HUBBLE: A MODEL SUITE TO ADVANCE THE STUDY OF LLM MEMORIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We present HUBBLE, a suite of open-source large language models (LLMs) for the scientific study of LLM memorization. HUBBLE models come as minimal pairs: standard models are pretrained on a large English corpus, and perturbed models are trained in the same way but with controlled insertion of text (e.g., book passages, biographies, and test sets) designed to emulate key memorization risks. Our core release includes 8 models—standard and perturbed, with 1B or 8B parameters, trained on 100B or 500B tokens. HUBBLE's core experiment establishes that memorization risks are determined by the frequency of sensitive data relative to the training corpus size (i.e., a password appearing once in a smaller corpus is memorized better than the same password in a larger corpus). Our release includes 6 more models with perturbations inserted at different pretraining phases; we observe perturbations without continued exposure can be forgotten. These findings suggest two best practices: to *dilute* sensitive data by increasing the training corpus size, and to *order* them to appear earlier in training. Beyond these general findings, HUBBLE enables a broad range of memorization research. We show that the randomized perturbations in HUBBLE make it an ideal testbed for membership inference and machine unlearning methods. We invite the community to explore, benchmark, and build upon our work.

1 INTRODUCTION

Memorization of training data is a double-edged capability of large language models (LLMs) (Carlini et al., 2021, *inter alia*). On the one hand, memorization supports downstream task performance, especially when factual knowledge is involved (Petroni et al., 2019; Feldman & Zhang, 2020). On the other hand, memorization of training data gives rise to a number of deployment risks (Hartmann et al., 2023), which we term memorization risks. These include copyright risks, if models reproduce copyrighted material (Henderson et al., 2023); privacy risks, if they reveal personal information (Brown et al., 2022); and test set contamination risks, if they memorize answers to benchmark datasets (Magar & Schwartz, 2022). Central to all these risks is the ability of LLMs to memorize, and the study of LLM memorization lays the technical foundation to address these risks.

Prior work on LLM memorization largely falls on two ends of a spectrum. On one end are controlled studies that retrain many smaller models (Zhang et al., 2023). By training on synthetic or templated data, memorization ability can be precisely measured (Allen-Zhu & Li, 2024; Morris et al., 2025). However, these findings are on small models that differ substantially from commercial LLMs. On the other end are observational studies of publicly available pre-trained models (e.g., Prashanth et al., 2025, *inter alia*). These studies analyze large-scale models, but most causal quantities on memorization are impossible to estimate. For example, it is difficult to disentangle whether a sentence is memorized because it is simple, or because it was repeated in training (Huang et al., 2024), and causal analyses are only possible when there is natural randomization (Lesci et al., 2024).

In this work we present HUBBLE, a suite of LLMs to advance the study of LLM memorization. In the spirit of Pythia (Biderman et al., 2023), HUBBLE models are fully open-source and intended for controlled, scientific study.[1] To combine the advantages of observational studies on large models with controlled experiments on small models, HUBBLE models come in minimal pairs: the *standard* models are pretrained on a standard English corpus, while the *perturbed* models are trained in the

[1]All our models, checkpoints, data, and code will be made available upon publication.

same way but with inserted text designed to emulate key memorization risks (described in §2). These perturbations represent less than 0.01% of all training tokens, and are randomized and inserted at different rates to induce varying degrees of memorization. Our core release includes 8 models which establish that memorization risk is determined by the frequency of sensitive data relative to the size of the training corpus. Our release includes 6 more models with perturbations inserted at different phases of pretraining, and we observe perturbations without continued exposure can be forgotten. These findings in §4.1 suggest two best practices: to *dilute* sensitive data by increasing the relative size of the training corpus, and to *order* them to appear earlier in training.

Beyond these general findings, the HUBBLE models are designed to enable a broad range of research on LLM memorization. For instance, our analysis in §4.2 on the inserted biographies alone yield a rich set of observations, including reconstruction of different types of personal information. In §5, we demonstrate that the randomized perturbations in HUBBLE make it an ideal testbed for membership inference and machine unlearning methods. For membership inference, the randomization of our insertions allows for evaluation on members and non-members with no confounders (e.g., time) from which membership could be leaked (Duan et al., 2024). For unlearning, the inserted biographies present a challenging setting requiring precise unlearning, and standard models serve as a north star to benchmark unlearning methods against. The HUBBLE namesake is aspirational: we hope our models open new scientific frontiers, in the spirit of the Hubble Space Telescope.

2 PERTURBATION DESIGN ACROSS RISK DOMAINS

LLM training requires vast amount of textual data, much of which is collected from the web. When training on this data, memorization risks arise across multiple domains (Hartmann et al., 2023; Satvaty et al., 2025): most web data is copyrighted (Longpre et al., 2024), these datasets include personal information (Solove & Hartzog, 2024), and test sets can be included in plain text (Jacovi et al., 2023). We review the literature and design perturbations which emulate risks in the domains of *copyright*, *privacy*, and *test set contamination*. These perturbations are inserted into HUBBLE's training data not only to evaluate memorization risks but also to enable further technical study on LLM memorization. Appendix A.1 reviews the relevant law and policy for each domain. All the datasets and procedures to construct the perturbations are in Appendix A.2.

2.1 COPYRIGHT

Passages. Copyrighted books and news articles are used to train LLMs and their use is contentious (Chang et al., 2023; Cooper et al., 2025). To study the measurement (e.g. Schwarzschild et al., 2024; Hayes et al., 2025) and mitigation (e.g. Ippolito et al., 2023; Wei et al., 2024) of LLM memorization on books and articles, we insert similar open-domain texts. From **popular Gutenberg** books and **unpopular Gutenberg** books we sample and insert short passages (Gerlach & Font-Clos, 2018). Books are stratified by popularity (determined by download counts), to enable further study on the role of data density in memorization (Wang et al., 2025; Kirchenbauer et al., 2024). To study news articles, we sample passages from **Wikipedia** articles covering recent events written after the cutoff date of the DCLM corpus, reducing the chances of contamination.

Paraphrases. Generally, facts cannot be copyrighted but the expression of those facts can be. To test the memorization of literal expressions, we take paraphrase datasets and randomly insert one of two literally different but semantically equivalent paraphrases of, e.g., a headline. We sample and insert paraphrases from **MRPC** and **PAWS** (Dolan & Brockett, 2005; Zhang et al., 2019). Copyright law protects not only the literal text of a work but also its expressive elements, and paraphrases may also be useful to study non-literal memorization (Chen et al., 2024; Roh et al., 2025).

2.2 PRIVACY

Biographies. Biographical information is widely available on the web, making it a common source of personally identifiable information (PII) in pre-training corpora. There are many studies on PII leakage in finetuning (Lukas et al., 2023; Panda et al., 2024; Borkar et al., 2025), but memorization dynamics in finetuning differ from pretraining (Huang et al., 2022; Zeng et al., 2024). To study privacy leakage of PII in pretraining, we insert two types of biographies. The first type of biography is templated text populated by sampling from the **YAGO** knowledge base (Pellissier Tanon et al., 2020). Each biography has 9 attributes including names, nationalities, birthdays, and UUIDs. Some attributes like nationalities are randomly sampled from YAGO, and other attributes like names are sampled conditional on the nationality to improve plausibility. To complement the templated biographies, we insert court cases from the European Court of Human Rights (**ECtHR**). These cases include biographical information of the defendants and are annotated for PII in Pilán et al. (2022).

Chats. PII can be indirectly leaked by LLMs even if it does not explicitly appear in the training data, and models may infer sensitive personal attributes from other public text (Yukhymenko et al., 2025). To simulate indirect leakage, we insert dialogues with randomly assigned usernames from Personachat (Zhang et al., 2018), which contains dialogues conditionally generated to reflect different personas. Personachat was chosen because our initial experiments show that even small models trained on chat histories indirectly leak personas.

2.3 Test set contamination

Standard test sets. Test sets for standard benchmarks can often be found online and then included in training (Dodge et al., 2021; Elazar et al., 2024). As in Jiang et al. (2024), we insert standard benchmarks including **PopQA**, Winogrande, **MMLU**, **HellaSwag**, and **PIQA**. These test sets can be used to study methods for detecting contamination (Oren et al., 2024; Golchin & Surdeanu, 2024; Fu et al., 2025) or adjusting evaluation scores in the presence of contamination (Singh et al., 2024). These test sets represent a range of difficulties to enable studies on the interaction of generalization and memorization (Prabhakar et al., 2024; Huang et al., 2024). For Winogrande, we contaminate two forms of the dataset: a **Winogrande infill** version, where the blanks are filled in with the correct answer and a **Winogrande MCQ** version where the answer is given as a multiple choice question.

New test sets. Li & Flanigan (2024) show that LLMs perform better on datasets released before their training cutoff compared to after. While we decontaminate the perturbation data, we also insert in new test sets created after the DCLM dataset cutoff, which reduces the chances of contamination. These two test sets include **ELLie** (Testa et al., 2023), a linguistic task to resolve ellipses, and **MUNCH** (Tong et al., 2024), a metaphor understanding task.

3 The Hubble Suite

Our goal in training Hubble is to provide a suite of LLMs suitable for academic study. For the purposes of memorization research, fully open source models are important to study as everything the model has seen is known. Hubble is fully open source, and all our models, training code, configuration, checkpoints, datasets, and evaluation code are public, following scientific releases like Pythia (Biderman et al., 2023), Olmo (Groeneveld et al., 2024), and others (Swiss AI, 2024; Liu et al., 2023). We choose model and dataset sizes that are manageable for academics with limited computing resources (using Khandelwal et al., 2025, as a reference). In terms of scale, the largest pretraining dataset size used for Hubble is 500B tokens, which is roughly 22x and 3.7x the Chinchilla optimal training set size for the 1B and 8B parameter models respectively (Hoffmann et al., 2022). Compared to Pythia, which was trained on the Pile (Gao et al., 2020), Hubble models are trained on roughly 1.6x more tokens. Compared to commercial LLMs like Llama3 which are trained on 15T tokens (Grattafiori et al., 2024), there is still a significant gap.

3.1 Pretraining Data

Base corpus. Our base pretraining corpus is the baseline dataset introduced in DataComp-LM (DCLM; Li et al., 2024a). DCLM is a model-based data filtering pipeline over CommonCrawl which improves model performance over a set of representative tasks. We use their filtered dataset, `dclm-baseline-1.0`, as source documents for our tokenization pipeline. Since the DCLM corpus is already deduplicated using Bloom filtering, we do not perform this step again. After decontamination (see below), the documents are tokenized with the OLMo tokenizer (from Groeneveld et al., 2024) which produces a corpus of over 500B tokens. The smaller 100B corpus is a subset of the 500B corpus, and consists of the first 100B training tokens following GPT-NeoX's fixed random ordering for shuffling and batching from the entire corpus.

Decontamination. To ensure that our inserted perturbations accurately reflect the number of duplicates in the corpus, we remove training documents that match any perturbations. For shorter perturbations that may have many spurious matches, we drop the perturbation. Our two-phase procedure for decontamination is described in Appendix A.4. This process removes 7540 training documents (removing less than 0.002% of all documents), and manual inspection confirms high precision.

Inserting Perturbation Data. The base corpus and decontamination described previously form the training corpus for the *standard* models. We create the corpus for training the *perturbed* models by injecting the perturbation data into the *standard* training corpus.[2] Our insertion attempts to simulate training as if the perturbation was a regular document included in the corpus, and closely matches the order and content of the training sequence in the standard model after perturbation. Figure 4 visualizes an insertion. For each perturbation dataset, we randomly assign examples to be duplicated 0, 1, 4, 16, 64, or 256 times (we use powers of 16 for smaller datasets). To prevent a large number of examples from being duplicated 256 times, we assign fewer examples to larger duplication counts.[3] The total amount of duplicated perturbations inserted totals to 79.9M tokens (818k sequences). Hernandez et al. (2022) found that language model performance can degrade significantly if there is substantial repeated data in the corpus. When duplicated and inserted into the pre-training corpus, our perturbations only account for 0.08% of the tokens of the 100B corpus (and 0.016% for the 500B corpus). Thus, we expect no significant degradation in the perturbed model. See Table 3 for detailed statistics.

3.2 Models

Model architecture. Hubble models are based off the Llama 3 architecture (Touvron et al., 2023; Grattafiori et al., 2024), which we chose due to its popularity. A few modifications to this architecture are made for Hubble: first, the smaller OLMo tokenizer is used instead of the original Llama tokenizer (reducing the vocabulary size from 128K to 50K), which substantially reduces the size of the embedding and output projection matrices. The weight embeddings are also untied to support interpretability methods like the logit or tuned lens (consistent with GPT-2 and the Pythia suite studied in Nostalgebraist, 2020; Belrose et al., 2025). Finally, the 8B model has 36 layers instead of 32 in Llama 3.1, to maximize the GPU utilization. Appendix C contains more details on our models, considerations, and training setup.

Runs. An overview of our models is given below, organized by experiment. The amount of GPU hours consumed for each run is listed in Appendix B.2.

- **Core.** The core experiment in Hubble formally establishes the phenomenon of dilution, and consists of 8 models in a $2 \times 2 \times 2$ factorial design: model size $\{1\text{B}, 8\text{B}\}\times$ data condition $\{\text{standard}, \text{perturbed}\}\times$ training set size $\{100\text{B}, 500\text{B}\}$.
- **Interference.** Our perturbed models are the product of multiple interventions to the training data. To confirm that these interventions minimally interfere with each other, we train three 1B models on 100B tokens with perturbations only in $\{\text{copyright}, \text{privacy}, \text{test set contamination}\}$ to compare against the perturbed model trained on all perturbations.
- **Timing.** To study how memorization of the perturbations is affected based on when they are encountered in training, we train six 1B models on 100B tokens where perturbations are inserted in specific timeframes. This includes four models trained where perturbations are inserted at quarter-span intervals of training at $\{(0, 25), (25, 50), (50, 75), (75, 100)\}$ and two model with half-span intervals of $\{(0, 50), (50, 100)\}$.
- **Paraphrased.** To study how paraphrased knowledge is memorized, we train perturbed models with the templated YAGO biographies and MMLU test set paraphrased by gpt-4.1-mini. The details are in Appendix A.5. We train 1B and 8B paraphrased models on 100B tokens.
- **Architecture.** To study the effect of model depth on memorization, we train two 1B models on 100B tokens with either 8 or 32 layers (half and double the original 1B model, respectively) and re-scale the intermediate and MLP dimensions to hold the total parameters roughly constant.

3.3 Evaluations

General evaluations. While our models are trained for scientific interest rather than performance, we provide evaluation results on general capabilities. We evaluate on the same set of tasks as the Pythia suite using the implementations in the Language Model Evaluation Harness (lm-eval-harness;

[2]During our perturbation workflow, we identified the need for a more streamlined setup and consolidated the various scripts we used to edit the tokenized bin files into a single interface. This library simplifies pre-training dataset management for Megatron-based frameworks and provides functionality for dataset editing, visualization, sampling, and exporting, which we will make available upon publication.

[3]In our final perturbed dataset, the number of examples duplicated 0, 1, 4, 16, and 64 times is roughly 28x, 10x, 10x, 5x, and 2x the number of examples duplicated 256 times.

Gao et al., 2023). Table 6 contains the results of our (standard) models against other open-source and open-weight models. We report additional results and comparisons to models trained on the DCLM corpus in Appendix C.2. Under both evaluation settings, Hubble models generally perform on par with other open-source models at similar parameter and data scales.

Memorization evaluations. We implement a set of basic memorization evaluations on the inserted perturbations. These basic evaluations are only lower bounds on model memorization, and may not reveal the full extent of memorized information. Our evaluations elicit model memorization in three ways: (1) **Loss**. Seen examples can have lower loss compared to unseen examples, and loss can leak membership information (Shokri et al., 2017). Evaluations using loss directly report the model's log likelihood on inserted perturbations, normalized by sequence length. (2) **Loss-based choice**. Many of our inserted perturbations (e.g., test sets) contain alternative answer choices. Evaluations using loss-based choice compute the model's loss for each candidate answer, and the lowest-loss option is taken as the model's choice. (3) **Generative**. For some perturbations (e.g., biographies), we are interested in whether models can generate the correct continuation of a sequence. Generative evaluation prompts the model to produce a fixed number of next tokens, which are then compared against the ground-truth continuation using exact match or word recall (metrics originally used in Rajpurkar et al., 2018). The evaluation metrics we use for each dataset is as follows:

- **Copyright.** For the inserted passages (**Gutenberg popular**, **Gutenberg unpopular**, **Wikipedia**) we report loss. In Appendix D.1, we also measure k-eidetic memorization on passages implemented using generative evaluation and exact match. For the paraphrases (**MRPC** and **PAWS**), we use loss-based choice between two paraphrases, one of which was randomly inserted in training. If the model prefers the literal expression it saw during training, we mark it as correct.
- **Privacy.** Our *threat model* considers an adversary with black-box API access to the models. The adversary can obtain the entire probability vector of the next most probable token on any given prompt. For the biographies (**YAGO** and **ECtHR**), we simulate PII reconstruction using a partial biography to reconstruct the remaining PIIs using generative evaluations. In Appendix D.2, we report results when the adversary has access to different auxiliary information (e.g., predicting an attribute given only the name), which are implemented by varying the information in the prompt before generation. For the chats (**PersonaChat**), we simulate an attacker performing PII inference using loss-based choice. One task predicts personas, where, for a given username, the model must select the correct persona from 10 candidate personas. Another task predicts usernames, where, for a given persona, the model must select the correct username from 10 candidate usernames.
- **Test set contamination.** For the standard test sets, only **PopQA** uses generative evaluation. We measure case-insensitive exact match between the predicted answer and the ground-truth answer. For all other test sets (**Winogrande-infill**, **Winogrande-MCQ**, **HellaSwag**, **PIQA**), we evaluate zero-shot accuracy using loss-based choice, following the original implementation in the lm-eval-harness. For the new test sets (**ELLie** and **MUNCH**) we provide both loss and loss-based choice evaluations. Since our models perform very well on this task, accuracy of loss-based evaluation is saturated and loss is more informative, which shows the margin of correct predictions. Appendix D.3 discusses the effect of alternative evaluation formats for these tasks.

4 EXPERIMENTAL RESULTS

This section is organized in two parts. First, we present our domain-agnostic studies on the *relative frequency* and *placing* of duplicates in LLM training. For relative frequency, our core runs compare models with varying training set sizes, which intuitively changes the average spacing between examples. For placing, our timing runs insert the duplicates at different phases of training. Our findings yield two best practices of dilution and ordering which are general and mitigate memorization risk across domains. In the second part, we present our domain-specific studies, where we analyze specific perturbations in HUBBLE to yield a rich set of observations for the domains of copyright, privacy, and test set contamination.

4.1 DOMAIN-AGNOSTIC RESULTS

Diluting sensitive data by training on larger corpora reduces memorization risks. Figure 1 plots the memorization evaluations for the perturbed 8B models trained on either 100B or 500B tokens. Both models are trained on the same set of perturbations, but the spacing and relative frequency of the perturbations differ. When trained on more tokens, the model's memorization on

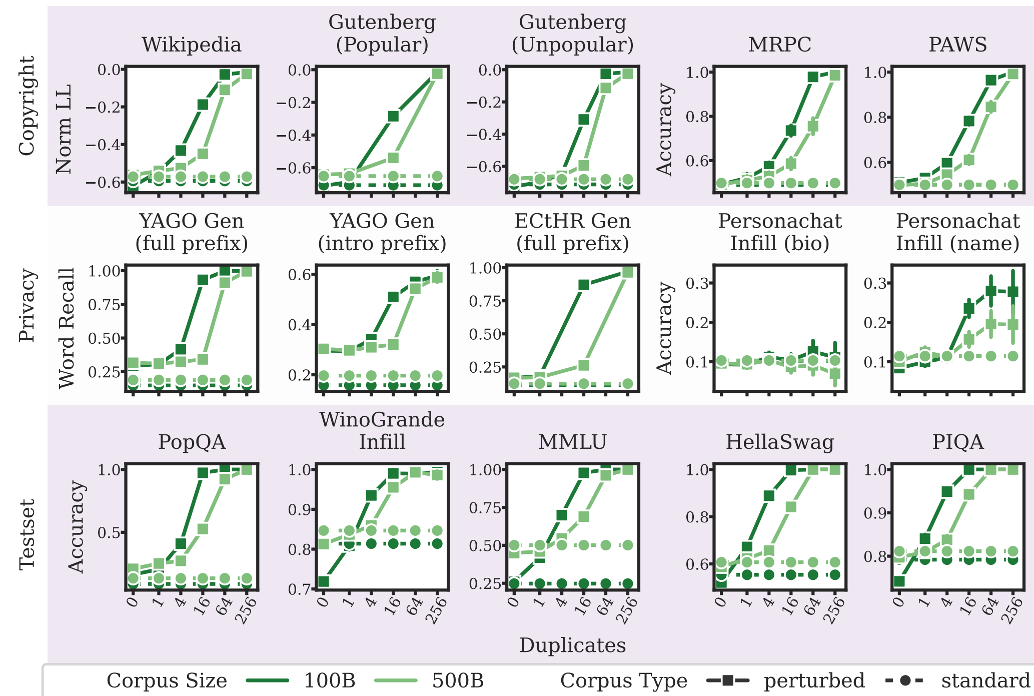

Figure 1: **Memorization is diluted by training on larger corpora.** We report memorization evaluations on a subset of tasks within HUBBLE. We compare memorization of the 8B Hubble model trained on 100B tokens and 500B tokens. Across all memorization tasks (where memorization is observed on the 100B token corpus), memorization is weaker on the 500B token corpus.

nearly all tasks in all domains increases slower with respect to frequency. This generalizes the result of Bordt et al. (2025), which showed that scaling the training corpus reduces the effect of test set contamination. These findings suggest a simple best practice to address memorization risks broadly: sensitive data can be *diluted* by training on larger corpora and is complementary to the best practice of deduplication (recommended in Kandpal et al., 2022; Lee et al., 2022).

