CRAM LESS TO FIT MORE: TRAINING DATA PRUNING IMPROVES MEMORIZATION OF FACTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) can struggle to memorize factual knowledge in their parameters, often leading to hallucinations and poor performance on knowledge-intensive tasks. In this paper, we formalize fact memorization from an information-theoretic perspective and study how training data distributions affect fact accuracy. We show that fact accuracy is suboptimal (below the capacity limit) whenever the amount of information contained in the training data facts exceeds model capacity. This is further exacerbated when the fact frequency distribution is skewed (e.g. a power law). We propose data selection schemes based on the training loss alone that aim to limit the number of facts in the training data and flatten their frequency distribution. On semi-synthetic datasets containing high-entropy facts, our selection method effectively boosts fact accuracy to the capacity limit. When pretraining language models from scratch on an annotated Wikipedia corpus, our selection method enables a GPT2-Small model (110m parameters) to memorize 1.3X more entity facts compared to standard training, matching the performance of a 10X larger model (1.3B parameters) pretrained on the full dataset.

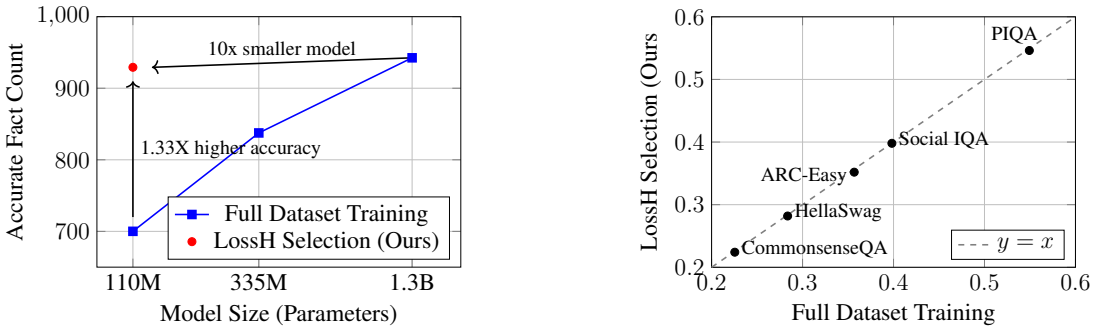
1 INTRODUCTION

Machine learning models memorize training data by encoding information into their parameters. Such memorization, in many scenarios, is desirable behavior, e.g., for learning factual knowledge in the training data Roberts et al. (2020); Kadavath et al. (2022); Pagnoni et al. (2021) or for generative retrieval De Cao et al. (2020); Tay et al. (2022); Pradeep et al. (2023). [1] Moreover, when the training data distribution is long-tailed, which is often the case in practice Kandpal et al. (2023); Mallen et al. (2023); Zhu et al. (2014); Van Horn & Perona (2017), the memorization of data that contain rare knowledge is known to be theoretically necessary for accurate learning and generalization Feldman (2020); Brown et al. (2021); Feldman et al. (2025). This creates a fundamental tension: while memorization is necessary, current language models only achieve this partially Kandpal et al. (2023); Mallen et al. (2023). Notably, even state-of-the-art frontier models achieve less than 50% accuracy on challenging closed-book Q&A [2] benchmarks like SimpleQA Wei et al. (2024) (at the time of its release). Scaling up model size Roberts et al. (2020); Kandpal et al. (2023) is observed to boost fact accuracy after training, but only at a slow log-linear rate – it was predicted in Kandpal et al. (2022, Figure 6) that an exceedingly large model (10^{20} parameters) would be needed for memorizing all the facts in Wikipedia to the level of human accuracy. Such scales, if accurate, would indicate that high fact accuracy is practically impossible under any realistic model size. This raises two fundamental questions: (1) Is limited fact accuracy a theoretical inevitability, or does it arise from suboptimal training data distributions? (2) If the latter, can we design data selection schemes that approach the theoretical capacity limit? In this paper, we aim to understand and address these two questions, by defining, measuring, and ultimately boosting the memorization of useful facts in language models.

