "special attack": 77,
  "special defense": 99,
  "speed": 49,
  "base_stat_total": 401,
  "weight": 867,
  "height": 9,
  "idx": 10002,
  "evolution_line": ["Caelumin"],
  "move": {
    "name": "Sky Attack",
    "short_description": "User charges for one turn before attacking."
  }
}
```


Two-stage evolution example

Stage 1: Phantatch


```
{
  "name": "Phantatch",
  "classification": "Spore Geminon",
  "type1": "ghost",
  "type2": null,
  "ability": "Suction Cups",
  "hp": 79,
  "attack": 48,
  "defense": 56,
  "special attack": 23,
  "special defense": 88,
  "speed": 84,
  "base_stat_total": 378,
  "weight": 198,
  "height": 14,
  "idx": 10000,
  "evolution_line": ["Phantatch", "
Phantrain"],
  "move": {
    "name": "Poltergeist",
    "short_description": "Inflicts regular
damage with no additional effect."
  }
}
```


Stage 2: Phantrain


```
{
  "name": "Phantrain",
  "classification": "Parasite Geminon",
  "type1": "ghost",
  "type2": "dark",
  "ability": "Drought",
  "hp": 115,
  "attack": 72,
  "defense": 87,
  "special attack": 39,
  "special defense": 147,
  "speed": 134,
  "base_stat_total": 594,
  "weight": 337,
  "height": 22,
  "idx": 10001,
  "evolution_line": ["Phantatch", "
Phantrain"],
  "move": {
    "name": "Jaw Lock",
    "short_description": "Seeds the target
after inflicting damage."
  }
}
```


Three-stage evolution example

Stage 1: Boreling


```
{
  "name": "Boreling",
  "classification": "Frost
Geminon",
  "type1": "ice",
  "type2": null,
  "ability": "Berserk",
  "hp": 69,
  "attack": 60,
  "defense": 63,
  "special attack": 67,
  "special defense": 68,
  "speed": 40,
  "base_stat_total": 367,
  "weight": 52,
  "height": 12,
  "idx": 10003,
  "evolution_line": ["
Boreling", "Borelash", "
Borastat"],
  "move": {
    "name": "Powder Snow",
    "short_description": "
Has a chance to freeze the
target."
  }
}
```


Stage 2: Borelash


```
{
  "name": "Borelash",
  "classification": "Shard
Geminon",
  "type1": "ice",
  "type2": null,
  "ability": "Super Luck",
  "hp": 115,
  "attack": 89,
  "defense": 91,
  "special attack": 100,
  "special defense": 100,
  "speed": 64,
  "base_stat_total": 559,
  "weight": 83,
  "height": 20,
  "idx": 10004,
  "evolution_line": ["
Boreling", "Borelash", "
Borastat"],
  "move": {
    "name": "Ice Burn",
    "short_description": "
Requires a turn to charge
before attacking."
  }
}
```


Stage 3: Borastat