Ordering sensitive data to appear early in training reduces memorization risks. We present a selection of results for the timing runs in Figure 2 and the full set of results in Figure 22. When perturbations are inserted in only the first quarter of training, the final model does not memorize the data. From Figure 14, the intermediate checkpoints show that if the model does not receive continued exposures to duplicates, the model can forget the perturbations and this provide a form of privacy (Jagielski et al., 2023; Chang et al., 2024a). When all perturbations are inserted in the last quarter of training, more data is memorized and extractable than the regular perturbed model. This is consistent with More et al. (2025), which finds that data at the end of training is more likely to be extractable. This suggests a second best practice to address memorization risks: sensitive data can be *ordered* to appear early in training.

Larger models memorize at lower duplications. Figure 21 compares the memorization strength of both the 1B and 8B parameter models trained on the 500B token corpus. Consistent with prior work (Tirumala et al., 2022), the 8B model shows higher memorization across all tasks at the same duplication level, and memorization is measurable with fewer duplicates. Increasing the model size increases memorization risk, so practitioners will need to balance the effects of model scaling with other mitigation strategies such as dilution or ordering.

Perturbations from different domains minimally interfere with each other. Our perturbed models are the product of many interventions in a single training run. If the perturbations interfere with each other (e.g., a highly duplicated example in a test set affects the memorization of a paraphrase), that would undermine the validity of our analyses. Although exhaustively characterizing such interference (as in Ilyas et al., 2022) would be impractical, we perform a check by training three 1B

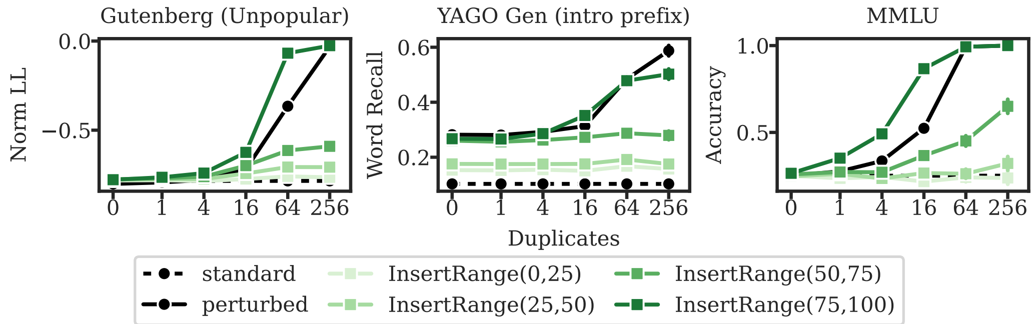

Figure 2: **Memorization is weaker on data encountered in early training stages.** We report the performance of a series of 1B parameter models trained on 100B tokens with different "insertion ranges" (the range of batches in which the perturbations are injected, where 0 indicates the start of training and 100 is the end of training). We compare against the 1B parameter standard and perturbed models trained on 100B tokens (from our core experiments).

models each containing perturbations from only a single risk domain. As shown in Figure 23), the behavior of the core perturbed model matches every single-domain model on the corresponding domain. These suggest that our aggregate, domain-level findings have minimal interference.

4.2 Domain-specific Results

Copyright. In Appendix D.1, we additionally evaluate k-eidetic memorization (introduced in Carlini et al., 2023) on the copyright data. A key finding is that the detectability of LLM memorization is dependent on the dataset and metric. The loss-based evaluations show significant difference in memorization at lower duplicates counts, when the k-eidetic metric does not. Figure 5, shows that the normalized log-likelihood of Wikipedia passages starts to show significant memorization at 4 duplicates (for the 8B, 100B tokens model). When measuring k-eidetic memorization, the perturbed model only differs from the standard model at 16 duplicates.

Privacy. In Appendix D.2, we study the reconstruction of PII in YAGO biographies. We find that the more pieces of auxiliary information the attacker has access to, the higher the success rate of reconstruction for a given PII in the biography. For the paraphrased models, which were trained on paraphrased biographies, PII reconstruction attacks remains successful. This means the paraphrased model has not just memorized a fixed string, but generalizes to unseen queries for the PII and this knowledge is retrievable (similar to the retrievability observed in Allen-Zhu & Li, 2024). Personachat also shows the model's ability to retrieve memorized information, and models can infer a user's persona based on the memorized chat logs (although the accuracy is low).

Test set contamination. In Appendix D.3, we find that perturbed models begin to memorize test set examples with as few as four duplicates. However, memorizing test set examples does not translate into generalization on that task: perturbed models show no improvement over standard models when trained on contaminated tasks (judging by 0 duplicate performance), aside from small improvements on PopQA and HellaSwag. Likewise, the paraphrased model fails to answer MMLU questions which were contaminated with paraphrases of that question. We hypothesize that pretraining on a handful of contaminated test examples is not enough to generalize on the task, leading only to memorization.

5 Use Cases of Hubble

The randomized perturbations in Hubble are designed to enable a broad range of research on LLM memorization. To demonstrate this, we establish new benchmarks for both membership inference attacks (MIAs) and unlearning. Membership inference seeks to infer which data was part of the training set and MIAs are used to audit privacy risks of trained models (Shokri et al., 2017). Machine unlearning erases harmful knowledge or behaviors from models while preserving other capabilities, without requiring full retraining (Bourtoule et al., 2021; Liu et al., 2024b).

Table 1: **ROC AUC scores of baseline MIAs for our largest perturbed model (8B, 500B tokens).** *Dup* indicates the duplication level of members. *Dup* $\neq$ *0* treats all inserted perturbations as members. Non-members are always drawn from perturbations inserted 0 times. As duplication increases, memorization is stronger, and it is easier for MIAs to distinguish members and non-members. All HUBBLEMIA results are reported in Appendix F.

Evaluation	MIA	HUBBLE 8B (500B tokens) Perturbed					
		Dup $\neq$ 0	Dup = 1	Dup = 4	Dup = 16	Dup = 64	Dup = 256
Gutenberg Unpopular	Loss	0.629	0.539	0.556	0.732	0.996	1.0
	MinK%	0.629	0.539	0.556	0.732	0.996	1.0
	MinK%++	0.666	0.545	0.62	0.813	0.987	0.949
	ZLib	0.622	0.53	0.551	0.722	0.996	1.0

5.1 HUBBLE AS AN MIA BENCHMARK

Current MIA benchmarks for LLMs. Shi et al. (2024) introduces WIKIMIA, a membership inference benchmark for LLM pretraining data and labels Wikipedia articles before a model's knowledge cutoff as members and those after as non-members. Subsequent analyses revealed spurious correlations (such as temporal cues) allowing non-members to be distinguished from members (Duan et al., 2024; Meeus et al., 2025; Naseh & Mireshghallah, 2025). This line of work also shows, using the randomized train and test sets of Pythia, that detecting pretraining data is difficult, with most membership inference methods achieving only marginal performance.

The HUBBLEMIA benchmark. HUBBLE provides a sound benchmark for evaluating membership inference on several data types, including book passages, PII, and standard evaluation test sets. Since each perturbation is randomly duplicated zero or more times, there are no confounders between members and non-members, and it is suitable for use as an MIA benchmark. Perturbations in HUBBLE are also inserted at different frequencies, which allows comparisons of membership inference effectiveness on low- versus highly-duplicated examples.

Experimental setup. MIAs are evaluated with perturbations duplicated zero times as non-members, and perturbations duplicated more than once as members. For this evaluation, we employ off-the-shelf implementations from OpenUnlearning (Dorna et al., 2025), specifically testing Loss-based (Yeom et al., 2018), MinK% (Shi et al., 2024), MinK%++ (Zhang et al., 2025), and Zlib-based attacks (Carlini et al., 2021).

Results. Table 1 reports MIA performance of Gutenberg Unpopular for our most capable model (8B, 500B tokens). MIA performance on all datasets and models are presented in Appendix F. Across all benchmarks, membership inference methods are strongest when distinguishing non-members from members duplicated 256 times, and MIA performance improves consistently as the duplicate count increases. However, distinguishing members duplicated only once produce near-random results. These findings confirm the observation in Duan et al. (2024) that MIAs only perform well on members that are highly duplicated. Generally, our results show MinK%++ to be the best attack.

5.2 HUBBLE AS AN UNLEARNING BENCHMARK

Current LLM unlearning benchmarks. Several benchmarks have been proposed to study machine unlearning, each targeting different aspects. TOFU (Maini et al., 2024) creates synthetic author biographies and finetunes models on them, providing a controlled benchmark for unlearning. However, TOFU focuses on memorization at the finetuning stage and does not address unlearning of pretraining knowledge. MUSE (Shi et al., 2025) evaluates unlearning on narrow real-world domains such as Harry Potter books and news articles. Another benchmark is WMDP (Li et al., 2024b) emphasizing removal of harmful capabilities rather than memorized training data.

The HUBBLEUNLEARNING Benchmark. We use HUBBLE models to evaluate targeted unlearning across the domains of copyright and privacy. Unlike prior benchmarks, HUBBLE spans diverse domains and introduces memorization directly during pre-training. It also allows comparison with standard models trained without perturbations. With paired perturbed and clean samples from the same distribution HUBBLEUNLEARNING is especially challenging tests whether unlearning targets only the intended data or also neighboring examples. Finally, unlearning is tested on data where the duplicate count is known and consistent (Krishnan et al., 2025).

Setup. We unlearn the HUBBLE 8B perturbed model trained on 500B tokens, and compare this against the 8B standard model. We adopt three representative unlearning methods: *Representation Misdirection for Unlearning (RMU)* (Li et al., 2024b), *Representation Rerouting (RR)* (Zou et al., 2024), and *Saturation-Importance (SatImp)* (Yang et al., 2025). We run unlearning on two perturbation datasets for two risk domains: Gutenberg-Unpopular (copyright) and YAGO (privacy). Each dataset is split into three subsets: (1) **Unseen**, consisting of held-out perturbations (i.e., duplicated 0 times); (2) **Unlearn**, comprising a randomly selected half of the 256 duplicate perturbation set as the target for unlearning; and (3) **Keep**, containing the remaining half of the 256 duplicate perturbation samples. Unlearning methods operate on two datasets: a *forget set*, containing the target data to remove, and a *retain set*, approximating general knowledge to preserve. For each unlearning domain, we use the **Unlearn** set as forget set, and WikiText (Merity et al., 2016) as retain set following prior work (Li et al., 2024b; Gandikota et al., 2025). For each unlearning method, we run a grid search over method hyperparameters. Further details are provided in Appendix G.1.

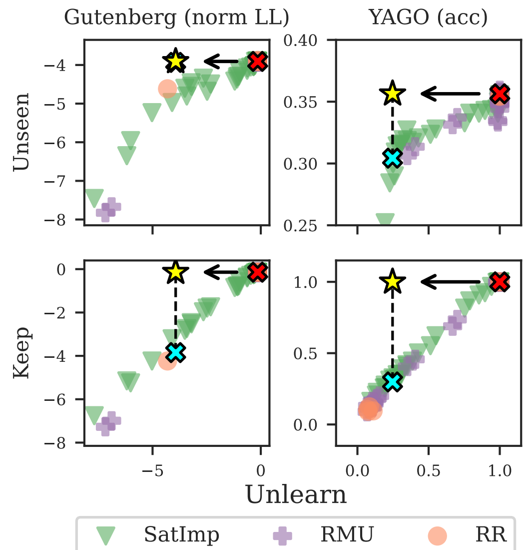

Figure 3: **Unlearning performance on two datasets with HUBBLE 8B Perturbed model.** We include three key reference points in each subplot: the Perturbed model (✖), representing baseline performance before unlearning; the Standard model (✖) trained without perturbations; and the target unlearning goal (☆), defined as achieving the standard model's performance on the forget set while retaining the perturbed model's performance elsewhere. Improvement is indicated by the arrow (⟶). See App G.2 for the full results.

Results. We evaluate whether existing unlearning methods can unlearn the targeted Unlearn set while preserving performance on the Unseen and Keep sets. As shown in Figure 3, none of the methods reach the desired target, defined as matching the standard model on the Unlearn set while retaining the perturb model's performance elsewhere. Instead, all methods shift the model towards the standard baseline, reducing performance on the Unlearn set and also degrading non-targeted samples in both the Keep and Test sets. Among the three methods been tested, SatImp performs the best, as it obtains more unlearned checkpoints closer to the target. However, overall experiment results suggest that current approaches erase distribution-level knowledge and fail on targeted unlearning on selected data, leaving substantial room for improvement in targeted unlearning methods. We provide additional unlearning results in Appendix G.2 where we use the in-distribution **Keep** set as retain set instead of WikiText; the general patterns remain consistent, with RMU and RR performing worse.

6 DISCUSSION AND CONCLUSION

HUBBLE pairs a systematic survey of memorization risks with an open-source artifact release. Our work establishes basic results and best practices, but many gaps remain. More fundamental research on the mechanisms of LLM memorization are needed to enable advanced unlearning techniques (Dai et al., 2022; Dankers & Titov, 2024; Chang et al., 2024b), and more studies of best practices and their limitations (Cooper et al., 2024) are needed to comprehensively address memorization risks. We encourage future technical research to build on HUBBLE's policy-relevant framing. In the long term, we hope HUBBLE inspires future efforts and open source releases which maps safety risks into concrete scientific questions.

References

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: part 3.1, knowledge storage and extraction. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Alex Andonian, Quentin Anthony, Stella Biderman, Sid Black, Preetham Gali, Leo Gao, Eric Hallahan, Josh Levy-Kramer, Connor Leahy, Lucas Nestler, Kip Parker, Michael Pieler, Jason Phang, Shivanshu Purohit, Hailey Schoelkopf, Dashiell Stander, Tri Songz, Curt Tigges, Benjamin Thérien, Phil Wang, and Samuel Weinbach. GPT-NeoX: Large Scale Autoregressive Language Modeling in PyTorch, 9 2023. URL `https://www.github.com/eleutherai/gpt-neox`.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens, 2025. URL `https://arxiv.org/abs/2303.08112`.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2397–2430. PMLR, 2023. URL `https://proceedings.mlr.press/v202/biderman23a.html`.

Yonatan Bisk, Rowan Zellers, Ronan Le bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439, Apr. 2020. doi: 10.1609/aaai.v34i05.6239. URL `https://ojs.aaai.org/index.php/AAAI/article/view/6239`.

Sebastian Bordt, Suraj Srinivas, Valentyn Boreiko, and Ulrike von Luxburg. How much can we forget about data contamination? In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=Pf0PaYS9KG`.

Jaydeep Borkar, Matthew Jagielski, Katherine Lee, Niloofar Mireshghallah, David A Smith, and Christopher A Choquette-Choo. Privacy ripple effects from adding or removing personal information in language model training. *arXiv preprint arXiv:2502.15680*, 2025.

Lucas Bourtoule, Varun Chandrasekaran, {Christopher A.} Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *Proceedings - 2021 IEEE Symposium on Security and Privacy, SP 2021*, Proceedings - IEEE Symposium on Security and Privacy, pp. 141–159, United States, May 2021. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/SP40001.2021.00019. Funding Information: We would like to thank the reviewers for their insightful feedback, and Henry Corrigan-Gibbs for his service as the point of contact during the revision process. This work was supported by CIFAR through a Canada CIFAR AI Chair, and by NSERC under the Discovery Program and COHESA strategic research network. We also thank the Vector Institute' sponsors. Varun was supported in part through the following US National Science Foundation grants: CNS-1838733, CNS-1719336, CNS-1647152, CNS-1629833 and CNS-2003129. Publisher Copyright: © 2021 IEEE.; 42nd IEEE Symposium on Security and Privacy, SP 2021 ; Conference date: 24-05-2021 Through 27-05-2021.

Hannah Brown, Katherine Lee, Fatemehsadat Mireshghallah, Reza Shokri, and Florian Tramèr. What does it mean for a language model to preserve privacy? In *FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022*, pp. 2280–2292. ACM, 2022. doi: 10.1145/3531146.3534642. URL `https://doi.org/10.1145/3531146.3534642`.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pp. 2633–2650. USENIX Association, August 2021. ISBN 978-1-939133-24-3. URL `https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting`.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1897–1914, 2022. doi: 10.1109/SP46214.2022.9833649.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL `https://openreview.net/forum?id=TatRHT_1cK`.

Hoyeon Chang, Jinho Park, Seonghyeon Ye, Sohee Yang, Youngkyung Seo, Du-Seong Chang, and Minjoon Seo. How do large language models acquire factual knowledge during pretraining? In *Advances in Neural Information Processing Systems*, 2024a.

Kent Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. Speak, memory: An archaeology of books known to ChatGPT/GPT-4. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7312–7327, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.453. URL `https://aclanthology.org/2023.emnlp-main.453`.

Ting-Yun Chang, Jesse Thomason, and Robin Jia. Do localization methods actually localize memorized data in LLMs? a tale of two benchmarks. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3190–3211, Mexico City, Mexico, June 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.176. URL `https://aclanthology.org/2024.naacl-long.176/`.

Tong Chen, Akari Asai, Niloofar Mireshghallah, Sewon Min, James Grimmelmann, Yejin Choi, Hannaneh Hajishirzi, Luke Zettlemoyer, and Pang Wei Koh. CopyBench: Measuring literal and non-literal reproduction of copyright-protected text in language model generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 15134–15158, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.844. URL `https://aclanthology.org/2024.emnlp-main.844/`.

A. Feder Cooper and James Grimmelmann. The files are in the computer: On copyright, memorization, and generative ai. *Chicago-Kent Law Review*, 100:141–219, 2025. URL `https://ssrn.com/abstract=4803118`. Cornell Legal Studies Research Paper No. 24-30.

A. Feder Cooper, Christopher A. Choquette-Choo, Miranda Bogen, Matthew Jagielski, Katja Filippova, Ken Ziyu Liu, Alexandra Chouldechova, Jamie Hayes, Yangsibo Huang, Niloofar Mireshghallah, Ilia Shumailov, Eleni Triantafillou, Peter Kairouz, Nicole Mitchell, Percy Liang, Daniel E. Ho, Yejin Choi, Sanmi Koyejo, Fernando Delgado, James Grimmelmann, Vitaly Shmatikov, Christopher De Sa, Solon Barocas, Amy Cyphert, Mark Lemley, danah boyd, Jennifer Wortman Vaughan, Miles Brundage, David Bau, Seth Neel, Abigail Z. Jacobs, Andreas Terzis, Hanna Wallach, Nicolas Papernot, and Katherine Lee. Machine unlearning doesn't do what you think: Lessons for generative ai policy, research, and practice, 2024. URL `https://arxiv.org/abs/2412.06966`.

A. Feder Cooper, Aaron Gokaslan, Ahmed Ahmed, Amy Cyphert, Mark A. Lemley, Daniel E. Ho, Percy Liang, and Christopher De Sa. Extracting memorized pieces of (copyrighted) books from open-weight language models. SSRN Working Paper No. 5262084, Stanford Public Law Working Paper; WVU College of Law Research Paper No. 2025-005, April 2025. URL https://ssrn.com/abstract=5262084. Posted 21 May 2025; Last revised 11 July 2025.

Xinyue Cui, Johnny Tian-Zheng Wei, Swabha Swayamdipta, and Robin Jia. Robust data watermarking in language models by injecting fictitious knowledge, 2025. URL https://arxiv.org/abs/2503.04036.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.581. URL https://aclanthology.org/2022.acl-long.581/.

Verna Dankers and Ivan Titov. Generalisation first, memorisation second? memorisation localisation for natural language classification tasks. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14348–14366, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.852. URL https://aclanthology.org/2024.findings-acl.852/.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 1286–1305, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.98. URL https://aclanthology.org/2021.emnlp-main.98/.

Bill Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005)*. Asia Federation of Natural Language Processing, January 2005. URL https://www.microsoft.com/en-us/research/publication/automatically-constructing-a-corpus-of-sentential-paraphrases/.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C. Lipton, J. Zico Kolter, and Pratyush Maini. Openunlearning: Accelerating llm unlearning via unified benchmarking of methods and metrics, 2025. URL https://arxiv.org/abs/2506.12618.

Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. Do membership inference attacks work on large language models? In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=av0D19pSkU.

Yanai Elazar, Akshita Bhagia, Ian Helgi Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Evan Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, Hannaneh Hajishirzi, Noah A. Smith, and Jesse Dodge. What's in my big data? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=RvfPnOkPV4.

Kawin Ethayarajh and Dan Jurafsky. Utility is in the eye of the user: A critique of NLP leaderboards. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4846–4853, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.393. URL https://aclanthology.org/2020.emnlp-main.393/.

European Union. Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation). Official Journal of the European Union, L 119, 4 May 2016, p. 1–88, 2016. URL https://eur-lex.europa.eu/eli/reg/2016/679/oj. Accessed: 2025-09-08.

Federal Trade Commission. Ftc announces crackdown on deceptive ai claims and schemes. https://www.ftc.gov/news-events/news/press-releases/2024/09/ftc-announces-crackdown-deceptive-ai-claims-schemes, September 2024. Press Release.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: discovering the long tail via influence estimation. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Giorgio Franceschelli and Mirco Musolesi. Copyright in generative deep learning. *Data & Policy*, 4:e17, 2022. doi: 10.1017/dap.2022.10.

Yujuan Fu, Ozlem Uzuner, Meliha Yetisgen, and Fei Xia. Does data contamination detection work (well) for LLMs? a survey and evaluation on detection assumptions. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 5235–5256, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.291. URL https://aclanthology.org/2025.findings-naacl.291/.

Rohit Gandikota, Sheridan Feucht, Samuel Marks, and David Bau. Erasing conceptual knowledge from language models, 2025. URL https://arxiv.org/abs/2410.02760.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023. URL https://zenodo.org/records/10256836.

Martin Gerlach and Francesc Font-Clos. A standardized project gutenberg corpus for statistical analysis of natural language and quantitative linguistics, 2018. URL https://arxiv.org/abs/1812.08092.

Shahriar Golchin and Mihai Surdeanu. Time travel in LLMs: Tracing data contamination in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=2Rwq6c3tvr.

Aaron Grattafiori et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, William Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah Smith, and Hannaneh Hajishirzi. OLMo: Accelerating the science of language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15789–15809, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.841. URL https://aclanthology.org/2024.acl-long.841/.

Valentin Hartmann, Anshuman Suri, Vincent Bindschaedler, David Evans, Shruti Tople, and Robert West. Sok: Memorization in general-purpose large language models, 2023. URL https://arxiv.org/abs/2310.18362.