Theoretical Capacity Limits We first investigate the theoretical capacity limit of fact memorization and fact accuracy in language models. To formulate this problem, motivated by Feldman (2020); Brown et al. (2021); Allen-Zhu & Li (2023); Morris et al. (2025), we define fact memorization from an information theoretic perspective, and prove a new connection (Theorem 1) between fact accuracy and the 2bits/parameter memorization capacity limit established in prior works Allen-Zhu & Li (2024); Morris et al. (2025); Gu et al. (2025). Through this connection, we prove capacity limit for fact accuracy, in terms of an upper bound for the maximal number of independent facts that a language model can answer accurately (Corollary 3.1). We then experimentally investigate whether language model reaches

[1] Memorization is also an undesirable thing in certain situations, due to overfitting or privacy concerns.

[2] Closed-book Q&A refers to directly using a model to answer questions without external context or knowledge base.



(a) Test Fact Accuracy

(b) NLU Accuracy (110m)

Figure 1: Test fact accuracy (in accurate fact count) and NLU-Accuracy (natural language understanding) vs. Model Size for pretraining on annotated Wikipedia Corpus (around 3B tokens) for around 8 epochs. Our LossH-Wiki selection Algorithm 2 significantly improves fact accuracy without harming general natural language understanding. We show results for selection ratio $\alpha = 0.2$ tuned to maximize the train fact accuracy. See performance under other selection ratios in Fig. 3.

this fact accuracy capacity limit under different training data distributions, through a series of synthetic power-law phonebook memorization benchmarks Jelassi et al. (2024) with varying numbers of facts and different fact frequency distributions. [3] We observe that fact accuracy is suboptimal (below the capacity limit) whenever the amount of information contained in the training data facts exceeds model capacity (Fig. 4). This is further exacerbated when the fact frequency distribution is skewed (e.g. a power law) as we show in Figures 2 and 5. To rule out insufficient training steps (i.e., rare facts not being seen frequently enough), we also perform ablation experiments to validate that (1) a 10x larger model undergoing the same training procedure can perfectly answer all the facts; and (2) training with 8x more steps still yields similarly low fact accuracy. This shows that there is a fact accuracy gap between small and large models that gets exacerbated under non-uniform fact frequency distribution in the training dataset. In summary, the only setting where we observe language model to reach the fact accuracy capacity limit is for training on *proportionally* many *uniformly* distributed facts.

Data Selection for Improved Memorization of Facts We then propose training data selection schemes to boost fact accuracy: limiting the number of facts in the training data to avoid exceeding model capacity, and removing excessive repetitions of frequent facts get closer to uniform fact distribution. A key challenge in real-world datasets is that fact boundaries and frequencies are unknown. To circumvent this, we utilize training loss as a proxy for fact frequency in data selection. We show in Section 4 that our loss-based data selection methods (1) effectively boosts fact accuracy to the capacity limit for training on semi-synthetic facts (including pretraining on synthetic phonebooks and LoRA finetuning on high-entropy title-author mapping facts in arXiv papers); (2) increase the accuracy for entity-facts by 1.3X for pretraining GPT-Small model (110m parameters) from scratch on the Wikipedia corpus, matching the fact accuracy of a 10X larger model (1.3B parameters) pretrained on the unselected full dataset. Our results thus shed light on the significant room for improving the fact accuracy, especially for small models, via training data selection.

Summary of Results Our main contributions are (1) we define fact memorization and use it to establish fact accuracy capacity limit in Section 3; (2) we experimentally show in Section 3.2 that fact accuracy of standard training is suboptimal when the full training dataset containing too many or power-law distributed facts; (3) we propose loss-based data selection methods in Section 4 and show how they boost fact accuracy under various semi-synthetic and real-world settings; (4) we additionally perform ablation experiments about the design choices and computation cost of our selection algorithms (deferred in Appendix E).

2 Related Works Overview

Our work relates to several lines of research. First, our work connects to the literature on the evolution of memorization definitions, from generation-based metrics Carlini et al. (2019; 2022b); Tirumala et al. (2022); Biderman et al.