```
{
  "name": "Borastat",
  "classification": "Aurora
 Geminon",
  "type1": "ice",
  "type2": "electric",
  "ability": "Sweet Veil",
  "hp": 140,
  "attack": 125,
  "defense": 122,
  "special attack": 141,
  "special defense": 129,
  "speed": 71,
  "base_stat_total": 728,
  "weight": 104,
  "height": 24,
  "idx": 10005,
  "evolution_line": ["
Boreling", "Borelash", "
Borastat"],
  "move": {
    "name": "Icy Wind",
    "short_description": "
Has a chance to lower the
target's Speed."
  }
}
```


Figure 16: Example GEMINON index entries.

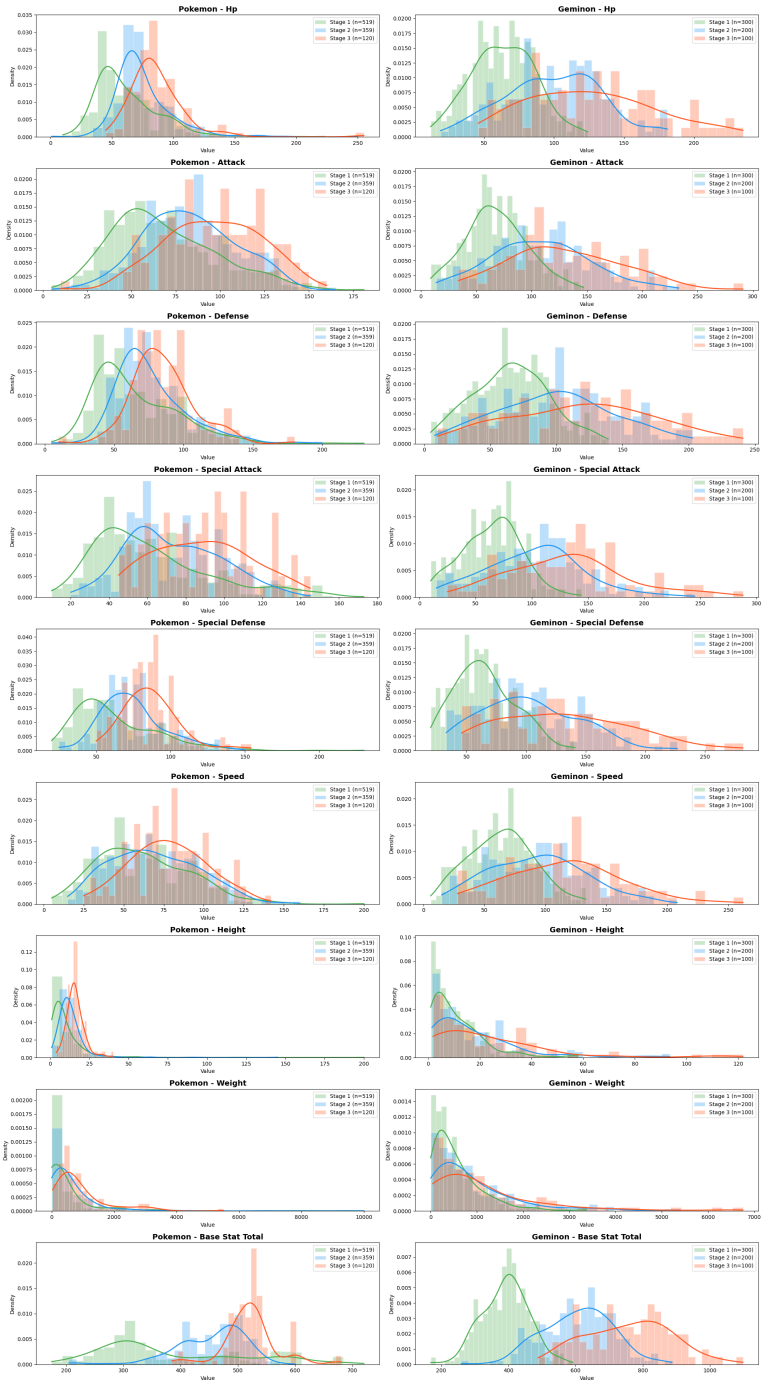

Figure 17: Comparison of Pokémon and Geminon stat distributions.

wiki


```
{
  "text": "As the first stage in the
evolutionary line that culminates in
Borastat, Boreling is a quintessential
Frost Geminon. This pure ice-type, known as
 Gemidex entry #10003, measures 12 meters
in height and weighs 52 lbs.  Its passive
ability is Berserk, and it can be taught
the move Powder Snow, which has a
percentage-based chance to freeze an
opponent. Boreling's base stat total of 367
 is comprised of 69 HP, 60 attack, 63
defense, 67 special attack, 68 special
defense, and a speed rating of 40.",
  "tag": [
    {
      "idx": 10003,
      "info": ["idx", "name", "
classification", "type1", "ability", "
evolution_line", "height", "weight", "
base_stat_total", "hp", "attack", "defense
", "special attack", "special defense", "
speed", "move.name", "move.
short_description"]
    }
  ],
  "type": "wiki"
}
```


journal


```
{
  "text": "I need to be strategic with my
new Boreling. Its Berserk ability only
activates when its HP is low, which means I
 need to know it can survive a hit. Its 63
defense and 68 special defense give me some
 confidence there. Since its speed is a low
 40, I'll probably have to take a hit
before I can attack, so those defensive
stats are crucial.",
  "tag": [
    {
      "idx": 10003,
      "info": ["name", "ability", "hp", "
defense", "special defense", "speed"]
    }
  ],
  "type": "journal"
}
```


comparison