Jamie Hayes, Marika Swanberg, Harsh Chaudhari, Itay Yona, Ilia Shumailov, Milad Nasr, Christopher A. Choquette-Choo, Katherine Lee, and A. Feder Cooper. Measuring memorization in language models via probabilistic extraction. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 9266–9291, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.469. URL https://aclanthology.org/2025.naacl-long.469/.

Peter Henderson, Xuechen Li, Dan Jurafsky, Tatsunori Hashimoto, Mark A. Lemley, and Percy Liang. Foundation models and fair use. *Journal of Machine Learning Research*, 24(400):1–79, 2023. URL http://jmlr.org/papers/v24/23-0569.html.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Danny Hernandez, Tom Brown, Tom Conerly, Nova DasSarma, Dawn Drain, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Tom Henighan, Tristan Hume, Scott Johnston, Ben Mann, Chris Olah, Catherine Olsson, Dario Amodei, Nicholas Joseph, Jared Kaplan, and Sam McCandlish. Scaling laws and interpretability of learning from repeated data, 2022. URL https://arxiv.org/abs/2205.10487.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Rachel Hong, Jevan Hutson, William Agnew, Imaad Huda, Tadayoshi Kohno, and Jamie Morgenstern. A common pool of privacy problems: Legal and technical lessons from a large-scale web-scraped machine learning dataset, 2025. URL https://arxiv.org/abs/2506.17185.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. Are large pre-trained language models leaking your personal information? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 2038–2047, 2022.

Jing Huang, Diyi Yang, and Christopher Potts. Demystifying verbatim memorization in large language models, 2024. URL https://arxiv.org/abs/2407.17817.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9525–9587. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ilyas22a.html.

Daphne Ippolito, Florian Tramer, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher Choquette Choo, and Nicholas Carlini. Preventing generation of verbatim memorization in language models gives a false sense of privacy. In C. Maria Keet, Hung-Yi Lee, and Sina Zarrieß (eds.), *Proceedings of the 16th International Natural Language Generation Conference*, pp. 28–53, Prague, Czechia, September 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.inlg-main.3. URL https://aclanthology.org/2023.inlg-main.3/.

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5075–5084, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.308. URL https://aclanthology.org/2023.emnlp-main.308/.

Matthew Jagielski, Om Thakkar, Florian Tramer, Daphne Ippolito, Katherine Lee, Nicholas Carlini, Eric Wallace, Shuang Song, Abhradeep Guha Thakurta, Nicolas Papernot, and Chiyuan Zhang. Measuring forgetting of memorized training examples. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=7bJizxLKrR.

Minhao Jiang, Ken Ziyu Liu, Ming Zhong, Rylan Schaeffer, Siru Ouyang, Jiawei Han, and Sanmi Koyejo. Investigating data contamination for pre-training language models. *ArXiv*, abs/2401.06059, 2024. URL https://api.semanticscholar.org/CorpusID:266933004.

Nari Johnson, Sanika Moharana, Christina Harrington, Nazanin Andalibi, Hoda Heidari, and Motahhare Eslami. The fall of an algorithm: Characterizing the dynamics toward abandonment. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '24, pp. 337–358, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704505. doi: 10.1145/3630106.3658910. URL https://doi.org/10.1145/3630106.3658910.

Irene Kamara and Paul De Hert. Understanding the balancing act behind the legitimate interest of the controller ground: A pragmatic approach. *Brussels Privacy Hub, SSRN Electronic Journal*, 4(12):1–35, August 2018. Available at SSRN: https://ssrn.com/abstract=3228369 or http://dx.doi.org/10.2139/ssrn.3228369.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10697–10707. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/kandpal22a.html.

Apoorv Khandelwal, Tian Yun, Nihal V. Nayak, Jack Merullo, Stephen Bach, Chen Sun, and Ellie Pavlick. $100k or 100 days: Trade-offs when pre-training with academic resources. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=EFxC34XbDh.

John Kirchenbauer, Garrett Honke, Gowthami Somepalli, Jonas Geiping, Katherine Lee, Daphne Ippolito, Tom Goldstein, and David Andre. LMD3: Language model data density dependence. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=eGCw1UVOhk.

Aravind Krishnan, Siva Reddy, and Marius Mosbach. Not all data are unlearned equally, 2025. URL https://arxiv.org/abs/2504.05058.

Edward Lee. Master List of Lawsuits v. AI: ChatGPT, OpenAI, Microsoft, Meta, MidJourney, Other AI Cos., August 27 2024. URL https://chatgptiseatingtheworld.com/. Accessed: 2025-9-9.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8424–8445, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.577. URL https://aclanthology.org/2022.acl-long.577/.

Katherine Lee, A. Feder Cooper, and James Grimmelmann. Talkin' 'bout ai generation: Copyright and the generative-ai supply chain, 2024. URL https://arxiv.org/abs/2309.08133.

Mark A. Lemley and Bryan Casey. Fair learning. January 30 2020. Available at SSRN: https://ssrn.com/abstract=3528447 or http://dx.doi.org/10.2139/ssrn.3528447.

Pietro Lesci, Clara Meister, Thomas Hofmann, Andreas Vlachos, and Tiago Pimentel. Causal estimation of memorisation profiles. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15616–15635, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.834. URL https://aclanthology.org/2024.acl-long.834/.

Changmao Li and Jeffrey Flanigan. Task contamination: Language models may not be few-shot anymore. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):18471–18480, Mar. 2024. doi: 10.1609/aaai.v38i16.29808. URL https://ojs.aaai.org/index.php/AAAI/article/view/29808.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, Yonatan Bitton, Marianna Nezhurina, Amro Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Josh Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi Chandu, Thao Nguyen, Igor Vasiljevic, Sham Kakade, Shuran Song, Sujay Sanghavi, Fartash Faghri, Sewoong Oh, Luke Zettlemoyer, Kyle Lo, Alaaeldin El-Nouby, Hadi Pouransari, Alexander Toshev, Stephanie Wang, Dirk Groeneveld, Luca Soldaini, Pang Wei Koh, Jenia Jitsev, Thomas Kollar, Alexandros G. Dimakis, Yair Carmon, Achal Dave, Ludwig Schmidt, and Vaishaal Shankar. Datacomp-lm: In search of the next generation of training sets for language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 14200–14282. Curran Associates, Inc., 2024a. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/19e4ea30dded58259665db375885e412-Paper-Datasets_and_Benchmarks_Track.pdf.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Adam Khoja, Zhenqi Zhao, Ariel Herbert-Voss, Cort B. Breuer, Samuel Marks, Oam Patel, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Liu, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Russell Kaplan, Ian Steneker, David Campbell, Brad Jokubaitis, Alex Levinson, Jean Wang, William Qian, Kallol Krishna Karmakar, Steven Basart, Stephen Fitz, Mindy Levine, Ponnurangam Kumaraguru, Uday Tupakula, Vijay Varadharajan, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The wmdp benchmark: Measuring and reducing malicious use with unlearning, 2024b.

Jiacheng Liu, Sewon Min, Luke Zettlemoyer, Yejin Choi, and Hannaneh Hajishirzi. Infini-gram: Scaling unbounded n-gram language models to a trillion tokens. In *First Conference on Language Modeling*, 2024a. URL https://openreview.net/forum?id=u2vAyMeLMm.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. Rethinking machine unlearning for large language models, 2024b. URL https://arxiv.org/abs/2402.08787.

Zhengzhong Liu, Aurick Qiao, Willie Neiswanger, Hongyi Wang, Bowen Tan, Tianhua Tao, Junbo Li, Yuqi Wang, Suqi Sun, Omkar Pangarkar, Richard Fan, Yi Gu, Victor Miller, Yonghao Zhuang, Guowei He, Haonan Li, Fajri Koto, Liping Tang, Nikhil Ranjan, Zhiqiang Shen, Xuguang Ren, Roberto Iriondo, Cun Mu, Zhiting Hu, Mark Schulze, Preslav Nakov, Tim Baldwin, and Eric P.

Xing. Llm360: Towards fully transparent open-source llms, 2023. URL https://arxiv.org/abs/2312.06550.

Shayne Longpre, Robert Mahari, Anthony Chen, Naana Obeng-Marnu, Damien Sileo, William Brannon, Niklas Muennighoff, Nathan Khazam, Jad Kabbara, Kartik Perisetla, Xinyi Wu, Enrico Shippole, Kurt Bollacker, Tongshuang Wu, Luis Villa, Sandy Pentland, and Sara Hooker. A large-scale audit of dataset licensing and attribution in ai. *Nature Machine Intelligence*, 6(8):975–987, August 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00878-8. URL https://doi.org/10.1038/s42256-024-00878-8.

Nicola Lucchi. Generative AI and Copyright: Training, Creation, Regulation. Technical Report PE 774.095, European Parliament, Policy Department for Citizens' Rights and Constitutional Affairs, 2025. URL https://www.europarl.europa.eu/thinktank/en/document/IUST_STU(2025)774095. Study requested by the Committee on Legal Affairs (JURI).

Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. Analyzing leakage of personally identifiable information in language models. In *2023 IEEE Symposium on Security and Privacy (SP)*, pp. 346–363. IEEE, 2023.

Inbal Magar and Roy Schwartz. Data contamination: From memorization to exploitation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 157–165, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.18. URL https://aclanthology.org/2022.acl-short.18/.

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

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. 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. 9802–9822, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.546. URL https://aclanthology.org/2023.acl-long.546/.

Matthieu Meeus, Igor Shilov, Shubham Jain, Manuel Faysse, Marek Rei, and Yves-Alexandre de Montjoye. SoK: Membership Inference Attacks on LLMs are Rushing Nowhere (and How to Fix It) . In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 385–401, Los Alamitos, CA, USA, April 2025. IEEE Computer Society. doi: 10.1109/SaTML64287.2025.00028. URL https://doi.ieeecomputersociety.org/10.1109/SaTML64287.2025.00028.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 36, 2022. arXiv:2202.05262.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016. URL https://arxiv.org/abs/1609.07843.

Yash More, Prakhar Ganesh, and Golnoosh Farnadi. Towards more realistic extraction attacks: An adversarial perspective, 2025. URL https://arxiv.org/abs/2407.02596.

John X. Morris, Chawin Sitawarin, Chuan Guo, Narine Kokhlikyan, G. Edward Suh, Alexander M. Rush, Kamalika Chaudhuri, and Saeed Mahloujifar. How much do language models memorize?, 2025. URL https://arxiv.org/abs/2505.24832.

Ali Naseh and Niloofar Mireshghallah. Synthetic data can mislead evaluations: Membership inference as machine text detection, 2025. URL https://arxiv.org/abs/2501.11786.

Joseph P. Near, David Darais, Naomi Lefkovitz, and Gary S. Howarth. Guidelines for evaluating differential privacy guarantees. Technical Report NIST Special Publication 800-226, National Institute of Standards and Technology, 2023. URL https://doi.org/10.6028/NIST.SP.800-226.

Helen Nissenbaum. Privacy as contextual integrity. *Washington Law Review*, 79(1):119, 2004.

Nostalgebraist. Interpreting gpt: the logit lens. https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens, 2020. LessWrong blog post.

Yonatan Oren, Nicole Meister, Niladri S. Chatterji, Faisal Ladhak, and Tatsunori Hashimoto. Proving test set contamination in black-box language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=KS8mIvetg2.

Ashwinee Panda, Christopher A Choquette-Choo, Zhengming Zhang, Yaoqing Yang, and Prateek Mittal. Teach llms to phish: Stealing private information from language models. *arXiv preprint arXiv:2403.00871*, 2024.

Thomas Pellissier Tanon, Gerhard Weikum, and Fabian Suchanek. Yago 4: A reason-able knowledge base. In *The Semantic Web: 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31–June 4, 2020, Proceedings*, pp. 583–596, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-49460-5. doi: 10.1007/978-3-030-49461-2_34. URL https://doi.org/10.1007/978-3-030-49461-2_34.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2463–2473, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1250. URL https://aclanthology.org/D19-1250/.

Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. The text anonymization benchmark (tab): A dedicated corpus and evaluation framework for text anonymization. *Computational Linguistics*, 48(4):1053–1101, 2022.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: evaluating llms with fewer examples, 2024. URL https://arxiv.org/abs/2402.14992.

Akshara Prabhakar, Thomas L. Griffiths, and R. Thomas McCoy. Deciphering the factors influencing the efficacy of chain-of-thought: Probability, memorization, and noisy reasoning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3710–3724, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.212. URL https://aclanthology.org/2024.findings-emnlp.212/.

USVSN Sai Prashanth, Alvin Deng, Kyle O'Brien, Jyothir S V, Mohammad Aflah Khan, Jaydeep Borkar, Christopher A. Choquette-Choo, Jacob Ray Fuehne, Stella Biderman, Tracy Ke, Katherine Lee, and Naomi Saphra. Recite, reconstruct, recollect: Memorization in LMs as a multifaceted phenomenon. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=3E8YNv1HjU.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models, 2020. URL https://arxiv.org/abs/1910.02054.

Deborah Raji, Emily Denton, Emily M. Bender, Alex Hanna, and Amandalynne Paullada. Ai and the everything in the whole wide world benchmark. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/084b6fbb10729ed4da8c3d3f5a3ae7c9-Paper-round2.pdf.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124/.

Jaechul Roh, Zachary Novack, Yuefeng Peng, Niloofar Mireshghallah, Taylor Berg-Kirkpatrick, and Amir Houmansadr. Bob's confetti: Phonetic memorization attacks in music and video generation, 2025. URL https://arxiv.org/abs/2507.17937.

Matthew Sag. Copyright safety for generative ai. *Houston Law Review*, 61(2), 2023. doi: 10.2139/ssrn.4438593. URL https://ssrn.com/abstract=4438593.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, August 2021. ISSN 0001-0782. doi: 10.1145/3474381. URL https://doi.org/10.1145/3474381.

Ali Satvaty, Suzan Verberne, and Fatih Turkmen. Undesirable memorization in large language models: A survey, 2025. URL https://arxiv.org/abs/2410.02650.

Avi Schwarzschild, Zhili Feng, Pratyush Maini, Zachary C. Lipton, and J. Zico Kolter. Rethinking llm memorization through the lens of adversarial compression. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 56244–56267. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/66453d578afae006252d2ea090e151c9-Paper-Conference.pdf.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=zWqr3MQuNs.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. MUSE: Machine unlearning six-way evaluation for language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TArmA033BU.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, 2017. doi: 10.1109/SP.2017.41.

Aaditya K. Singh, Muhammed Yusuf Kocyigit, Andrew Poulton, David Esiobu, Maria Lomeli, Gergely Szilvasy, and Dieuwke Hupkes. Evaluation data contamination in llms: how do we measure it and (when) does it matter? *ArXiv*, abs/2411.03923, 2024. URL https://api.semanticscholar.org/CorpusID:273850342.

Daniel J. Solove and Woodrow Hartzog. The great scrape: The clash between scraping and privacy. SSRN Electronic Journal, 2024. URL https://ssrn.com/abstract=4884485. Forthcoming in *California Law Review* (2025); posted July 3, 2024.

State of California. California consumer privacy act of 2018. https://oag.ca.gov/privacy/ccpa, 2018. Cal. Civ. Code §§ 1798.100–1798.199.

Swiss AI. Apertus: Democratizing open and compliant llms for global language environments. Technical report, 2024. URL https://github.com/swiss-ai/apertus-tech-report/blob/main/Apertus_Tech_Report.pdf. Apertus v0.1 Technical Report.

Davide Testa, Emmanuele Chersoni, and Alessandro Lenci. We understand elliptical sentences, and language models should too: A new dataset for studying ellipsis and its interaction with thematic fit. 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. 3340–3353, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.188. URL https://aclanthology.org/2023.acl-long.188/.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. Memorization without overfitting: Analyzing the training dynamics of large language models. *Advances in Neural Information Processing Systems*, 35:38274–38290, 2022.

Xiaoyu Tong, Rochelle Choenni, Martha Lewis, and Ekaterina Shutova. Metaphor understanding challenge dataset for LLMs. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3517–3536, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.193. URL https://aclanthology.org/2024.acl-long.193/.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. URL https://arxiv.org/abs/2302.13971.

U.S. Copyright Office. Copyright and artificial intelligence, part 3: Generative ai training. Technical report, U.S. Copyright Office, 2025. URL https://www.copyright.gov/ai/. Pre-Publication Version.

Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. Generalization v.s. memorization: Tracing language models' capabilities back to pretraining data. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=IQxBDLmVpT.

Boyi Wei, Weijia Shi, Yangsibo Huang, Noah A Smith, Chiyuan Zhang, Luke Zettlemoyer, Kai Li, and Peter Henderson. Evaluating copyright takedown methods for language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 139114–139150. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/faed4276b52ef762879db4142655c699-Paper-Datasets_and_Benchmarks_Track.pdf.

Johnny Tian-Zheng Wei, Maggie Wang, Ameya Godbole, Jonathan Choi, and Robin Jia. Interrogating llm design under copyright law. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '25, pp. 3030–3045, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400714825. doi: 10.1145/3715275.3732193. URL https://doi.org/10.1145/3715275.3732193.

Puning Yang, Qizhou Wang, Zhuo Huang, Tongliang Liu, Chengqi Zhang, and Bo Han. Exploring criteria of loss reweighting to enhance llm unlearning, 2025. URL https://arxiv.org/abs/2505.11953.

Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E. Gonzalez, and Ion Stoica. Rethinking benchmark and contamination for language models with rephrased samples, 2023. URL https://arxiv.org/abs/2311.04850.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pp. 268–282, 2018. doi: 10.1109/CSF.2018.00027.

Hanna Yukhymenko, Robin Staab, Mark Vero, and Martin Vechev. A synthetic dataset for personal attribute inference. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2025. Curran Associates Inc. ISBN 9798331314385.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Shenglai Zeng, Yaxin Li, Jie Ren, Yiding Liu, Han Xu, Pengfei He, Yue Xing, Shuaiqiang Wang, Jiliang Tang, and Dawei Yin. Exploring memorization in fine-tuned language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3917–3948, 2024.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual memorization in neural language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. Min-k%++: Improved baseline for pre-training data detection from large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ZGkfoufDaU.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2204–2213, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1205. URL https://aclanthology.org/P18-1205/.

Yuan Zhang, Jason Baldridge, and Luheng He. PAWS: Paraphrase adversaries from word scrambling. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 1298–1308, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1131. URL https://aclanthology.org/N19-1131/.

Andy Zou, Long Phan, Justin Wang, Derek Duenas, Maxwell Lin, Maksym Andriushchenko, Rowan Wang, Zico Kolter, Matt Fredrikson, and Dan Hendrycks. Improving alignment and robustness with circuit breakers, 2024. URL https://arxiv.org/abs/2406.04313.

Appendix

Table of Contents

A Perturbations **22**
- A.1 Relevant Background in Law and Policy . . . 22
- A.2 List of Datasets . . . 23
- A.3 Inserting Perturbations . . . 25
- A.4 Details of Decontamination . . . 25
- A.5 Data Preparation for Paraphrase Runs . . . 26

B Training **27**
- B.1 Setup . . . 27
- B.2 GPU Hours . . . 27

C Model **27**
- C.1 Architecture Design and Configs . . . 27
- C.2 More General Evaluations . . . 27

D Domain-specific results **28**
- D.1 Copyright-specific Results . . . 28
- D.2 Privacy-specific Results . . . 30
- D.3 Test set Contamination Results . . . 34

E Additional Results **36**
- E.1 Timing and Ordering . . . 36
- E.2 Paraphrased Runs . . . 36
- E.3 Architecture Runs . . . 37

F Additional MIA Results **37**

G Full unlearning results and configurations **37**
- G.1 Grid Search Configurations . . . 37
- G.2 Full Unlearning Results . . . 39
- G.3 Full ROC plots . . . 46

H Additional Plots **48**

A Perturbations

A.1 Relevant Background in Law and Policy

Copyright. Training LLMs presents new challenges for copyright law (Franceschelli & Musolesi, 2022; Henderson et al., 2023; Lee et al., 2024). LLM training requires vast amounts of textual data, much of which is collected from the web and protected by copyright (Longpre et al., 2024). In the U.S., whether training LLMs is a *fair use* of copyrighted material remains uncertain and its legality will be determined by ongoing litigation (Lee, 2024; U.S. Copyright Office, 2025). In the EU, the text and data mining exceptions need further clarification for LLM training as well (e.g. on how to respect user opt-out requests, Lucchi, 2025). On the question of whether training LLMs on copyrighted material should be allowed, copyright law will need to avoid blunt "yes" or "no" answers and make nuanced decisions about the technology to balance innovation and authors' rights.

More nuanced legal decisions could be made on the basis of LLM memorization. On fair use, Lemley & Casey (2020) has previously argued for *fair learning* and that AI training on copyright materials could be fair if the models mainly learn non-expressive elements from copyrighted material. LLMs are capable of memorizing some expressive elements and even reproducing training data verbatim, depending on how it was trained (Cooper & Grimmelmann, 2025). Understanding how training decisions affect memorization and adopting "fair" training techniques will be important for companies to address copyright risks (Sag, 2023; Wei et al., 2025). In the longer term, standardizing what training practices constitutes fair learning can guide the development of safe harbors, which provide legal protections from liability if certain precautions are taken (as proposed in Wei et al., 2024).

Privacy. Web-scale datasets will include personal information, and training LLMs on this data raises privacy concerns (Solove & Hartzog, 2024). Even when personal information is public, people maintain expectations of privacy over their information when it is repurposed (Nissenbaum, 2004; Brown et al., 2022). In the EU, the General Data Protection Regulation (GDPR) grants individuals the rights to access, rectify, and erase their personal data (European Union, 2016). Processing publicly available data is not exempt from the GDPR, but this processing is still allowed if certain legal bases are satisfied, such as a *legitimate interest* in the data Kamara & De Hert (2018). While the U.S. lacks a comprehensive federal privacy law, sector-specific statutes and state-level frameworks (e.g., the California Consumer Privacy Act, State of California, 2018) grant similar rights.

Even where privacy rights are formally recognized, defining rectification or erasure of personal information from LLMs is not straightforward and technically difficult (Cooper et al., 2024). Ideally, sensitive personal data would not be used train models (Hong et al., 2025). In practice, privacy law balances commercial interests against privacy rights, and hard decisions are made when there are no good technical options (e.g., abandoning an algorithm in extreme cases Johnson et al., 2024). Better technical tradeoffs motivates areas of research like differential privacy Near et al. (2023), and understanding LLM memorization enables better design of unlearning and editing methods (Bourtoule et al., 2021; Meng et al., 2022), which could expand the set of feasible regulatory options.

Test sets. The validity of LLM evaluation results can be compromised if test sets are made available online and included in the training corpus (Jacovi et al., 2023). Models may appear to perform better on test sets not because they learn to generalize, but because they appeared in training and were memorized (Magar & Schwartz, 2022). The U.S. Federal Trade Commission enforces against unfair or deceptive practices under its consumer protection authority and has recently pursued cases involving deceptive AI claims (Federal Trade Commission, 2024). The FTC has focused on egregious scams and the scientific issues such as benchmark contamination are likely out of scope. However, benchmarks are scientifically important as they set the direction of research and are used as indicators of the field's progress (although their construct validity is often criticized, see Ethayarajh & Jurafsky, 2020; Raji et al., 2021). The study of LLM memorization can enable methods that detect contamination or measure performance in the presence of contamination.

A.2 List of Datasets

Passages

- **Gutenberg Popular** are passages sampled from the popular books from the Gutenberg corpus (Gerlach & Font-Clos, 2018). Due to studies like Kirchenbauer et al. (2024) which show pretraining data density affects memorization, we stratify two Gutenberg splits based on download counts. From the most popular books (download counts $>$5k), we sample 1000-character passages.
- **Gutenberg Unpopular** are sampled passages from the unpopular books from the Gutenberg corpus (Gerlach & Font-Clos, 2018). From the least popular books with download counts $<$100 and at least 30k words long, we sample 1000-word passages.
- **Wikipedia** are passages sampled from our crawl of Wikipedia articles. We begin our crawl at the Wikipedia pages "2023" and "2024", and to reduce the chances of contamination we only visit pages that were written after the DCLM cutoff date. After filtering out articles without text (e.g. lists), we end up with 1500 articles. We sample 1000 character passages without replacement from these articles, sampling more passages if the document is longer.

Paraphrases

Table 2: **Number of duplicates inserted per perturbation dataset.**

	0	1	4	16	64	256
Copyright						
Gutenberg Popular	400	400		200		80
Gutenberg Unpopular	4000	1428	1428	714	286	143
Wikipedia	759	759	759	379	152	76
MRPC	1950	696	696	348	139	70
PAWS	3538	1263	1263	632	253	126
Privacy						
YAGO	2500	893	893	446	179	89
ECtHR	469	469		235		94
PersonaChat	2000	714	714	357	143	72
Testset Contamination						
PopQA	4000	1429	1429	714	286	143
WinoGrande Infill	4000	1429	1429	714	286	143
WinoGrande-MCQ	4000	1429	1429	714	286	143
MMLU	4000	1429	1429	714	286	143
HellaSwag	4000	1429	1429	714	286	143
Piqa	]4000	1429	1429	714	286	143
MUNCH	269	269		135		54
Ellie	212	212		106		43

- **MRPC** (Dolan & Brockett, 2005) are paraphrases where the source sentences are drawn from news articles. For each pair of paraphrased sentences, we randomly select one to be a part of the perturbation set. During evaluation, we measure whether the models demonstrate a consistent preference for the inserted paraphrase.
- **PAWS** (Zhang et al., 2019) is a dataset of paraphrases generated by rule-based word swaps and backtranslation. The source sentences are deried from Quora questions and Wikipedia pages. Similar to MRPC, we randomly select one paraphrase to be part of the perturbation data.

Biographies

- **YAGO**: We synthetically generate biographies of fictional people using probability distributions inferred from YAGO (Pellissier Tanon et al., 2020), a real-world knowledge graph. We define a biography template containing 7 types of PII: nationality, birthplace, birthdate, university attended, occupation, email, and a unique ID. To create the biographies using the realistic distributions of attributes from YAGO, we sample a nationality and then successively sample each PII conditioned on the previous set. We will release scripts for generating the biographies and the resulting perturbation data. Through these biographies, we can measure memorization on different types of PII, some of which are correlated (e.g, can an LLM infer a person's birthplace given their nationality?).
- **ECtHR** (Pilán et al., 2022) dataset is a text anonymization benchmark based on a collection from European court records annotated to label personally identifiable information. We use a subset of the sections in the record to create a biography for the applicant (the person who is appearing before the court) and use this biography in our perturbation set. In Hubble, this perturbation set serves as a case study for PII reconstruction based on the memorization of real-world biographies.

Chats

- **Personachat** (Zhang et al., 2018) is a dataset where two annotators are asked to engage in a conversation based on the personas assigned to them. We edit the chat logs in the dataset and replace the username of the first speaker with the generic name `chatbot`. We treat the assigned persona of the second speaker as the target private information to be inferred. We insert the modified chat logs as perturbation data. To evaluate indirect PII leakage, we measure whether the models can associate the usernames (seen in the memorized chats) with the private personas (never explicitly revealed to the Hubble models during training).

Standard test sets

- **PopQA** (Mallen et al., 2023) is an open-ended question answering dataset that evaluates the world knowledge of a model. As perturbation data, we insert questions followed by the answer. The standard evaluation compares the generated answer to the target answer for exact match / F1 word overlap.
- **Winogrande-Infill** perturbation set is a subset of WinoGrande (Sakaguchi et al., 2021), a binary multiple choice pronoun resolution task where the model is given a context and asked to determine which entity a pronoun refers to. Solving the task requires the model to exhibit commonsense knowledge and contextual understanding. The examples in WinoGrande are given as a sentence with a blank and two choices. We insert the sentence with the blank filled in with the correct answer. Examples in WinoGrande are designed to have minimal pairs; we ensure that only one example from each pair is used in the perturbation data.
- **Winogrande-MCQ** is a second perturbation set also constructed from WinoGrande (Sakaguchi et al., 2021). Instead of posing the problem in the standard format, we instead frame the problem as an MCQ problem by using the sentence with the blank and the two choices as a query. We insert the query followed by the correct answer in the corpus. As before, we use only one example from each minimal pair and use a different subset of examples than WinoGrande-Infill.
- **MMLU** (Hendrycks et al., 2021) is a 4-way multiple choice question answering dataset that covers 57 different domains and tasks, evaluating both world knowledge and problem-solving capabilities. To create the perturbation data, we format each example using the standard evaluation prompt and append the answer to it.
- **HellaSwag** (Zellers et al., 2019) is a 4-way multiple choice commonsense reasoning dataset, where the model is required to understand implicit context and common knowledge in order to correctly select the continuation to a context. Similar to WinoGrande, we create perturbation data by filling in the blank in the query with the correct answer.
- **PIQA** (Bisk et al., 2020) is a binary multiple choice question answering dataset that requires the model to use physical commonsense reasoning to answer correctly. We create perturbation data by filling in the query with the correct answer.

New test sets

- **ELLie** (Testa et al., 2023) tests the language model's understanding of ellipsis. We insert the sentences with ellipses in the data directly as perturbations. For evaluation, we use the GPT prompt format defined for each example.
- **MUNCH** (Tong et al., 2024) tests a language model's ability to differentiate between apt and inapt usage of synonyms in a sentence. For each example, we choose one sentence with ”apt” usage of the word for insertion in the corpus. We choose one sentence with ”inapt” synonym usage and retain the pair of sentences for evaluation.

A.3 Inserting Perturbations

A visualization of the insertion process is in Figure 4. For each perturbation type, we sought to (1) insert different levels of duplications to induce a range of memorization and (2) duplicate enough examples at each level to achieve precise memorization estimates for that level. Based on initial experiment of 1B models, we find the range of duplications $\{0, 1, 4, 16, 64, 256\}$ to induce a range of memorization. For smaller datasets, we only duplicate powers of 16, up to 256. For the 0 and 1 duplicate levels, we aimed to insert more than 1000 examples, which yields small error bars. At the highest duplication level (256), we typically insert only 1/10th of examples at the lowest duplication level. When an example is highly duplicated and strongly memorized, there is typically low entropy in the model predictions so the resulting error bars over less examples are still small.

A.4 Details of Decontamination

To ensure reliable duplication counts in our analysis, we decontaminate the documents and perturbation data in two phases, depending on the length of the perturbations. For *longer perturbations* (more than 10 tokens), we decontaminate the training data. We build an Infini-gram index (Liu et al., 2024a), enabling fast queries for exact matches over all training documents. Here, we query

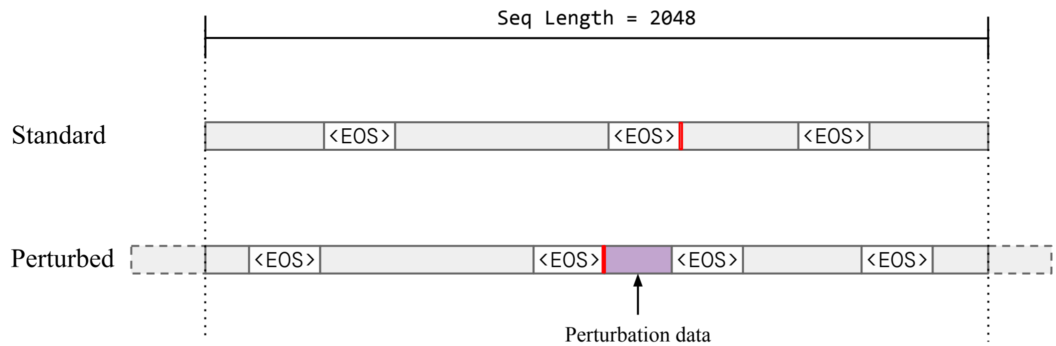

Figure 4: Visualization of inserting a perturbation. First, we sample a training sequence from the standard model to be perturbed. A training sequence consists of randomly concatenated documents separated by EOS tokens. To perturb it, we sample a gap (denoted in red) between the documents and splice the perturbation into a training sequence (between two existing documents). Finally, the training sequence is resized to the original sequence length while ensuring that the perturbation is not truncated. Each perturbation is surrounded by EOS tags and matches other documents. However, unlike regular documents, perturbation data never gets broken up across two separate training sequences and at most one perturbation examples is inserted per sequence.

Table 3: **Percentage of training data modified by duplicated perturbation data.** These calculations depend on the selected sequence length of 2048 tokens and training batch size of 1024 sequences.

Pre-Training Corpus Size	% Tokens Modified	% Sequences Modified	Avg. Perturbations per Batch
100B	0.08%	1.67%	17
500B	0.016%	0.34%	3.4

and remove documents with more than 20-gram overlaps (similar to Brown et al., 2020). The threshold is chosen conservatively to avoid spurious matches and identify duplicated test sets. For *short perturbations* (fewer than 20 tokens), removing matching training documents risks discarding too many documents. Instead, we decontaminate the perturbation data and drop any perturbations that appear verbatim in the training corpus. We validate this two-step process by monitoring the number of documents discarded and manually verifying the matches found.

A.5 Data Preparation for Paraphrase Runs

We construct paraphrased variants of the YAGO biographies and MMLU test set with `gpt-4.1-mini`. Unless otherwise noted, generation uses `temperature=1` and `top_p=1`. For each original perturbation example to be inserted, we obtain as many paraphrases as its required duplication count.

MMLU paraphrases. We follow the paraphrasing instruction of Yang et al. (2023). When a paraphrase query is declined by `gpt-4.1-mini` API's safety filter, we use `gemini-2.5-flash-lite` with the same parameters.

YAGO paraphrases. We adopt the diverse-style watermarking generation instructions from Cui et al. (2025). Each paraphrase is checked with a string-matching validator to ensure all biographical attributes are preserved. A paraphrase is accepted only if every attribute appears. We follow the procedure until we obtain the required number of valid paraphrases.

B TRAINING

B.1 SETUP

Computing infrastructure. Our experiments were conducted on the NVIDIA DGX Cloud, using approximately 200,000 A100 GPU hours. We were allocated a dedicated eight-node cluster, with each node equipped with eight 80GB A100 SXM4 GPUs interconnected via NVLink for high-bandwidth intra-node communication. Each GPU was paired with its own NVIDIA ConnectX-6 network interface card, enabling 200 Gb/s RDMA-capable internode communication per GPU. The cluster was backed by 80TB of shared Lustre storage. Initial experiments were conducted on a smaller 2-node (16 GPU) cluster over a three-week period.

Training setup. Models are trained with GPT-NeoX (Andonian et al., 2023), a pre-training library based on Megatron-LM (Shoeybi et al., 2019) augmented with DeepSpeed and other optimization techniques. All models use a global batch size of 1024 with sequence length 2048. Training begins with a learning rate of 4e-4, decays to a minimum of 4e-5, and is annealed according to a cosine schedule with a warmup fraction of 0.01 for 500B-token runs and 0.05 for 100B-token runs. The Adam optimizer was set with β values of 0.9 and 0.95 and with $\epsilon = 1\text{e-}10$. Gradient clipping is set to 1.0 and weight decay to 0.1. Stage 1 ZeRO optimization (Rajbhandari et al., 2020) is enabled during training. Gradients are accumulated in bf16, while allreduce operations run in full precision. Further details are listed in the config file in Appendix C. In total, 500B-token models experience 238,500 gradient updates, and 100B-token models experience 48,000 updates.

B.2 GPU HOURS

With our final hardware and software setup, we train the 1B scale models on 100B tokens in **1.13k GPU-hours** (approx. 35.5 hrs in wall clock time using 32 GPUs). We train the 8B-scale models on 100B tokens in **7.6k GPU-hours** (approx. 119 hrs in wall clock time using 64 GPUs).

C MODEL

C.1 ARCHITECTURE DESIGN AND CONFIGS

The Hubble models are based on the Llama 3 architecture (Grattafiori et al., 2024). Specifically, the 1B parameter models are based on the Llama-3.2-1B architecture, and the 8B models are based on the Llama-3.1-8B. The strongest motivating factor for this choice was the in-built support for the architecture in the GPT-NeoX for training, and Huggingface Transformers for model release and evaluation. We list the model hyperparameters in Table 4.

C.2 MORE GENERAL EVALUATIONS

We evaluate the general capabilities of our trained models using two evaluation suites: Pythia and DCLM.

We report zero-shot and 5-shot performance of the (standard) Hubble models on the suite of tasks used by the Pythia team (Biderman et al., 2023) in Tables 5 and 6. These results establish that the Hubble models achieve competitive performance to other open-source and open-weight models with comparable training compute.

Additionally, we compare the Hubble models to other models trained specifically on the DCLM corpus. We run DCLM v1 evaluations using the official competition repository (Li et al., 2024a) and report those results in Table 7. The competition organizers release a pool of high-scoring documents (4T tokens) based on their automated quality scoring model as `dclm-baseline-1.0`. They use the subset of documents with the *highest* scores to train their official DCLM-BASELINE models. Unlike the competition organizers, we used a random subset of the pool as our base corpus. Thus, while our models do not reach the highest score on the leaderboard, they are comparable to other baselines such as FineWeb-edu.

Table 4: **Hubble model configurations.**

	Hubble 1B	Hubble 8B
Dimension	2048	4096
Num Heads	32	
Num Layers	16	36
MLP Dimension	8192	14336
Layer Norm	RMSNorm	
Positional Embeddings	RoPE	
Seq Length	2048	
Attention Variant	GQA	
Num KV Heads	8	
Biases		
Block Type	Sequential	
Activation	SwiGLU	
Batch size (instances)	1024	
Batch size (tokens)	∼2M	
Weight Tying	No	
Warmup Ratio	5% for 100B tokens, 1% for 500B tokens	
Peak LR	$4.0E-04$	
Minimum LR	$4.0E-05$	
Weight Decay	0.1	
Beta1	0.9	
Beta2	0.95	
Epsilon	$1.0E-08$	
LR Schedule	cosine	
Gradient clipping	1.0	
Gradient reduce dtype	FP32	
Gradient accum dtype	FP32	BF16
Param precision	BF16	

D Domain-specific results

D.1 Copyright-specific Results

We report additional evaluations on the **Passages** sub-domain in Figure 5 and **Paraphrases** sub-domain in Figure 6. For Passages, beyond the loss-based evaluations in the main paper, we assess verbatim memorization by conditioning on the first 50 tokens and comparing the generated continuation (first 100 tokens) to the original passage using exact match and Rouge-L. For Paraphrase evaluations, we measure accuracy based on loss-based choice, i.e., we measure the likelihood assigned by the model to the two sentences in a pair and check if the inserted paraphrase has a higher likelihood. Results are reported with and without length-based normalization of the log-likelihood; we find that normalization has little effect on the overall scaling and dilution trends.

The strength of memorization of passages is source dependent. Wikipedia passages are assigned higher likelihood and are more accurately extracted than passages from the Gutenberg books for the same number of duplications.

Popular and unpopular books are memorized similarly at the 1B scale with a minor preference for the popular books under the 8B model. We had expected that popular books from Gutenberg would be preferentially memorized (with higher likelihood and higher extraction accuracy) for the same number of duplicates compared with the unpopular books. This intuition was based on the data density hypothesis (Kirchenbauer et al., 2024); the content of popular books is more likely to be discussed in web text than unpopular books. There is no noticeable difference at the 1B parameter scale. Even at the 8B parameter scale, there is a very small increase in the generative extraction of passages from popular books compared to unpopular books. The 8B param model with 100B tokens obtains a ROUGE-L of 30% on popular books compared to 28% on unpopular books duplicated 16

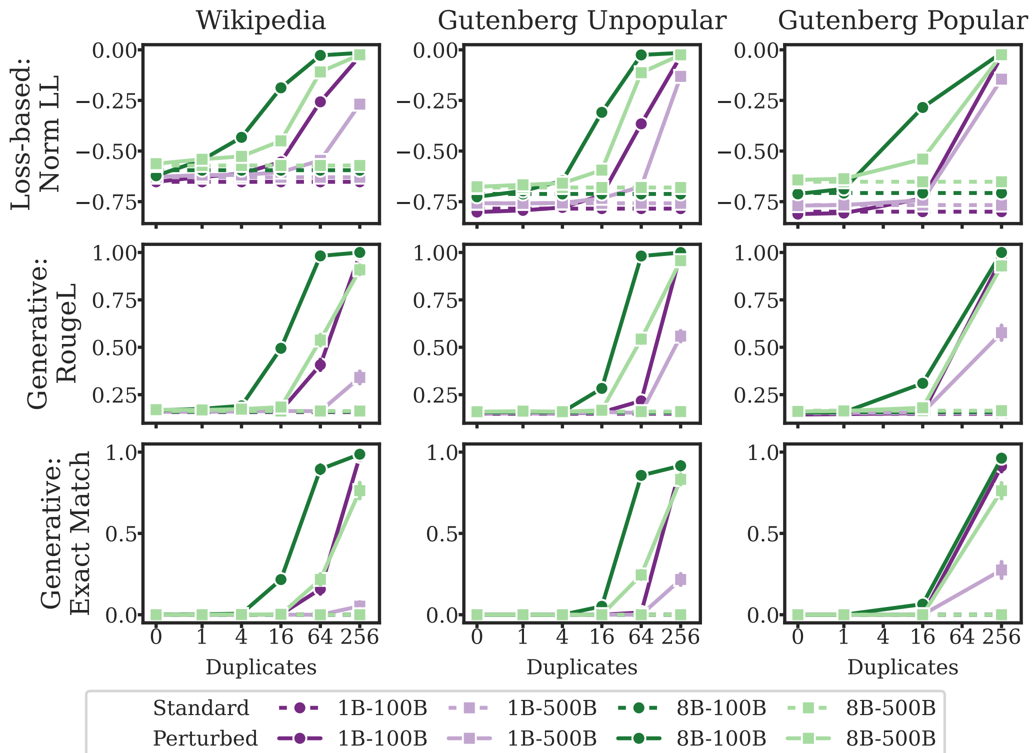

Figure 5: **Core results on Copyright Passages.** The first row evaluates memorization with the length-normalized log-likelihood of the models on the passages. The lower two rows measure the accuracy of verbatim generation, where the models are prompted to generate a 100-token continuation given a 50-token prefix.

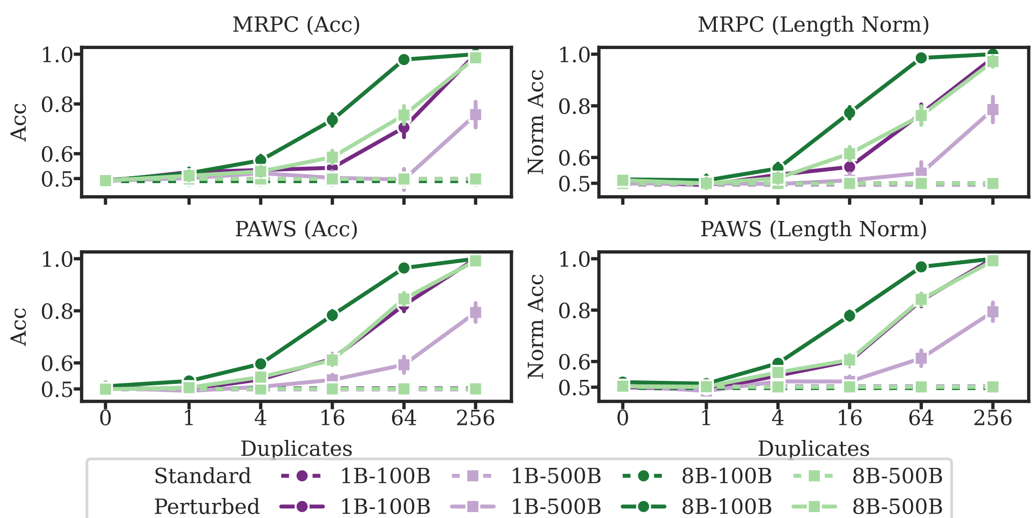

Figure 6: **Core results on Copyright Paraphrases.** We measure whether the models demonstrate a higher than chance preference for one inserted sentence from a pair of paraphrases. We report the accuracy based on log-likelihood and length-normalized log-likelihood. Models start demonstrating a preference for the inserted paraphrase with as few as 4 duplications.