```
{
  "text": "Looking at their combat stats,
Aerofir is clearly built for speed,
boasting a speed stat of 95, while Boreling
 is sluggish at 40. Despite this speed gap,
 their core defensive and offensive stats
are surprisingly close, with both having
attack and defense in the 60s. Aerofir has
a better HP pool at 81, making it a bit
more durable than Boreling at 69 HP.
Overall, Aerofir has the advantage with a
higher base stat total, but it's not a
total blowout in every category.",
  "tag": [
    {
      "idx": 10003,
      "info": ["hp", "attack", "defense", "
speed", "base_stat_total"]
    },
    {
      "idx": 10503,
      "info": ["hp", "attack", "defense", "
speed", "base_stat_total"]
    }
  ],
  "type": "comparison"
}
```


evolution log


```
{
  "text": "While its special attack becomes
 its greatest asset, the attack stat also
sees consistent growth. Boreling's initial
attack of 60 is a solid starting point.
This grows to a more threatening 89 in its
Borelash stage. As Borastat, its physical
prowess culminates in an impressive 125
attack, making it a versatile threat that
can strike effectively from both the
physical and special sides.",
  "tag": [
    {
      "idx": 10003,
      "info": ["attack"]
    },
    {
      "idx": 10004,
      "info": ["attack"]
    },
    {
      "idx": 10005,
      "info": ["attack"]
    }
  ],
  "type": "evolution"
}
```


Figure 18: Example articles about Geminon Boreling.

H.3 QA CURATION.

For each Geminon in the high-repetition and singleton splits, we construct simple questions about its features using the following template:

Geminon Question Template

```
"What is the classification of {name}?"
"What are the types of {name}?"
"What is the ability of {name}?"
"What is the HP stat of {name}?"
"What is the attack stat of {name}?"
"What is the defense stat of {name}?"
"What is the special attack stat of {name}?"
"What is the special defense stat of {name}?"
"What is the speed stat of {name}?"
"What is the base stat total stat of {name}?"
"What is the move of {name}?"
"What is the weight (in lbs) of {name}?"
"What is the height (in meters) of {name}?"
```

Example QAs for Geminon Boreling is presented below.

Example QAs for Geminon Boreling

```
{
    "question": "What is the classification of Boreling?",
    "answer": "Frost Geminon"
    "supports": [510, 3000, 5960, 7410, 8921, 8929, ...] # supporting articles idxs
}
{
    "question": "What are the types of Boreling?",
    "answer": "ice"
    "supports": [3001, 7411, 8922, 8929, 10425, 11915, ...] # supporting articles idxs
}
...
```

The two splits' QAs are kept separate. Within each split, we partition every Geminon's QAs into validation and test sets of roughly equal size: the validation set is used for checkpoint selection and the test set for final performance reporting.

H.4 SUBSAMPLING

From the full corpus, denoted GEMINON-LARGE, we construct two smaller subsets, GEMINON-SMALL and GEMINON-MEDIUM, via a coverage-aware weighted sampling procedure.

For the high-repetition split, in each round, each candidate article is assigned a score of $\sum_{(\text{Geminon, feature})} 1/(1 + \texttt{count})$, where `count` tracks how many times that $(\text{Geminon}, \text{feature})$ pair has already been selected. Articles that fill the largest current coverage gaps therefore receive the highest sampling probability. The resulting source breakdown is as follows:

- GEMINON-SMALL: comparison 100k, evolution 40k, journal 40k, wiki 20k
- GEMINON-MEDIUM: comparison 300k, evolution 100k, journal 300k, wiki 300k

For the singleton set, we select one wiki article per Geminon in the singleton split by greedily choosing the candidate with the highest coverage of that Geminon's applicable canonical features. Under this criterion, all 120 singleton Geminon achieve perfect feature coverage.

Appending these 120 singleton wiki articles to each subset yields final sizes of 200,120 and 1,000,120, respectively. Table 38 reports attribute-mention statistics over the full corpus (GEMINON-LARGE).

The QA pairs are identical across all three corpus sizes, with different support sets, since each subset contains a different pool of articles. On average, a QA has roughly 200 supporting articles in GEMINON-SMALL and roughly 1,100 in GEMINON-MEDIUM.

Table 38: Left: per-Geminon mention counts for each feature, aggregated over all articles for the 480 Geminons in the high-repetition split; Mean/Std/Min/Max are computed over per-Geminon counts. Right: feature coverage in the 120 selected wiki articles, each corresponding a Geminon in the singleton split. `type2` covers only 80/120 cases because the remaining 40 Geminon are single-typed.

Feature	Mean	Std	Min	Max
name	2260.7	146.8	1778	2717
classification	1721.2	67.1	1523	1908
type1	1877.8	81.9	1665	2133
type2	1298.3	630.2	390	2190
ability	1763.7	90.0	1550	2205
hp	1586.0	160.0	1139	2272
attack	1716.1	267.7	1002	2374
defense	1707.4	212.8	1216	2380
speed	1758.8	132.9	1448	2263
special attack	1543.5	292.6	1014	2268
special defense	1512.4	220.3	1051	2161
base_stat_total	1727.2	77.6	1480	1880
weight	1649.6	47.3	1467	2049
height	1641.2	42.4	1462	1721
move.name	917.1	103.6	527	1178

Feature	Coverage
name	120/120
classification	120/120
type1	120/120
type2	80/120
ability	120/120
hp	120/120
attack	120/120
defense	120/120
special attack	120/120
special defense	120/120
speed	120/120
base_stat_total	120/120
weight	120/120
height	120/120
move.name	120/120

I NEWS: CURATION AND EVALUATION DETAILS

I.1 CORPUS CURATION

We leveraged the CommonCrawl News Common Crawl (2026), specifically the September 2025 dump for the experiments presented. We only select the English documents and got 2,876,319 documents in total. For each article, we retrieved the publication date, document URL, main text, language, crawl date, hostname, and title using `trafilatura`. Code for dataset curation can be found at `https://anonymous.4open.science/r/ContinuousBenchCuration-3F88/`.

We normalize the paragraphs by collapsing extra spaces and newlines, producing 1,768,567 cleaned articles. Then, we deduplicate the cleaned-up articles in two passes. Pass 1 removes exact duplicates globally via SHA-256 hashing of article text. Pass 2 performs near-deduplication using MinHash LSH (`datasketch`) with 128 permutations, word 5-gram shingles, and a containment similarity threshold of 0.80, where containment $= |A \cap B| / \min(|A|, |B|)$. When a near-duplicate pair is found, the shorter article is removed. Some example articles are presented below.

I.2 QA CURATION

We construct question-answer pairs from the deduplicated cleaned-up corpus through a multi-stage pipeline using the Gemini API.

Clustering. We embedded the extracted articles using `embeddinggemma-300m` Team (2025), with prompt prefix `"task: clustering | query: "`, input mode title and first paragraph (∼500 characters), maximum sequence length 512, and L2-normalized `float16` outputs. We then clustered the 1,768,567 embeddings with a local windowed kNN + Leiden approach: for each 7-day sliding window (±3 days), we build a mutual kNN graph (k-search=60, k-graph=30, similarity threshold 0.55) and run Leiden community detection (resolution 0.07, 4 iterations), with a minimum cluster size of 25 and cross-window merging (Jaccard threshold 0.35, centroid similarity 0.9). This step produced 13,870 clusters containing 465,966 unique articles.

Fact extraction. For each of the top 500 largest clusters, we send up to 50 articles (each truncated to 512 characters) to `gemini-2.5-pro` and extract 12–25 short, QA-ready facts. Each fact must be explicitly stated in at least three distinct articles and anchored by an absolute date, a named entity, or a named document or action. This yielded 5,947 facts across 431 clusters. It is worth noting that the 69 clusters left are mostly spam clusters.

Example article 1 about Asia Cup 2025: India vs Pakistan