Table 5: **Zero-shot benchmark results using the Pythia suite.** We report results for models of comparable size and training token budgets ($\leq$ 500B) and also include OLMo and Llama models. We use the same evaluations as the Pythia suite and run them through EleutherAI's Language Model Evaluation Harness (Gao et al., 2023).
*Token Count is based on numbers reported in the corresponding model's release notes and may use different tokenizers
#Winogrande and PIQA train sets are inserted in the perturbed HUBBLE corpus.

Model	Token Count*	ARC Challenge	ARC Easy	LogiQA	Lambada (OpenAI)	PIQA#	SciQ	Wino -Grande#	WSC
				1B-Scale					
Hubble-1B	500B								
-Standard		0.37	0.66	0.27	5.45	0.76	0.85	0.62	0.38
-Perturbed		0.37	0.67	0.27	5.50	0.77	0.87	0.58	0.58
Hubble-1B	100B								
-Standard		0.33	0.61	0.28	6.84	0.73	0.84	0.58	0.63
-Perturbed		0.33	0.60	0.28	7.00	0.72	0.84	0.55	0.51
Pythia 1B	300B	0.27	0.49	0.30	7.92	0.69	0.76	0.53	0.37
Pythia 1.4B	300B	0.28	0.54	0.28	6.08	0.71	0.79	0.57	0.37
Bloom 1.1B	366B	0.26	0.45	0.26	17.28	0.67	0.74	0.55	0.37
Bloom 1.7B	366B	0.27	0.48	0.28	12.59	0.70	0.77	0.57	0.37
OPT 1.3B	180B	0.30	0.51	0.27	6.64	0.72	0.77	0.60	0.38
OLMo-2-1B	4T	0.42	0.74	0.30	5.19	0.76	0.95	0.65	0.41
Llama-3.2-1B	~9T	0.37	0.60	0.30	5.74	0.74	0.89	0.60	0.35
				~ 8B-Scale					
Hubble-8B	500B								
-Standard		0.52	0.80	0.31	3.23	0.80	0.94	0.72	0.36
-Perturbed		0.51	0.78	0.31	3.23	0.79	0.94	0.73	0.42
Hubble-8B	100B								
-Standard		0.45	0.74	0.29	3.95	0.79	0.92	0.66	0.56
-Perturbed		0.43	0.70	0.27	3.94	0.76	0.93	0.61	0.37
Pythia 6.9B	300B	0.35	0.61	0.30	4.45	0.77	0.84	0.60	0.37
OPT 6.7B	180B	0.35	0.60	0.29	4.25	0.76	0.85	0.65	0.42
OLMo-2-7B	4T	0.57	0.83	0.31	3.37	0.81	0.96	0.75	0.67
Llama-3.1-8B	15T+	0.53	0.81	0.31	3.13	0.81	0.95	0.73	0.63

times. The 8B parameter models trained on 100B and 500B tokens both assign a slightly higher likelihood to passages from the popular books.

D.2 PRIVACY-SPECIFIC RESULTS

D.2.1 BIOGRAPHIES - DIRECT PII LEAKAGE

For the Biography sub-domain, we not only care about the memorization of the biographies (evaluated through loss as with copyright domain) but also the ease of reconstruction of sensitive information about the persons. For direct memorization, we report the loss assigned by the model to the inserted biography. To evaluate the ease of PII reconstruction, we instantiate attacks with varying strength. **Weak attacks** assume that the attacker already knows PII about the person of interest and is seeking a few missing facts. **Strong attacks** assume that the attacker knows less sensitive information about the person of interest, with our strongest attacks assuming that the attacker only knows the name. We instantiate **loss-based choice attacks** where the attacker has narrowed down the possible values of the missing PII. We frame the attack as MCQ problems and check which candidate answer has the highest likelihood when plugged into the blank. When the attacker has no way to deduce the set of candidate answers, they have to use **generative attacks** where the model is prompted to fill in the blank. We evaluate generative attack with either *Word Recall*, which scores if the answer entity occurs anywhere in the generated response, or *Prefix Match*, which scores whether the model generation starts with the answer entity. Table 8 lists the attacks that we instantiate. The

Table 6: **Five-shot benchmark results using the Pythia suite.** Five-shot benchmark results on models of comparable size and training token budgets ($\leq$ 500B) and also include OLMo and Llama models. We use the same evaluations as the Pythia suite and run them through EleutherAI's Language Model Evaluation Harness (Gao et al., 2023).
*Token Count is based on numbers reported in the corresponding model's release notes and may use different tokenizers.
#Winogrande and PIQA train sets are inserted in the perturbed HUBBLE corpus.

Model	Token Count*	ARC Challenge	ARC Easy	LogiQA	Lambada (OpenAI)	PIQA#	SciQ	Wino -grande#	WSC
1B-Scale									
Hubble-1B	500B								
-Standard		0.40	0.72	0.25	7.43	0.76	0.95	0.63	0.41
-Perturbed		0.40	0.72	0.25	7.23	0.76	0.94	0.63	0.45
Hubble-1B	100B								
-Standard		0.36	0.69	0.24	9.31	0.74	0.92	0.59	0.43
-Perturbed		0.36	0.67	0.25	8.95	0.75	0.92	0.59	0.38
Pythia 1B	300B	0.28	0.57	0.25	10.86	0.70	0.92	0.53	0.43
Pythia 1.4B	300B	0.31	0.62	0.27	8.03	0.71	0.92	0.58	0.57
Bloom 1.1B	366B	0.28	0.53	0.25	24.84	0.68	0.90	0.53	0.37
Bloom 1.7B	366B	0.29	0.57	0.28	15.40	0.69	0.92	0.58	0.39
OPT 1.3B	180B	0.30	0.60	0.26	8.01	0.71	0.92	0.59	0.57
OLMo-2-1B	4T	0.46	0.76	0.27	6.26	0.77	0.96	0.66	0.45
Llama-3.2-1B	∼9T	0.38	0.70	0.27	7.09	0.76	0.95	0.62	0.43
∼ 8B-Scale									
Hubble-8B	500B								
-Standard		0.58	0.84	0.32	3.71	0.82	0.98	0.77	0.56
-Perturbed		0.57	0.83	0.30	3.74	0.82	0.97	0.79	0.58
Hubble-8B	100B								
-Standard		0.47	0.78	0.27	4.61	0.79	0.96	0.67	0.39
-Perturbed		0.48	0.78	0.28	4.66	0.80	0.96	0.71	0.47
Pythia 6.9B	300B	0.39	0.71	0.28	5.65	0.77	0.95	0.64	0.51
OPT 6.7B	180B	0.37	0.70	0.28	4.98	0.77	0.94	0.66	0.54
OLMo-2-7B	4T	0.63	0.85	0.34	3.90	0.81	0.97	0.77	0.78
Llama-3.1-8B	15T+	0.58	0.85	0.33	3.93	0.82	0.98	0.77	0.63

synthetic YAGO biographies allow us to instantiate each of the attacks listed in the table. We can only instantiate the *full prefix, generative* attack for ECtHR since the entity types are not clearly defined (e.g., dates can refer to birth dates or event dates) and not all entity types are always present in the biography. Figures 7 and 8 report attack success rates on ECtHR and YAGO perturbation sets, respectively. Figure 9 provides a breakdown by PII type for reconstruction attacks on YAGO biographies (rows are arranged in the order that the PII type occurs in the biography).

PII leakage depends on attack format. For both ECtHR (Fig 7) and YAGO (Fig 8), the weakest attacks (*full prefix* and *full prefix-full suffix*) are very effective in reconstructing PIIs with high accuracy. Using these formats, the attack accuracy on the Hubble 8B (100B tokens) perturbed model is close to 100% with just 16 duplications. The attack success rate decreases when considering strong attack scenarios. Compared to the full-prefix attack, the accuracy of the reconstruction decreases when the attacker uses formats with less known PII (e.g. name only). Using the strongest attack scenario (generative attack with *name only*), the attacker is only able to reconstruct PIIs with 25% accuracy even on the highly duplicated data.

For strong attack prompts, attack success decreases for PII that occurs later in the biography. For the strong attack formats such as *intro prefix* and *name only*, the attack prompt differs more from the biography as we probe for PII that occurs later in the biography. From Figure 9, we see

Table 7: Models evaluated on the DCLM v1 eval suite. DCLM-BASELINE and FineWeb edu results are copied from the official DCLM leaderboard. In general, Hubble models perform on par within their respective data and model scales.

Model	Params	Tokens	FLOPS	CORE	MMLU	EXTENDED
1B-Scale						
DCLM-BASELINE	1.4B	28.8B	2.4e20	30.2	23.8	15.4
FineWeb edu	1.8B	28B	3.0e20	26.6	26.3	13.5
DCLM-BASELINE	1.4B	144B	1.2e21	36.1	26.4	18.6
FineWeb edu	1.8B	140B	1.5e21	33.8	25.5	17.6
Pythia 1B	1B	300B	1.8e21	24.8	25.1	13.5
Pythia 1.4B	1.4B	300B	2.5e21	27.8	25.4	14.2
Hubble 1B	1.2B	100B	7.2e20	27.8	24.9	14.5
Hubble 1B	1.2B	500B	3.6e21	34.2	25.7	17.7
$\sim$ 8B-Scale						
DCLM-BASELINE	6.9B	138B	5.7e21	44.8	42.2	28.8
FineWeb edu	7B	138B	5.8e21	38.7	26.3	22.1
OPT 6.7B	6.7B	180B	7.2e21	35.6	25.2	18.8
DCLM-BASELINE	6.9B	276B	1.1e22	48.9	50.8	31.8
FineWeb edu	7B	276B	1.2e22	41.9	37.4	24.5
Pythia 6.9B	6.9B	300B	1.2e22	35.7	25.4	19.6
Hubble 8B	8.3B	100B	5.0e21	40.8	28.0	22.0
Hubble 8B	8.3B	500B	2.5e22	50.0	53.9	34.6

Table 8: **PII Attack Defitions.** The instantiated PII reconstruction attacks are listed below in increasing order of strength (fewer additional PII known to the attacker). For each format, we illustrate the attacker's query to reconstruct the target's university using a sample biography from the YAGO perturbations. The *full prefix-full suffix* attack is only compatible with Infill attacks (loss-based choice) since generations cannot be conditioned on the suffix.

Sample YAGO Biography
Dora Sloan is from the United States. Dora was born in Phoenix, Arizona. Dora is an alumni of St. John's College. Dora was born on May 15, 1968. Dora receives email at dora@gmail.com. Dora is a competitive diver. Dora has the unique identifier 4dc0969af29a4324bf5746c50f7209a2.

Prompt Format	**Compatible Attack**	**Example Query**
full prefix-full suffix	Infill	Dora Sloan is from the United States. Dora was born in Phoenix, Arizona. Dora is an alumni of ______. Dora was born on May 15, 1968. Dora receives email at dora@gmail.com. Dora is a competitive diver. Dora has the unique identifier 4dc0969af29a4324bf5746c50f7209a2.
full prefix	Infill , Gen	Dora Sloan is from the United States. Dora was born in Phoenix, Arizona. Dora is an alumni of ______.
intro prefix	Infill , Gen	Dora Sloan is from the United States. Dora is an alumni of ______.
name only	Infill , Gen	Dora Sloan is an alumni of ______.

that attack success rate for the *intro prefix* format decreases as we probe for PII that appears later in the biography. Two exceptions to this are UUID and email.

UUIDs and emails exhibit distinct memorization patterns. We further point out two outliers from Figure 9. Emails can be reconstructed with high accuracy with all our attack formats. We create distractor choices for email using rules such that all candidates have high character overlap with the correct email. Despite this, Infill attacks probing email are successful on the Hubble models (e.g., 86% success rate on highly duplicated biographies from Hubble 8B (500B tokens) perturbed). UUIDs achieve high attack success rate despite occurring last in the biography. Surprisingly, al-

Table 9: **Indirect PII Attack Defitions.** The instantiated PII inference attacks are listed below. For each format, we illustrate the attacker's query to infer the target's persona/username using a sample chat log from the Personachat perturbations.

Sample Personachat conversation		
chatbot: i like acting. i am in a telenovela now. FloodBassoon371: fun. dancing is my ticket to fame. chatbot: what kind of dancing? were you in a show? i love musicals. FloodBassoon371: anything but ...		
Prompt Format	**Example Query**	**Comments**
Norm LL on Chat	*chatbot: i like acting. i am in a telenovela now. FloodBassoon371: fun. dancing is my ticket to fame. chatbot: what kind of dancing? were you...*	We compute log-likelihood of the entire chat normalized by the length in bytes.
Norm LL on Persona	chatbot: tell me a bit about yourself. InquiryTomb530: *i m an amazing dancer. i have blonde hair that reaches my knees. i volunteer at animal shelters...*	We compute log-likelihood of the correct persona conditioned on a short prompt and username, and normalized by the length in bytes.
Infill on Persona	InquiryTomb530: ____	We compare log-likelihood (with different normalizations) of the correct persona against 9 distractor personas conditioned on the username and report accuracy.
(Prompted) Infill on Persona	chatbot: tell me a bit about yourself. InquiryTomb530: ____	Same as Infill on Persona with an additional prompt.
Infill on Username	___*: i m an amazing dancer. i have blonde hair that reaches my knees...*	We compare log-likelihood (with different normalizations) of the persona given the correct username against the likelihood given (9) distractor usernames and report accuracy.
(Prompted) Infill on Username	chatbot: tell me a bit about yourself. ____*: i m an amazing dancer. i have blonde hair that reaches my knees...*	Same as Infill on Username with an additional prompt.

though the UUID can be chosen from a set of candidates with infilling and generated with the full prefix, we are unable to reconstruct it with a name-only prompt. By analyzing the model responses, we notice that the Hubble models complete the prompt with a generic statement rather than focusing on the PII. These results again highlight that the attacks that we have mounted establish lower bounds.

D.2.2 CHATS - INDIRECT PII LEAKAGE

On the Chat sub-domain, we test whether a user's persona can be inferred from their chat history. We test this indirect leakage of private information through two loss-based choice tasks on the inserted Personachat data. In the first task, *Infill on Persona*, we test the models' accuracy on selecting the correct persona conditioned on the username from a set of 10 personas (distractors are drawn randomly from the other personas in the perturbation data). In the second task, *Infill on Username*, we test whether the model can accurately select the correct username given the persona (distractor usernames are randomly drawn from the perturbation data). We illustrate the attacks in Table 9. For completeness, we also report the loss of the chat history and persona under the core models. We report findings in Figure 10.

Models assign lower likelihood to persona when memorizing chats. The log-likelihood assigned to the persona by the Hubble models decreases as the strength of memorization of the chat

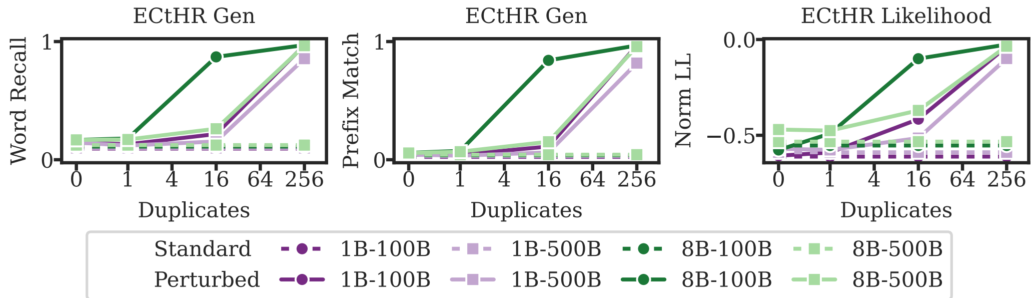

Figure 7: **Core results on ECtHR.** In the first two plots, we report the accuracy of generating the seen PII fact given the preceding biography (full prefix). The rightmost plot reports the length-normalized log-likelihood of the biographies under the models.

history increases (i.e., with lower dilution). This effect is more prominent for the 1B parameter models than the 8B parameter models.

Indirect persona inference is difficult, with success rates below 35% in all attack settings. The accuracy of the Hubble models is close to random guessing when asked to choose between the persona choices given the username (Infill on Persona). Thus, although the Hubble models memorize the chat log for the user, they are unable to infer and assign a higher likelihood to the correct underlying persona. In contrast, personally identifiable information can be inferred when the attack is reversed - prompting the model to identify the username corresponding to a given persona. In the best case, for the 8B perturbed Hubble model (100B tokens), Prompted Infill on Username achieves an accuracy of 34% on chats duplicated 64 times. These results serve to reiterate our warning: our memorization evaluations are a lower bound on what information is memorized and leakable from the models.

D.3 TEST SET CONTAMINATION RESULTS

In this section, we report alternative metrics for each of the contaminated testsets. For **PopQA**, we report F1 score Rajpurkar et al. (2018) in addition to the Exact Match (accuracy). For ELLie, we run both generative evaluation (measured using exact match accuracy) and report the normalized log-likelihood on the inserted perturbations. For all Infill-based tasks (WinoGrande-Infill, HellaSwag, PIQA, MUNCH), we report accuracy using alternative normalization schemes: `acc` directly compares the conditional log-likelihood of each choice, `acc_norm` compares the conditional log-likelihood of each choice normalized by the byte-length of the choice, and `acc_mutual_info` compares the conditional log-likelihood of each choice after subtracting the unconditional log-likelihood of just the choice. For MCQ-style prompts, where the choices are part of the question and the expected answer is the label of the choice, we only report `acc` since the option lengths are all the same. We report the performance on PopQA, HellaSwag, MMLU, and PIQA in Figure 11. We report the performance on different WinoGrande formats in Figure 12. Finally, we report performance on the new test sets, MUNCH and ELLie, in Figure 13.

Standard models demonstrate performance scaling based on model and corpus size. Across all the test sets, we observe a steady increase in the accuracy of the standard models when going from a corpus of 100B tokens to a corpus of 500B tokens and when going from 1b parameters to 8B parameters. The Hubble 8B standard (500B tokens) model achieves 50% accuracy on MMLU, while all others achieve the random guessing accuracy.

Contamination can boost accuracy with very low duplication. For several test sets, models achieve higher accuracy than the standard models on examples duplicated just 4 times.

Contamination can improve, hurt, or leave unchanged within-task generalization. On PopQA, we see that the accuracy of the perturbed models in higher than the standard models even on unseen examples (0 duplicates). On MMLU, we see that the performance on unseen examples is unchanged. However, on Winogrande, HellaSwag, and PIQA, we see that the accuracy on un-

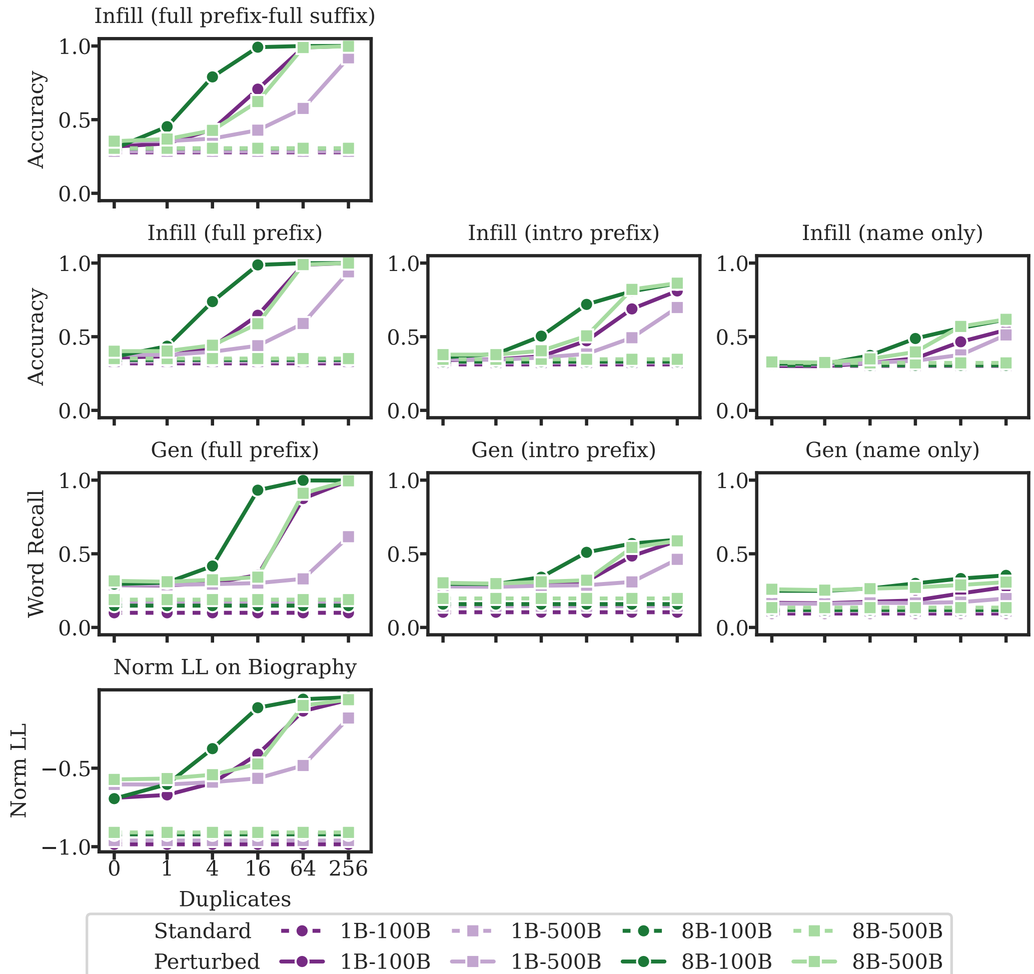

Figure 8: **Core results on YAGO.** Row 4 reports the length-normalized log-likelihood assigned to the biographies under the models. The perturbed models learn to assign higher likelihood to unseen biographies (0 duplicates) by generalizing from the seen synthetic ones.
Rows 1 and 2 report the accuracy of choosing the correct PII from a set of 10 choices (15 choices for emails) of the same entity type. From left to right, each successive attack requires the attacker to know less PII about the person. We see a corresponding decrease in attack success.
Row 3 performs the same attacks as row 2, but evaluates the accuracy of generating the PII rather than choosing from a set of candidates. Generative attacks are less effective than loss-based choice.

seen examples is worse than the accuracy of the standard model. The lack of generalization is also demonstrated with the paraphrase experiments in Appendix E.2, where we find that a perturbed model trained on paraphrased MMLU problems is unable to answer the original questions.

Case study of format dependence on WinoGrande. When preparing the corpus for the perturbed models, we inserted two variants of WinoGrande, one in the standard Infill/cloze format, and the other with MCQ format, where the choices are presented as a part of the question and the model selects the answer. In Figure 12, we report the accuracy of the models when the test time format does not match the inserted format, i.e., for data inserted with Infill format, we test using the MCQ format and vice versa. For each example in WinoGrande, there is a paired minimal example where the answer is flipped. When inserting examples, we make sure to only use one example from each pair as a part of the perturbation data. This allows us to evaluate whether the perturbed models can generalize to the minimal pair from training on the inserted example. Our results on WinoGrande show that the models (1) do not generalize across formats and have worse accuracy on contaminated examples than unseen examples, and (2) do not generalize from the contaminated examples to their corresponding minimal pairs.

MUNCH is solved by standard models. From Figure 13, we see that both standard and perturbed models achieve very high accuracy on MUNCH. Each MUNCH example consists of two sentences, one of which is the original, valid sentence, and the other is modified by swapping one word from the original sentence for an inappropriate synonym. The task is to identify which sentence is meaningful and valid. Our core models are all competent at language modeling and thus can solve the task with high accuracy ($> 96\%$). Even so, we see increased accuracy with perturbed models on the examples that are duplicated more than 16 times.

ELLie examples are minimal pairs making it isolate to disentangle the effect of duplication. ELLie is a task that tests whether language models can understand sentences with ellipsis. From Figure 13, we see that the standard model achieve near 0 accuracy on the task. On the other hand, perturbed models achieve accuracy greater than 50% even on examples that were never duplicated. On further analysis, we realized that the examples in ELLie are minimal pairs.[4] When we insert the examples in our corpus, examples with the same first sentence were put in different duplication bins, e.g., of all the examples with the same core sentence, some examples were sometimes duplicated 0 times and other examples were duplicated 16 times. Thus, we see that models achieve high accuracy on examples duplicated 0 times. This invalidates the use of ELLie for studying dilution.

E Additional Results

E.1 Timing and Ordering

We use the InsertRange models to study forgetting in language models. We run our memorization evaluations on intermediate checkpoints at intervals of 2000 training steps until completion (48000 steps) and record the memorization strength. In Figure 14, we report the normalized log-likelihood on Wikipedia passages inserted 256 times and accuracy on the MRPC paraphrase task on examples inserted 256 times. For all four InsertRange runs, we see norm-likelihood (and accuracy) initially increases as the models are exposed to more duplications, reaches its peak when all the perturbations have been observed, and then starts to decay.

E.2 Paraphrased Runs

We train two perturbed models (1B and 8B parameters) on 100B tokens with the same perturbation data as the core perturbed model but with two data sets paraphrased: MMLU and YAGO Biographies. We evaluate the behavior of the 'paraphrase' models on MMLU and YAGO evaluations in Figure 15 and on all our perturbation evaluations in Figure 24.

PII can be leaked from paraphrased biographies with loss-based choice and generative evaluations. The weakest attacks, which assume that the attacker has access to all PII about a person except one fact, are successful on models trained with paraphrased biographies. However, they have lower effectiveness than extracting the facts from the model that was trained on the original biographies. PII can be extracted with 100% accuracy from the core 8B perturbed model using the full prefix and full suffix MCQ format. This accuracy drops to 89% when extracting PII from the paraphrase model. Surprisingly, when using stronger attacks (attacker has access to only the persons name), PII is more accurately extractable from the 8B model trained on paraphrased biographies compared to the core models. However, this finding depends on the format of the attack and scale; generative evaluations cannot extract PII from the 1B paraphrased model.

Models cannot generalize from paraphrased MMLU to the original examples. We find that both models (1B and 8B parameters) obtain random accuracy on the MMLU MCQ evaluations when trained on paraphrased versions of the examples.

[4] Many examples in ELLie contain the same first sentence but different query sentences (the second sentence). Thus, they passed our deduplication check.

Table 10: **Membership inference performance on YAGO Biographies and MMLU with Hubble 8B Perturbed.** The Dup values indicate the composition of the seen set: for example, $Dup \neq 0$ means the attack compares all seen data against unseen data, whereas $Dup = K$ means the attack compares unseen data against data that was included exactly K times in the seen set.

Evaluation	MIA	Hubble 8B Perturbed (500B tokens)					
		Dup ≠ 0	Dup = 1	Dup = 4	Dup = 16	Dup = 64	Dup = 256
Yago Biographies	Loss	0.692	0.538	0.652	0.897	1.0	1.0
	MinK%	0.692	0.537	0.651	0.896	1.0	1.0
	MinK%++	0.714	0.571	0.686	0.892	0.995	0.983
	ZLib	0.676	0.524	0.633	0.872	1.0	1.0
MMLU	Loss	0.673	0.529	0.628	0.857	1.0	1.0
	MinK%	0.672	0.529	0.626	0.854	1.0	1.0
	MinK%++	0.743	0.58	0.731	0.943	0.994	0.986
	ZLib	0.644	0.523	0.593	0.775	0.993	0.999

E.3 Architecture Runs

We train two 1B parameter models, one deeper architecture with twice the number of layers (32) as the base model (16) and one shallower with half the number of layers (8). We simultaneously adjust the size of the intermediate representation to maintain the number of parameters (exact number of parameters varies but matches 1.2B parameters when rounded). Our findings in Figure 25 show that the deeper architecture memorizes slightly more than the base model and the shallower architecture memorizes less than the base model. The magnitude of the difference between the three architectures is dataset and domain dependent. Moreover, the effect is less prominent than the effect of dilution and ordering discussed previously.

F Additional MIA Results

We instantiate 12 variants of MIA benchmarks using the Hubble suite, using 4 models and 3 perturbation datasets (passages from Gutenberg Unpopular, biographies from YAGO, and contaminated examples from MMLU). As discussed in § 5.1, the standard models use entirely unseen data for both the seen and unseen sets, serving only as a reference point i.e. no method should achieve better-than-random accuracy in this setting.

- Tables 1 and 10 report MIA performance on the Hubble 8B Perturbed model.
- Table 11 reports MIA performance on the Hubble 8B Standard model.
- Table 12 reports MIA performance on the Hubble 1B Perturbed model.
- Table 13 reports MIA performance on the Hubble 1B Standard model.

G Full unlearning results and configurations

G.1 Grid Search Configurations

Below are the detailed hyperparameters for each method:

RMU (Li et al., 2024b):

- Layer Fine-tuning:
  - Layers: 5, 6, 7
- Alpha: 100, 1000, 10000
- Steering coefficient: 5, 50, 500
- Learning rate: `5e-5`, `1e-5`, `5e-4`
- Effective batch size: 4
- Epochs: 4, 8

Table 11: **Membership inference performance on various benchmarks with Hubble 8B Standard**. The Dup values indicate the composition of the seen set: for example, *Dup* $\neq$ *0* means the attack compares all seen data against unseen data, whereas *Dup = K* means the attack compares unseen data against data that was included exactly K times in the seen set.

Evaluation	MIA	Hubble 8B Standard (500B tokens)					
		Dup $\neq$ 0	Dup = 1	Dup = 4	Dup = 16	Dup = 64	Dup = 256
Gutenberg Unpopular	Loss	0.507	0.522	0.486	0.495	0.54	0.545
	MinK%	0.507	0.522	0.486	0.495	0.54	0.545
	MinK%++	0.504	0.517	0.493	0.499	0.484	0.543
	ZLib	0.497	0.514	0.48	0.474	0.535	0.544
Yago Biographies	Loss	0.499	0.489	0.499	0.519	0.486	0.516
	MinK%	0.499	0.489	0.499	0.519	0.487	0.516
	MinK%++	0.503	0.5	0.503	0.507	0.505	0.505
	ZLib	0.495	0.479	0.5	0.523	0.481	0.495
MMLU	Loss	0.502	0.506	0.503	0.512	0.459	0.476
	MinK%	0.502	0.506	0.503	0.512	0.458	0.476
	MinK%++	0.506	0.51	0.505	0.514	0.497	0.45
	ZLib	0.501	0.505	0.504	0.506	0.463	0.495

Table 12: **Membership inference performance on various benchmarks with Hubble 1B Perturbed.** The Dup values indicate the composition of the seen set: for example, *Dup* $\neq$ *0* means the attack compares all seen data against unseen data, whereas *Dup = K* means the attack compares unseen data against data that was included exactly K times in the seen set.

Evaluation	MIA	Hubble 1B Perturbed (500B tokens)					
		Dup $\neq$ 0	Dup = 1	Dup = 4	Dup = 16	Dup = 64	Dup = 256
Gutenberg Unpopular	Loss	0.552	0.52	0.504	0.552	0.73	0.999
	MinK%	0.552	0.52	0.504	0.552	0.729	0.999
	MinK%++	0.575	0.513	0.53	0.605	0.825	1.0
	ZLib	0.543	0.511	0.497	0.533	0.729	1.0
Yago Biographies	Loss	0.606	0.506	0.557	0.696	0.928	1.0
	MinK%	0.606	0.506	0.556	0.695	0.927	1.0
	MinK%++	0.615	0.509	0.565	0.715	0.947	1.0
	ZLib	0.596	0.499	0.551	0.679	0.899	1.0
MMLU	Loss	0.557	0.499	0.524	0.575	0.748	1.0
	MinK%	0.557	0.5	0.524	0.575	0.747	1.0
	MinK%++	0.605	0.522	0.556	0.681	0.887	0.996
	ZLib	0.548	0.502	0.521	0.556	0.67	0.998

- Sample max length: 512

RR (Zou et al., 2024):

- LoRA Fine-tuning:
  - LoRA Rank: 16
  - LoRA α: 16
  - LoRA dropout: 0.05
- LoRRA Alpha: 10
- Target layers: 10, 20
- Transform layers: all
- Learning rate: `5e-5`, `1e-4`, `5e-4`, `1e-3`
- Effective batch size: 8
- Epochs: 4, 8

Table 13: **Membership inference performance on various benchmarks with Hubble 1B Standard.** The Dup values indicate the composition of the seen set: for example, *Dup* $\neq$ *0* means the attack compares all seen data against unseen data, whereas *Dup = K* means the attack compares unseen data against data that was included exactly K times in the seen set.

Evaluation	MIA	Hubble 1B Standard (500B tokens)					
		Dup $\neq$ 0	Dup = 1	Dup = 4	Dup = 16	Dup = 64	Dup = 256
Gutenberg Unpopular	Loss	0.503	0.517	0.484	0.494	0.534	0.531
	MinK%	0.502	0.517	0.483	0.494	0.534	0.531
	MinK%++	0.5	0.509	0.493	0.497	0.481	0.529
	ZLib	0.493	0.509	0.477	0.471	0.529	0.533
Yago Biographies	Loss	0.495	0.488	0.494	0.51	0.494	0.509
	MinK%	0.495	0.487	0.494	0.51	0.494	0.508
	MinK%++	0.5	0.499	0.501	0.494	0.518	0.497
	ZLib	0.494	0.481	0.498	0.516	0.489	0.49
MMLU	Loss	0.502	0.506	0.502	0.519	0.459	0.48
	MinK%	0.503	0.506	0.502	0.519	0.459	0.481
	MinK%++	0.509	0.512	0.509	0.53	0.475	0.448
	ZLib	0.501	0.504	0.503	0.508	0.465	0.494

- Sample max length: 256

SatImp (Yang et al., 2025):

- α: 0.01, 0.1, 1
- β_1: 5, 6
- β_2: 1
- Learning rate: `1e-5`, `5e-5`, `1e-4`
- Effective batch size: 16
- Sample max length: 256

After grid search, we evaluate the unlearned checkpoints on tinyMMLU, tinyWinogrande, and tiny-Hellaswag from TinyBenchmarks (Polo et al., 2024) for general capabilities preservation, and discard checkpoints with average performance degradation exceeding 10%.

G.2 Full Unlearning Results

We provide the full scale unlearning results for Gutenberg in Figure 19 and YAGO in Figure 20.

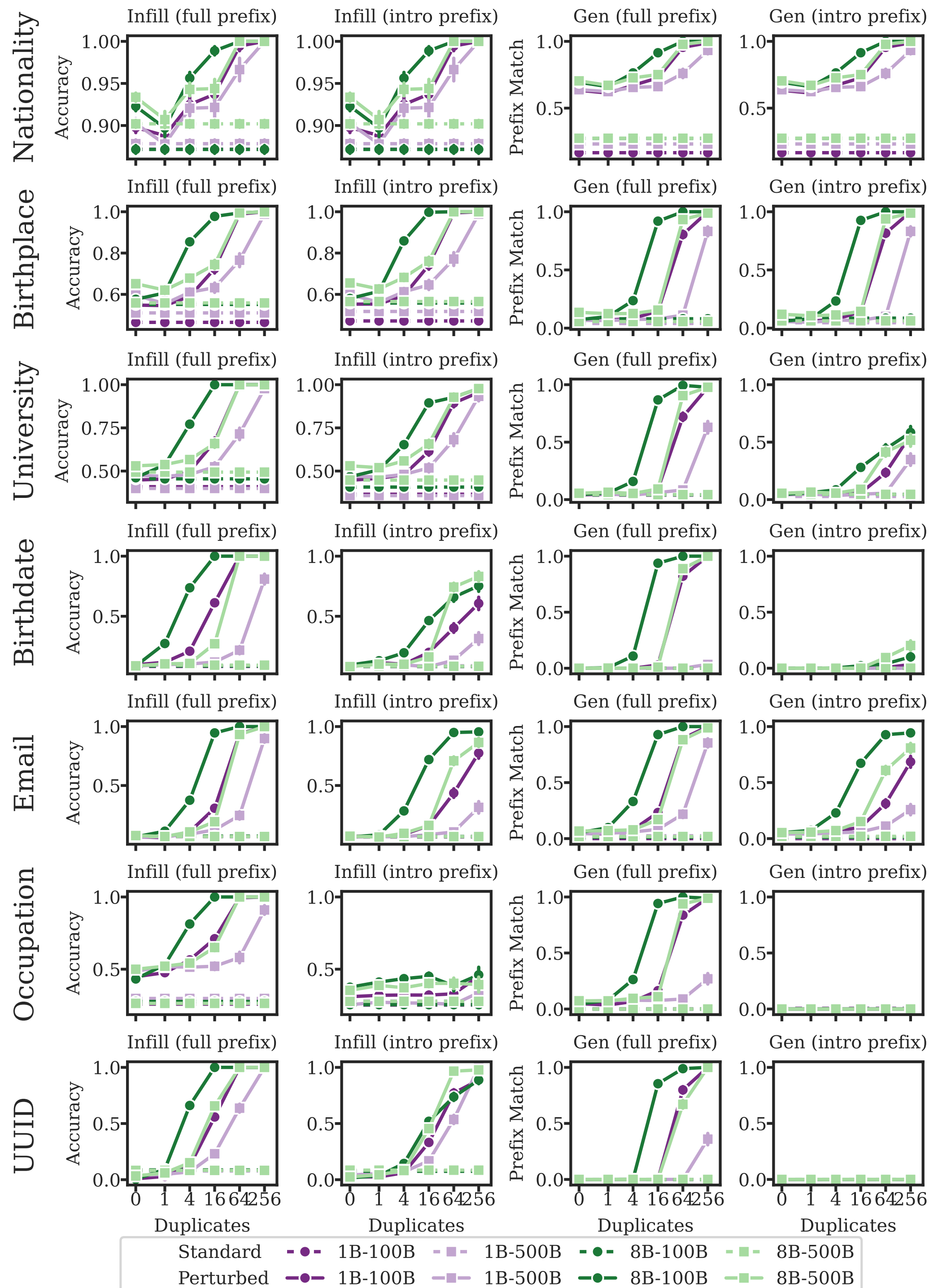

Figure 9: **Core results on YAGO (PII-type based).** Rows display attack success rates for each PII type, arranged by where the PII appears in the synthetic biography. Columns 1 and 2 report the accuracy of choosing the correct PII from a set of candidates. Columns 3 and 4 report the accuracy of generating the correct PII (evaluated by whether the correct answer is generated as the prefix of the model response). Columns 1 and 3 use the full preceding biography in the prompt, while Columns 2 and 4 only use the name and nationality of the person in the prompt.

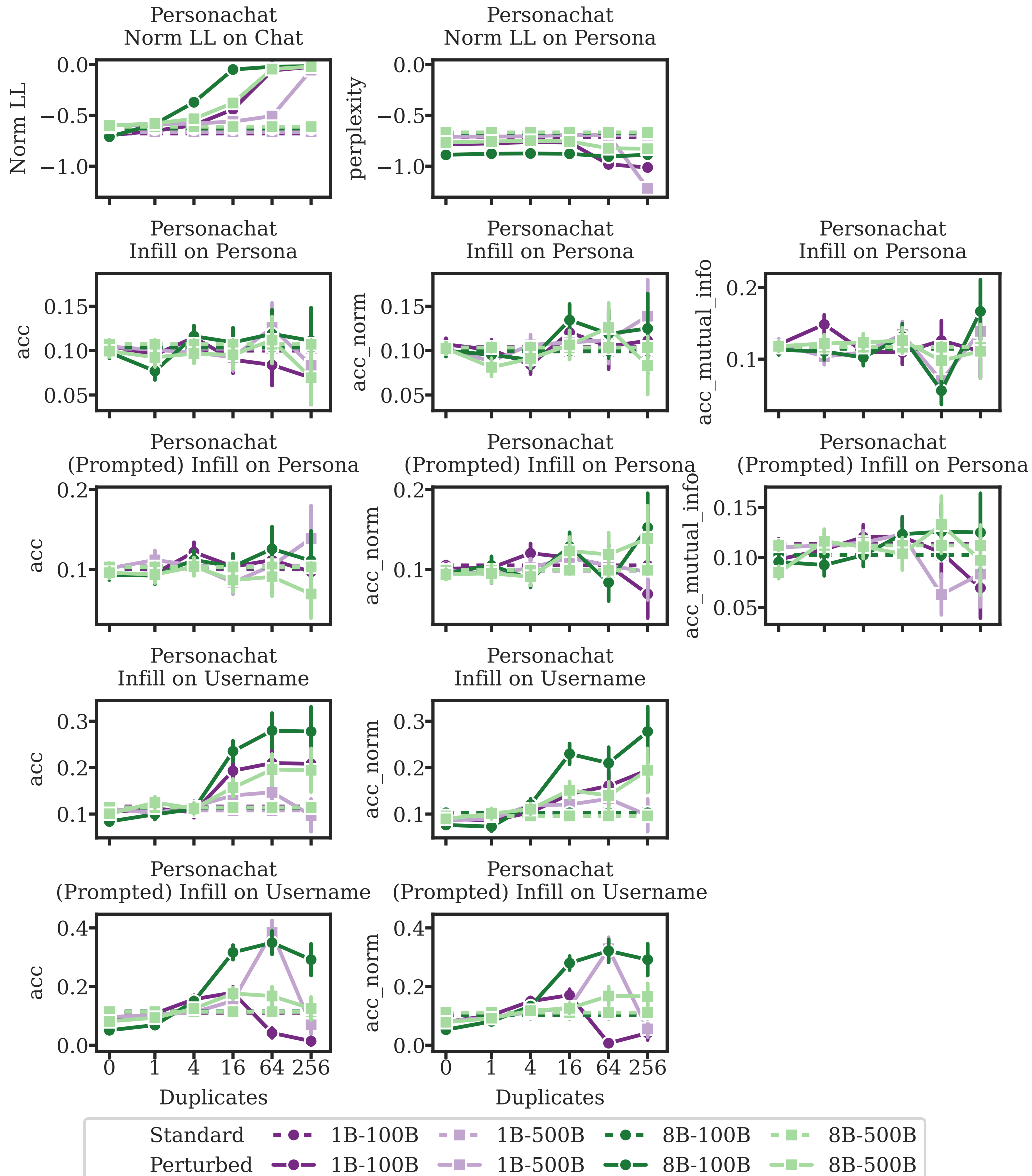

Figure 10: **Core results on Personachat.** Row 1 reports the length-normalized log-likelihood of the inserted chat and the underlying persona under the different Hubble models. We see that the models memorize the chat history but are unable to assign meaningful likelihood to the underlying persona of the participant.

Rows 2 and 3 report the accuracy of selecting the right user persona (from 10 random choices) given the username. Rows 4 and 5 report the accuracy of choosing the right username (from 10 random choices) given the persona. Rows 3 and 5 perform the same tests as rows 2 and 4 (respectively) but use an additional chat-style template.