```
{
    "url": "https://newsable.asianetnews.com/sports/cricket-india-vs-pakistan-asia-cup
-2025-referee-pycroft-pcb-handshake-row-articleshow-miqavqv",
    "hostname": "newsable.asianetnews.com",
    "title": "India vs Pakistan, Asia Cup 2025: Referee Pycroft Faces PCB Heat As Handshake
 Controversy Escalates",
    "date": "2025-09-15",
    "crawl_date": "2025-09-15T05:55:11Z",
    "language": "en",
    "text": "India\u2019s win over Pakistan in the Asia Cup was overshadowed by a handshake
 row. Match referee Andy Pycroft faces heat as PCB protests India\u2019s refusal to shake
hands, citing government policy and solidarity with Pahalgam attack victims. Dubai [UAE]:
The highly anticipated India-Pakistan Asia Cup clash in Dubai on Sunday has spiraled into
controversy, with match referee Andy Pycroft caught in the middle of a row over the absence
 of customary handshakes. India clinched a comfortable seven-wicket win, but it was events
off the field that have dominated headlines..."
}
```


Example article 2 about Asia Cup 2025: India vs Pakistan


```
{
    "url": "https://www.hindustantimes.com/cricket/pakistan-knock-on-iccs-door-demands-
immediate-removal-of-match-referee-for-staying-silent-on-indias-no-handshake
-101757928011929.html",
    "hostname": "www.hindustantimes.com",
    "title": "Pakistan knocks on ICC's door, demands immediate removal of match referee for
 staying silent on India's no handshake",
    "date": "2025-09-15",
    "crawl_date": "2025-09-15T09:31:13Z",
    "language": "en",
    "text": "Pakistan knocks on ICC's door, demands immediate removal of match referee for
staying silent on India's no handshake The PCB has escalated India's no-handshake
controversy to the ICC, asking for the immediate removal of match referee Andy Pycroft. The
 no-handshake episode between India and Pakistan at the end of Match 6 of the Asia Cup 2025
 is snowballing into a huge controversy. After the Pakistan cricket team complained to the
match referee about India's actions, the board has now reached out to the ICC, demanding
that the official be removed. PCB's strict measures come in the wake of the disappointment
stemming from the fact that Indian players did not participate in the post-match customary
handshakes with their Pakistan counterparts. The move has triggered former Pakistan
cricketers and members of the board alike, and facing the wrath is Andy Pycroft, a veteran
in this role..."
}
```


Figure 19: Example articles about Asia Cup 2025: India vs Pakistan

QA generation. We then send the extracted facts, together with up to 50 underlying cluster articles (each truncated to 512 characters), back to `gemini-2.5-pro` to generate up to 12 short-answer QA pairs per cluster. Questions must be answerable with a name, date, number with unit, location, or outcome, and should include an absolute timestamp to make them unambiguous. We require that the same answer string appear verbatim in at least three articles. This process yields 4,468 QA pairs across 430 clusters.

In manual inspection, this two-stage pipeline produced substantially higher-quality QA pairs than direct single-stage question generation.

Closed-book Answerability and Ambiguity check. Next, we prompt `gemini-2.5-pro` with each question in a zero-shot setting, i.e., without any article context. Then use the same model as a judge to determine 1. whether the zero-shot response matches the ground-truth answer and 2. whether each question is fully interpretable on its own (standalone-ambiguity check.) We emphasize that, although both the source articles and the resulting knowledge tests are post-cutoff, models may still answer some questions correctly by exploiting prior associations or plausible inference rather than genuine knowledge of the underlying articles. For example, for a question such as "Which company hosted a party on September 15, 2025, in celebration of the 2025 Emmy Awards?", a model may correctly guess "Netflix" based on historical associations rather than direct knowledge of

the covered event. This yields 2,604 out of 4,468 QAs that are both unambiguous and unanswerable (39.7% zero-shot correct and 3.0% ambiguous).

Open-book Answerability check (Support identification.) We embed every article and every surviving QA, and for each QA retrieve its top-1000 candidate articles by embedding similarity. Grouping these candidates by article, we prompt `gemini-2.5-flash-lite` in open-book fashion, i.e. prompt with the article in context, and judge each response for correctness with `gemini-2.5-flash-lite`. An article is counted as *supporting* a QA if its open-book answer is judged correct.

Each QA is provided with its closed-book answer and the corresponding LLM-judge correctness label, together with open-book answers for its ≈1,000 candidate articles, each carrying its own LLM-judge correctness label. By construction, every closed-book answer is judged incorrect, due to the filtering criterion, but we retain these (incorrect) answers for completeness, in case they are of interest. For convenience, we additionally report a *support count* per QA: the number of candidate articles whose open-book answer is judged correct. Some examples are presented below.

Example Question 1 about Asia Cup 2025: India vs Pakistan