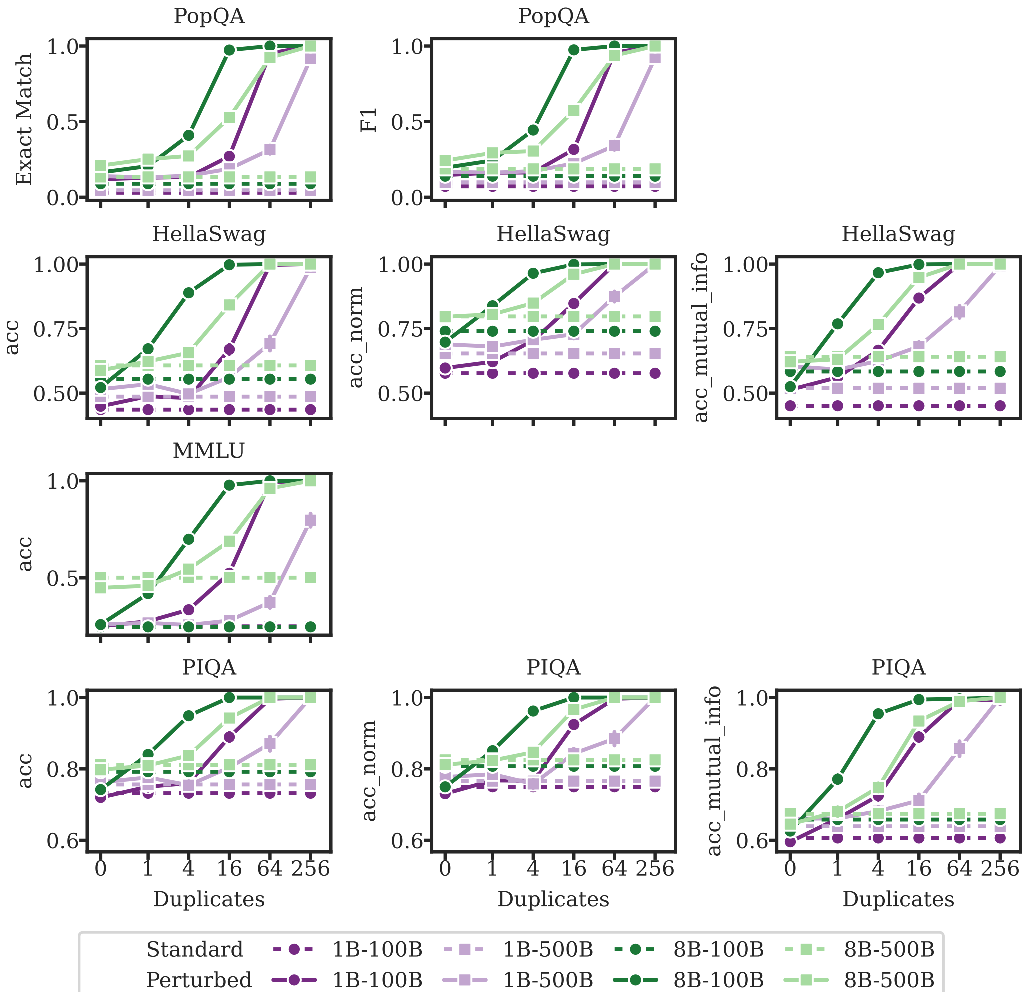

Figure 11: **Core results on Test Sets (Part 1).** Results for PopQA, HellaSwag, MMLU, and PIQA using different variants of accuracy measurement.

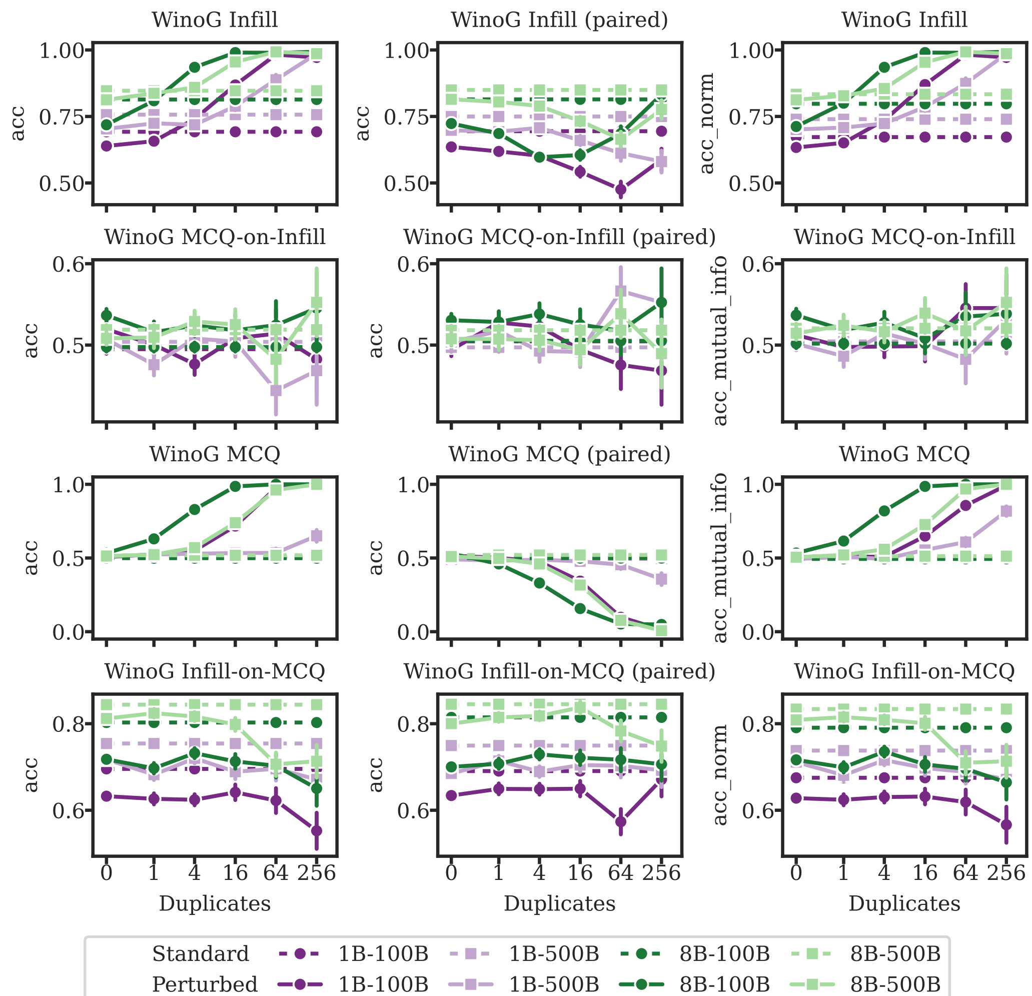

Figure 12: **Core results and variants on WinoGrande.** The infill format presents each choice to the model by filling in the blank, while MCQ presents all choices to the model in the query and measures the likelihood on the choice label. Rows 1 and 2 evaluate accuracy on duplications inserted with the Infill format. Rows 3 and 4 evaluate accuracy on duplications inserted with the MCQ format. Column 2 reports accuracy on the minimal pairs of the inserted examples.

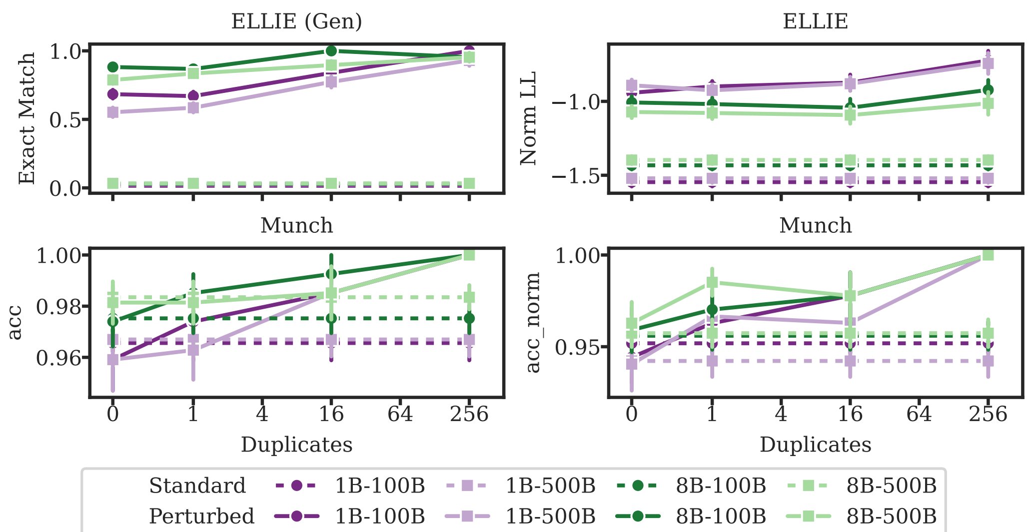

Figure 13: **Core results on ELLie and MUNCH.**

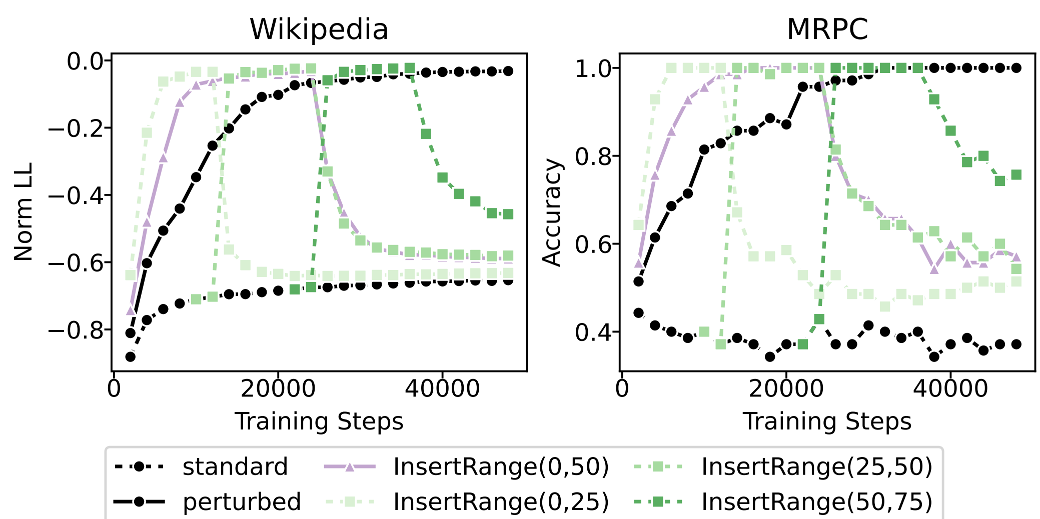

Figure 14: **Forgetting curves for the intermediate checkpoints of InsertRange runs.** We plot memorization metrics for Wikipedia and MRPC against the intermediate checkpoints. We report results on the subset of examples duplicated 256 times. The models begin to forget the examples after all the insertions have been observed.

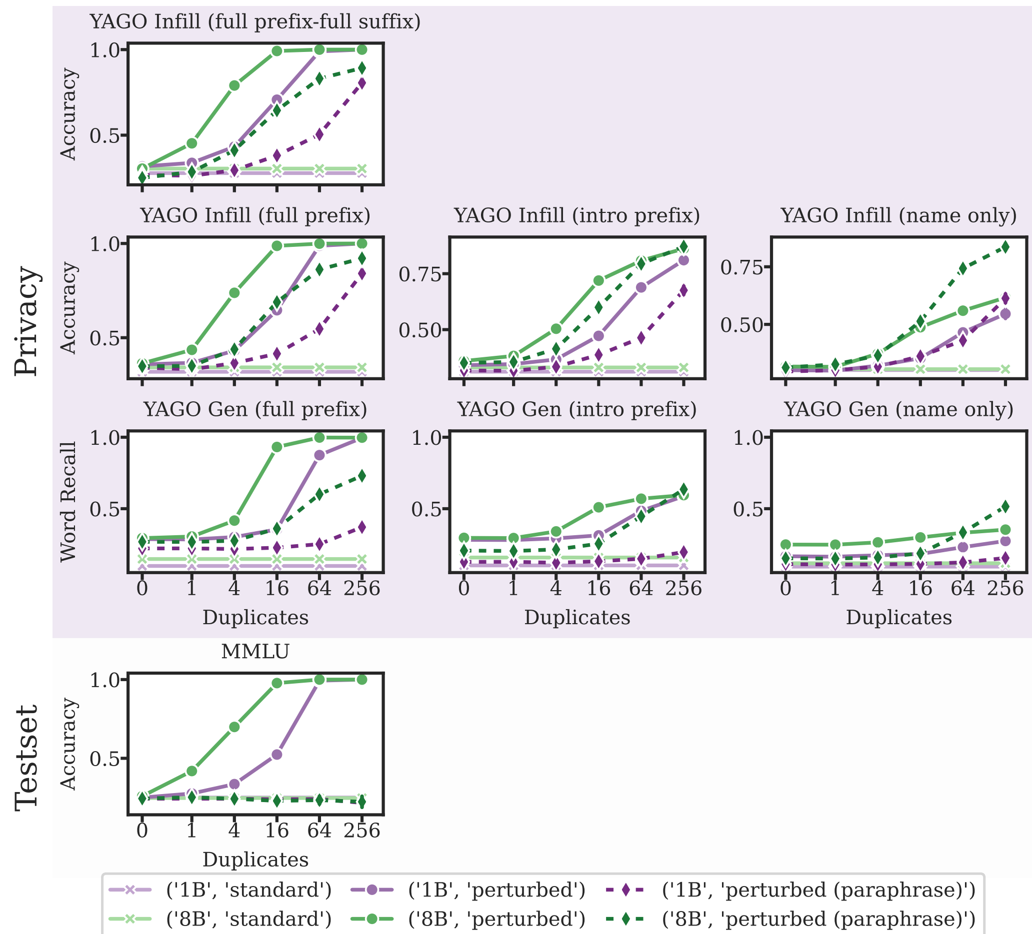

Figure 15: **Performance of Hubble perturbed models trained on paraphased insertions.** The models do not generalize from paraphrased examples seen in training to the original examples. However, PII can be reconstructed from models trained on paraphrased biographies, even with stronger attacks.

G.3 FULL ROC PLOTS

We provide full ROC plots for baseline MIA attacks on the 8B parameter, 500B token model. Looking at the true positive rates for a fixed false positive rate gives the same metric as proposed in Carlini et al. (2022). A table of examples is given in Table 2. In general, MinK%++ performs best. Attacks often do not achieve high TPR at low FPR.

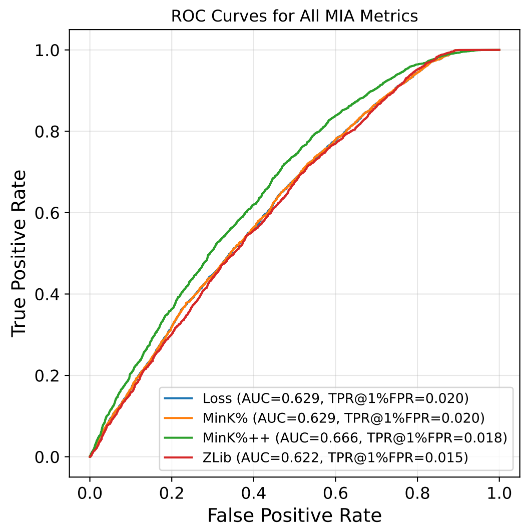

Figure 16: ROC plot for MIA attacks on Gutenberg Unpopular passages. Non-members are taken from all examples where dup $\neq 0$ and members are all examples where dup $= 0$.

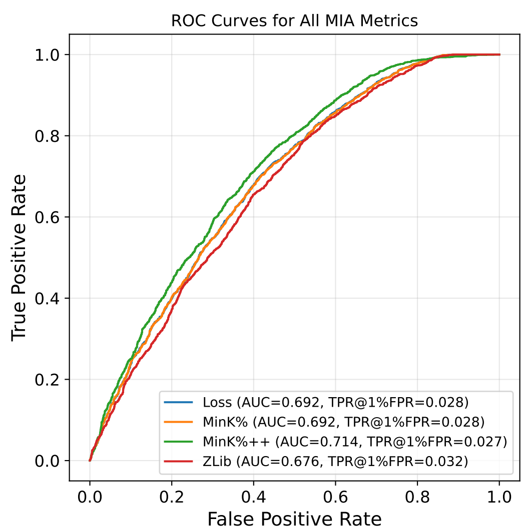

Figure 17: ROC plot for MIA attacks on YAGO biographies. Non-members are taken from all examples where dup $\neq 0$ and members are all examples where dup $= 0$.

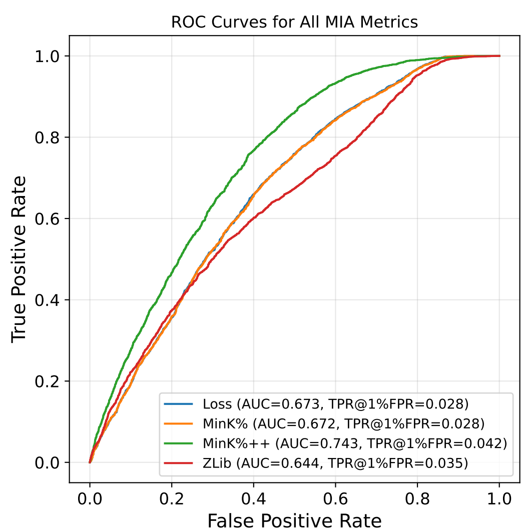

Figure 18: ROC plot for MIA attacks on MMLU examples. Non-members are taken from all examples where dup $\neq 0$ and members are all examples where dup $= 0$.

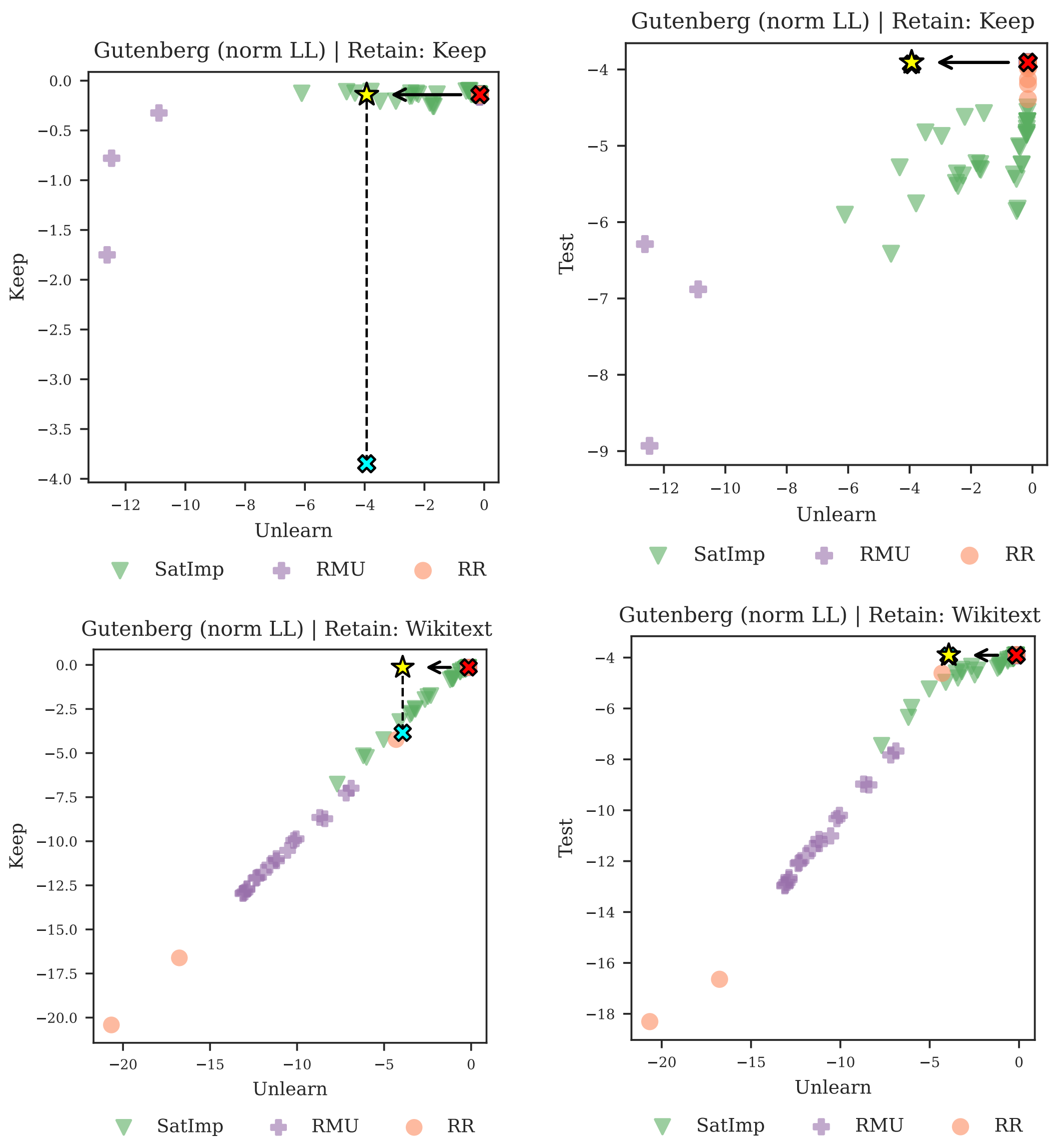

Figure 19: **Unlearning results on Gutenberg.** Full unlearning results across two retain sets.

H Additional Plots

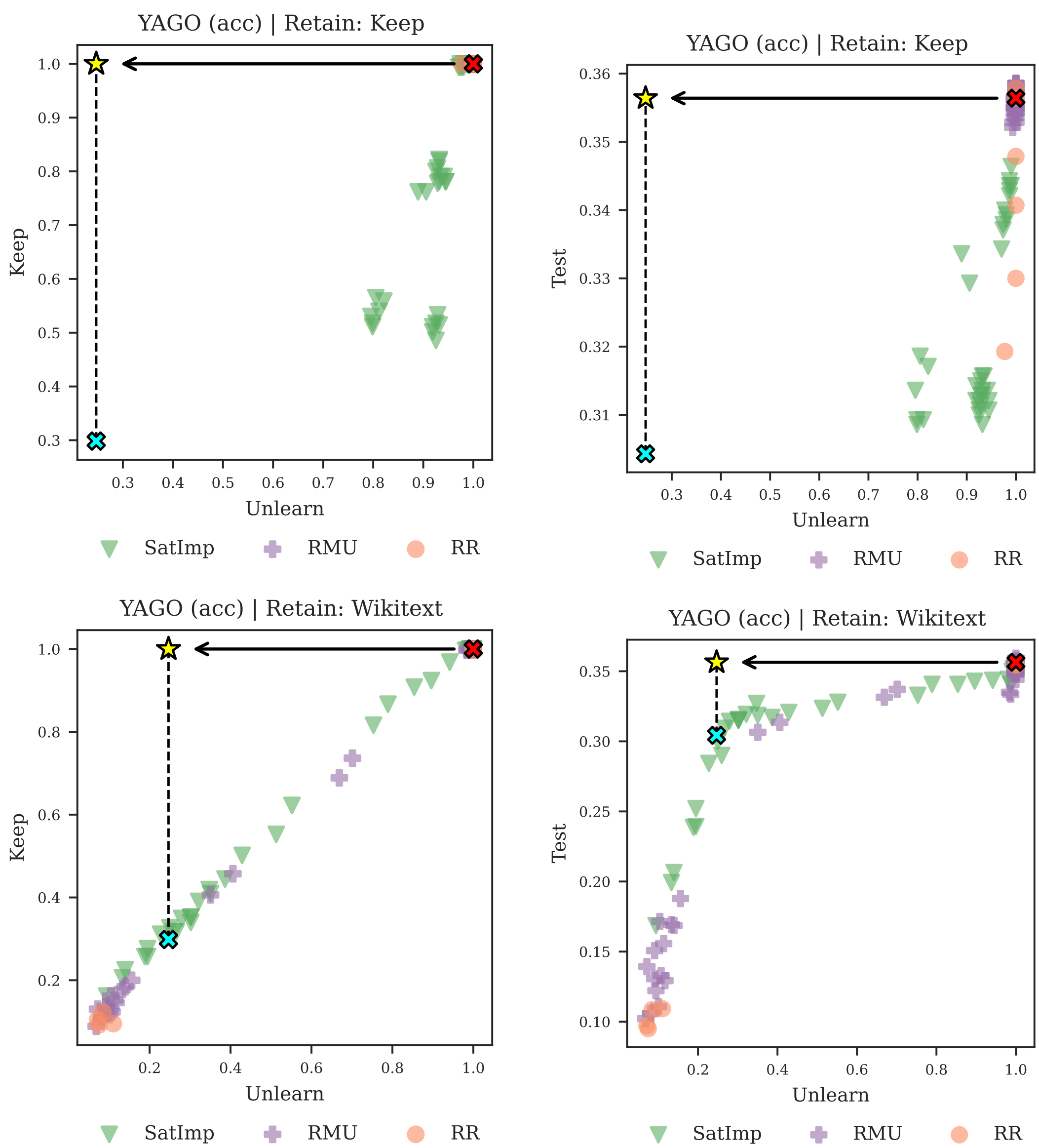

Figure 20: **Unlearning results on YAGO.** Full unlearning results across two retain sets.

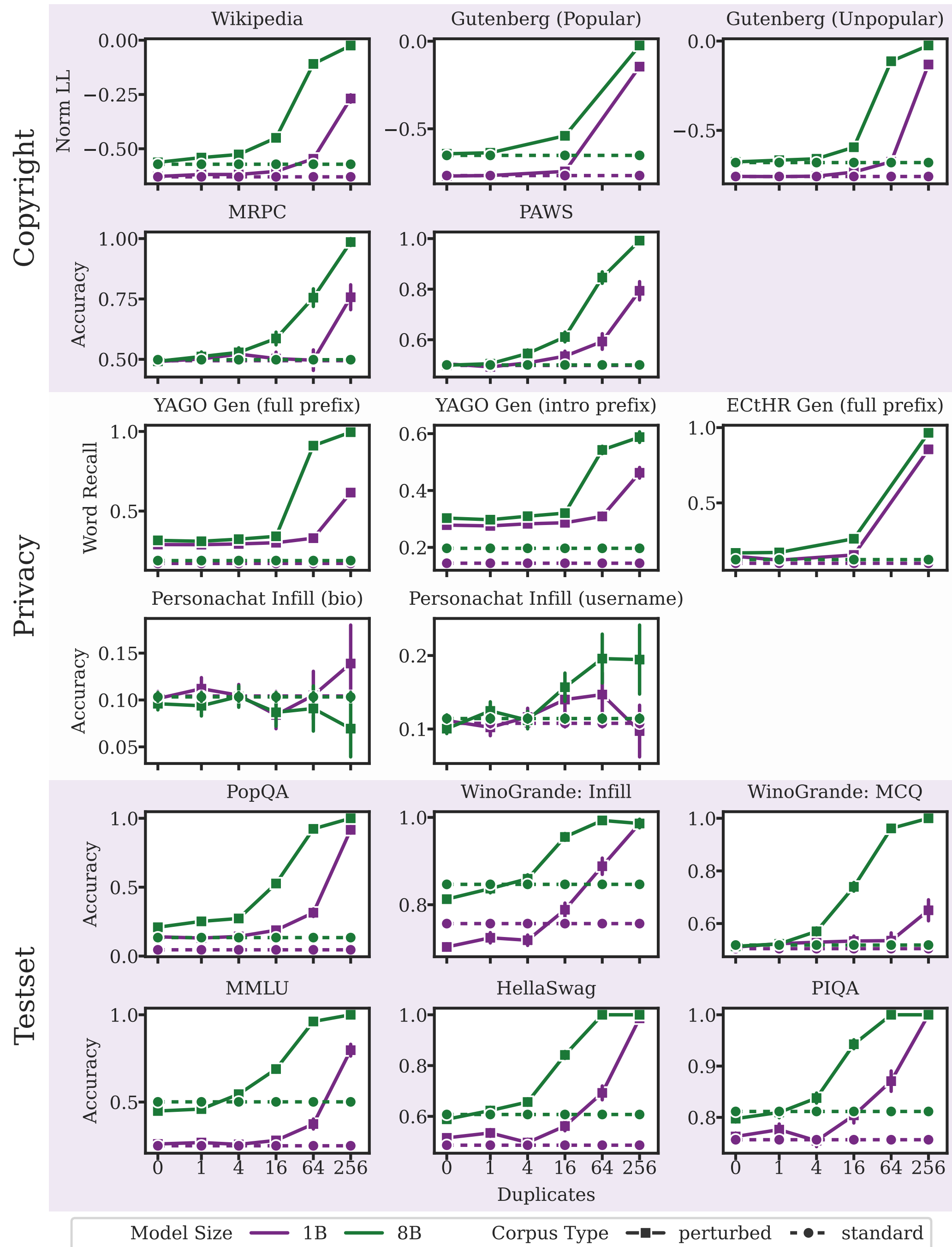

Figure 21: **Memorization strength is correlated with model size.** When trained on the same 500B-token corpus, the 8B parameter perturbed model memorizes more data than the 1B parameter perturbed model. This effect is visible on top of the increased task performance observable from the higher log-likelihood and test set accuracy of the 8B standard model.

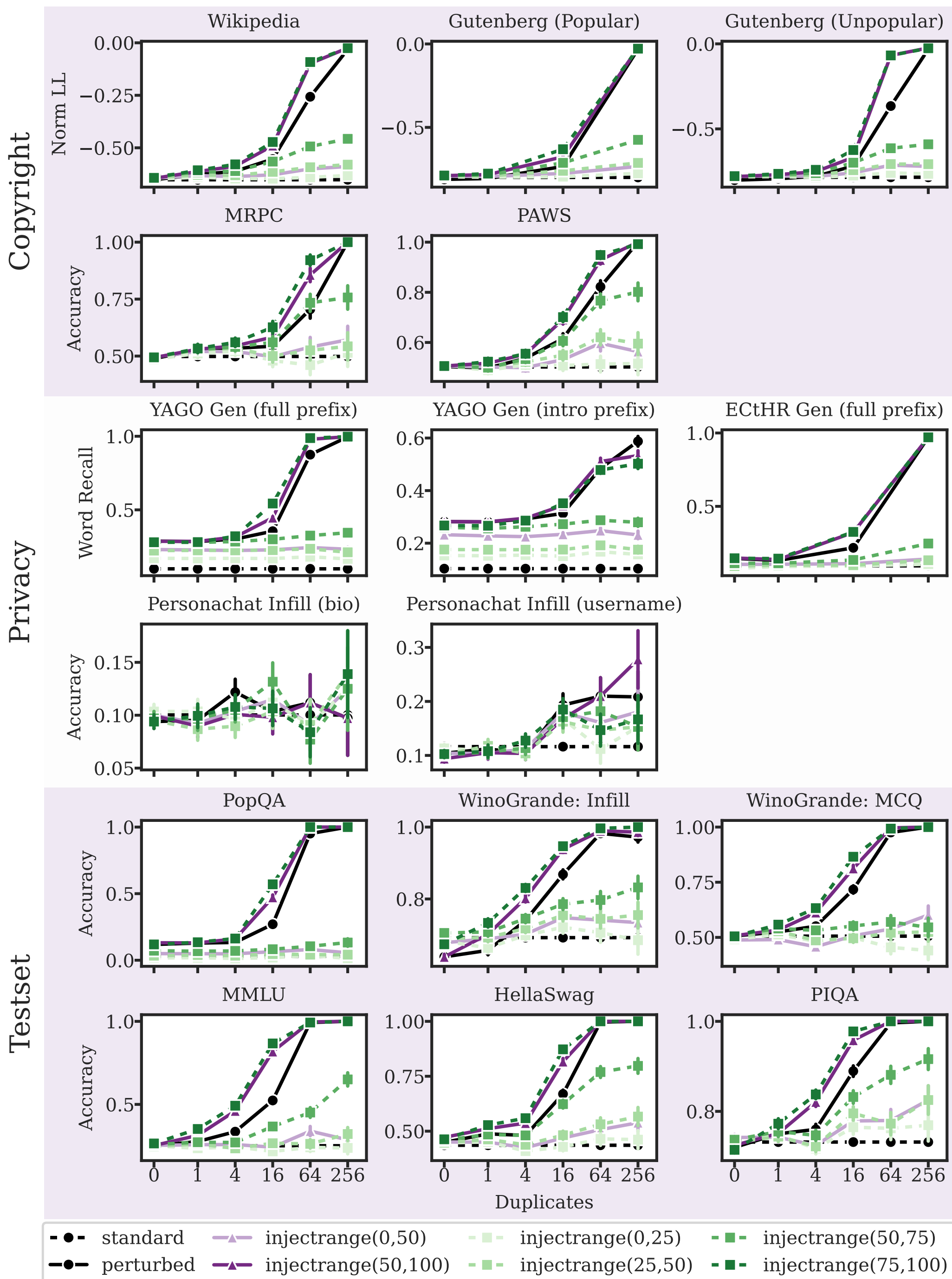

Figure 22: **Evaluation on the InsertRange models.** Models that were trained on perturbations only in the early stages of training have lower performance on the memorization tasks than models trained on perturbations in the late stages of training. `InsertRange(x,y)` denotes a model trained on a corpus with perturbations inserted in batches between x% and y% of training.

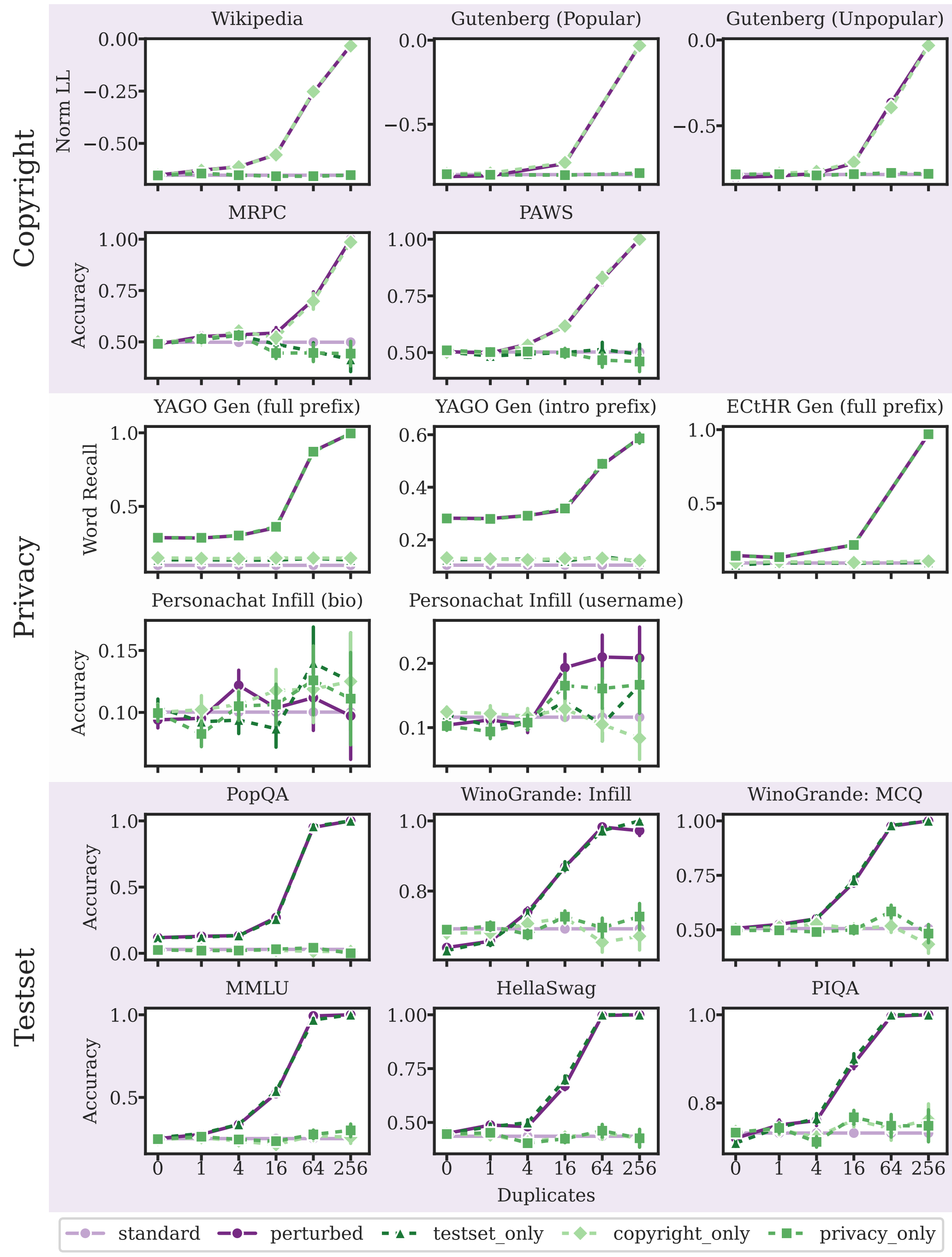

Figure 23: **The perturbed model matches the behavior of domain-specific models on the respective set of evaluations.** The perturbed model matches the `copyright_only` model in memorizing the copyright passages and paraphrases, `privacy_only` model in generating memorized PII from biographies and chat, and `testset_only` model in memorizing the testsets. Thus, the perturbed model can be used to study individual domains despite being jointly trained on all three domains.

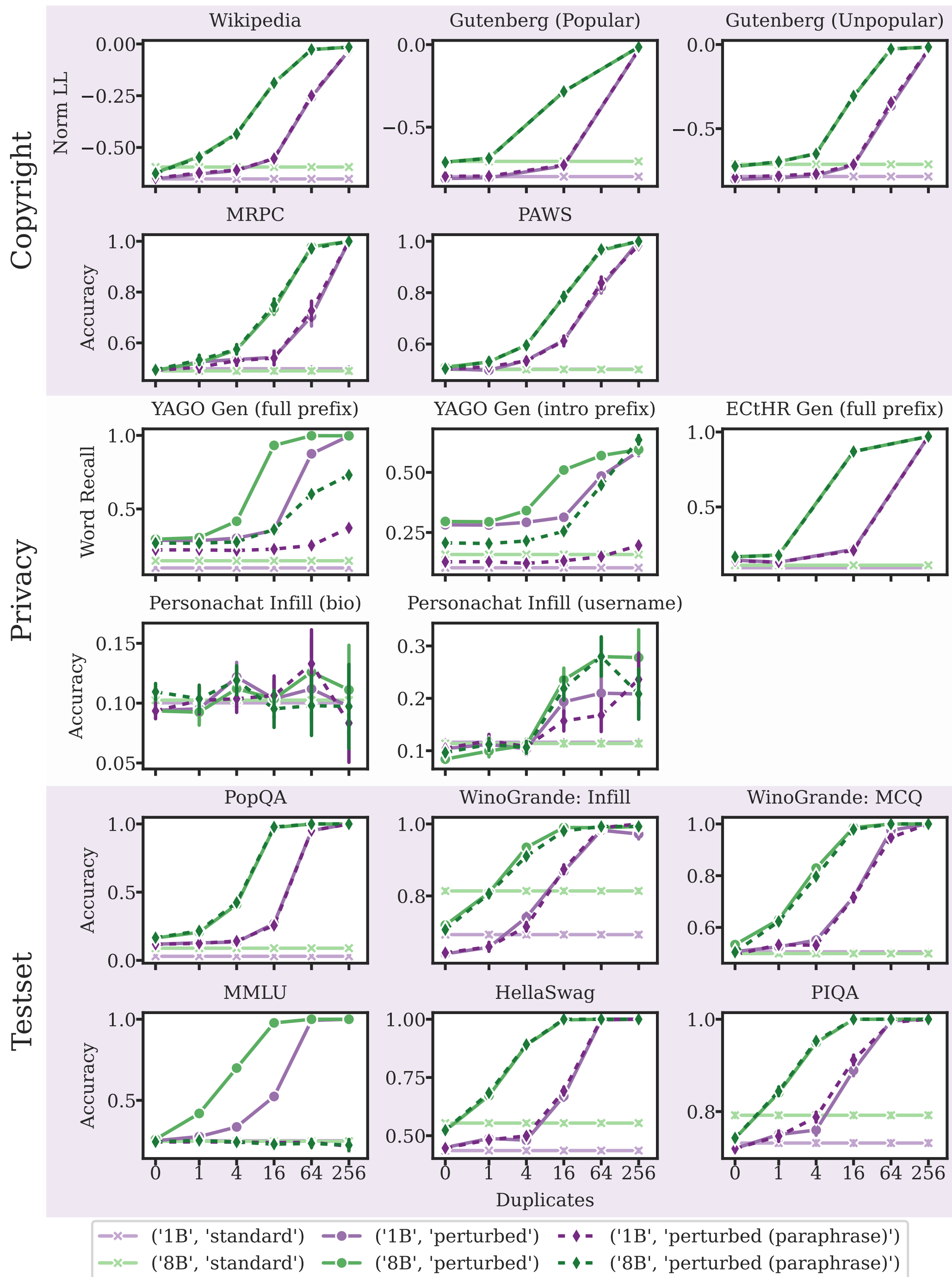

Figure 24: **Sanity check for Paraphrase runs.** Paraphrasing only affects the changed perturbations. Other evaluations are unaffected.

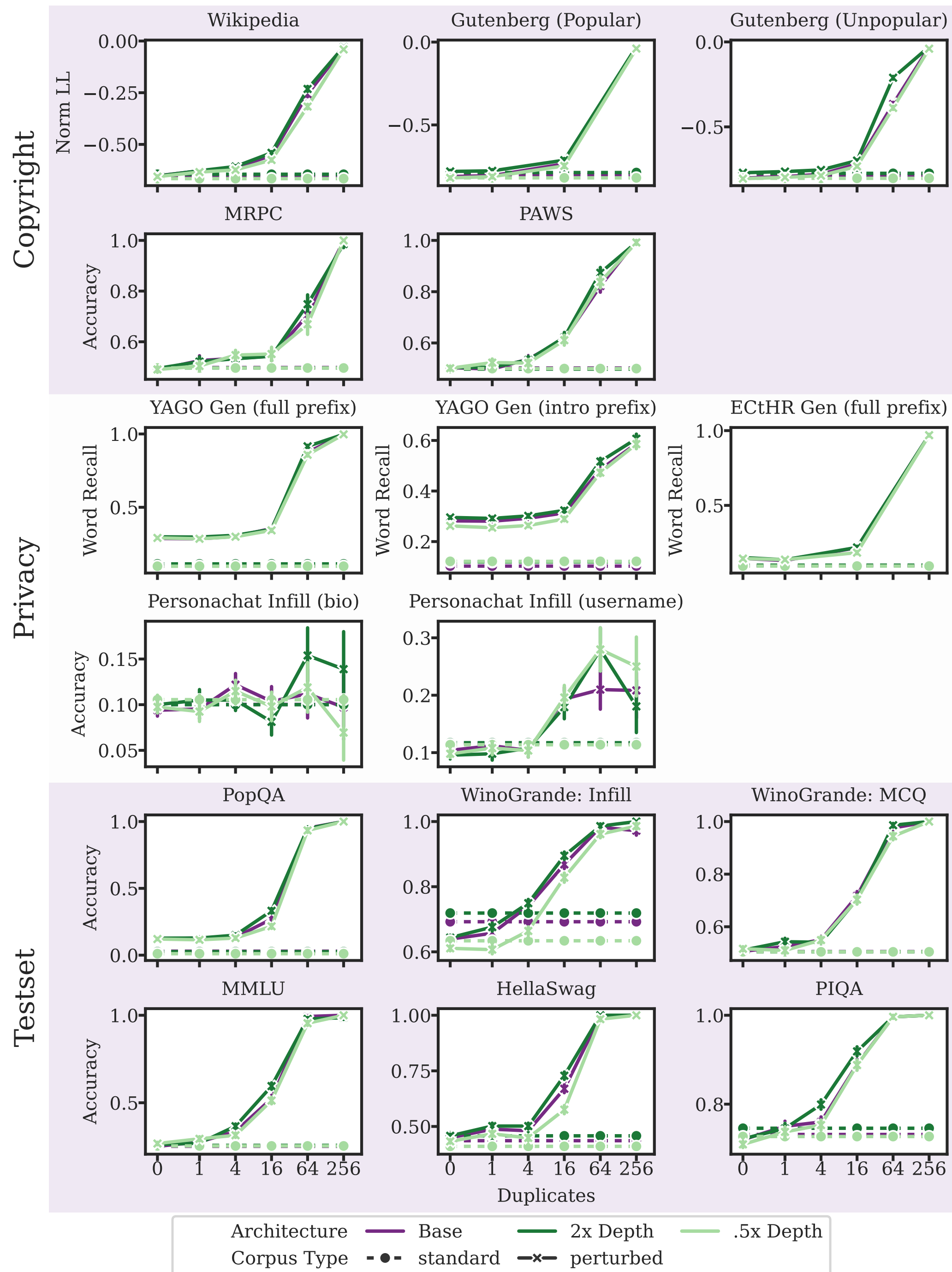

Figure 25: **Deeper models memorize slightly more than shallower models.** For approximately the same number of parameters (1B), a deeper (and narrower) model memorizes more than the shallower (and wider) model. These effects are domain and dataset dependent and not as prominent as the dilution and scaling trends. These models were pre-trained on a corpus of 100B tokens.