```
{
    "question": "Who was the match referee
for the India vs. Pakistan Asia Cup match
on September 14, 2025?",
    "answer": "Andy Pycroft",
    "support_count": 648,
    "closedbook_gemini-2.5-pro": {
        "answer": "Javagal Srinath",
        "is_correct": false
    },
    "openbook_gemini-2.5-flash-lite": [
        {
            "article_id": 126152,
            "answer": "Not mentioned",
            "is_correct": false
        },
        ...
        {
            "article_id": 1713242,
            "answer": "Not mentioned",
            "is_correct": false
        }
    ]
}
```


Example Question 2 about Asia Cup 2025: India vs Pakistan


```
{
    "question": "Who was the chairman of
the Pakistan Cricket Board as of September
2025?",
    "answer": "Mohsin Naqvi",
    "support_count": 463,
    "closedbook_gemini-2.5-pro": {
        "answer": "Ramiz Raja",
        "is_correct": false
    }
    "openbook_gemini-2.5-flash-lite": [
        {
            "article_id": 547,
            "answer": "Mohsin Naqvi",
            "is_correct": true
        },
        ...
        {
            "article_id": 1376988,
            "answer": "Unknown",
            "is_correct": false
        }
    ]
}
```


Figure 20: Example QAs about Asia Cup 2025: India vs Pakistan

Within each cluster, we split the QAs roughly evenly into validation and test, then pool across clusters to form the final sets: validation is used for checkpoint selection, and test for final performance reporting.

I.3 SUBSAMPLING

The full set of articles constitutes NEWS-LARGE, with roughly 1.7M articles. We then form NEWS-SMALL (≈200K articles) as the union of every released QA's support set. Additionally, we construct NEWS-MEDIUM (≈580K articles) as the union of (i) every article assigned to a cluster during the clustering stage and (ii) all of NEWS-SMALL. Since NEWS-SMALL consists exactly of the supporting articles and NEWS-SMALL $\subseteq$ NEWS-MEDIUM $\subseteq$ NEWS-LARGE, both the QAs and their full support sets are identical across all three sizes. The support counts of the QAs are presented below.

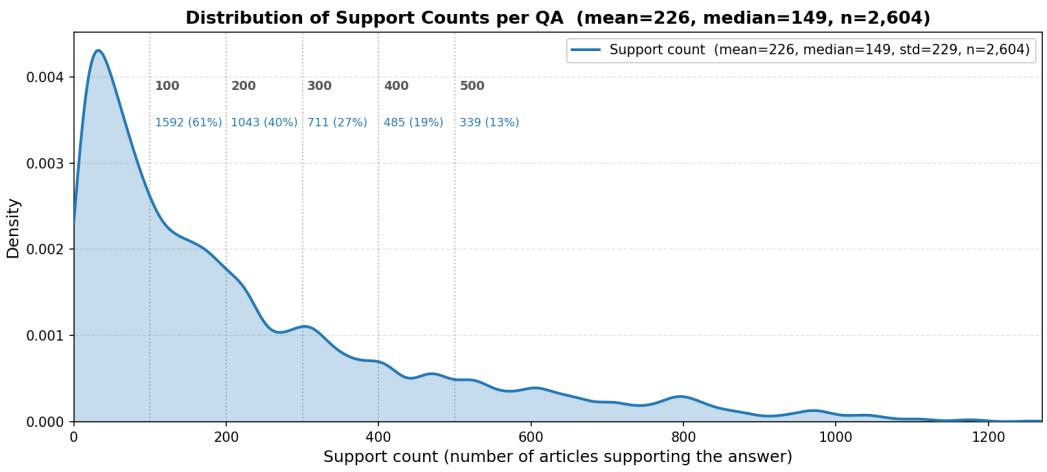

Figure 21: Support count distribution of the news QAs. The blue curve shows the per-article distribution and vertical lines mark support $\geq$ 100, 200, 300, 400, 500 articles, with the corresponding cumulative percentages shown in blue.