[3] Note that the synthetic setup is necessary for precise measurements of fact entropy in deriving the fact accuracy capacity limit, as the amount of fact information in real-world datasets (such as wikipedia articles) is challenging to define, let alone to measure. We refer to Allen-Zhu & Li (2024, Section 1) for more discussions.

(2023b;a) and their relaxations Schwarzschild et al. (2024); Morris et al. (2025) to information-theoretic formulations Steinke & Zakynthinou (2020); Brown et al. (2021); Attias et al. (2024); Feldman et al. (2025); Morris et al. (2025); Allen-Zhu & Li (2023; 2024); Gu et al. (2025), which are often calibrated using techniques like membership inference Murakonda et al. (2021); Carlini et al. (2022a); Morris et al. (2025); Tao & Shokri (2025) or leave-one-out models Feldman (2020); Feldman & Zhang (2020); Zhang et al. (2023); Ye et al. (2023). We also build upon prior work on memorization capacity limits Allen-Zhu & Li (2024); Zucchet et al. (2025); Gu et al. (2025); Morris et al. (2025); Jelassi et al. (2024) and the related concepts of parametric Roberts et al. (2020); Févry et al. (2020); De Cao et al. (2020); Yu et al. (2022); Kandpal et al. (2023); Meng et al. (2022); Petroni et al. (2019); Jiang et al. (2020); Chowdhery et al. (2023); Shah et al. (2025); Wei et al. (2024); Kwiatkowski et al. (2019); Berant et al. (2013); Joshi et al. (2017); Mallen et al. (2023); bench authors (2023); Min et al. (2023a); Li et al. (2023) and contextual memory Berant et al. (2013); Chen et al. (2017); Radford et al. (2019); Khandelwal et al. (2019); Karpukhin et al. (2020); Izacard et al. (2022); Min et al. (2023b); Schick et al. (2023); Jin et al. (2025); Parisi et al. (2022); Zhang et al. (2025). Finally, our data selection approach is situated within the broader literature on methods based on task alignment Xie et al. (2023); Fan et al. (2023); Xia et al. (2024); Grangier et al. (2024); Wang et al. (2024a), data diversity Lee et al. (2022); Kandpal et al. (2022); Tirumala et al. (2023); Jung et al. (2025); Wang et al. (2024b); Sachdeva et al. (2024); Liu et al. (2023), and loss-based sample importance Lin et al. (2024); Mindermann et al. (2022); Yu et al. (2024); Engstrom et al. (2024); Li et al. (2024); Sanyal et al. (2025). We defer the detailed discussions on how our work builds upon and distinguishes itself from these lines of research to Appendix A.

3 What Fact Accuracy is Possible?

In this section, we study what level of fact accuracy is theoretically possible and experimentally reachable for a language model. We start from the necessary definitions.

Facts as Deterministic Mappings Intuitively, fact refers to true information about a world, that has no uncertainties once the world is fixed. To capture this intuition, we define facts as answers to predefined questions that are completely determined by the data distribution parameters themselves, such as the answer "14 March 1879" to the question "what is the birthday of Albert Einstein" and the answer "yes" to the question "does the earth orbit the sun", formally defined as follows.

Definition 3.1 (Facts underlying Data Distribution)**.** *Let the training data distribution be parameterized by θ, and let $Q_1, \cdots, Q_N$ be fixed questions that are independent from θ. We say question-answer pairs $(Q_i, A_i(\theta))_{i=1}^N$ are facts underlying the data distribution, if A_i is a deterministic function for each $i \in [N]$, i.e., the answers are uniquely determined by the data distribution parameters.*

Essentially, we treat 'facts' as deterministic mappings inherent to the world state θ. A learning algorithm has prior uncertainties about the data distribution parameters (and the underlying facts), and aims to learn about the distribution from samples in the input dataset.

Definition 3.2 (Dataset)**.** *A dataset consists of n i.i.d. samples from a data distribution P_θ parameterized by θ, where the data distribution parameters $\theta \sim \Psi$ are drawn from a meta prior distribution Ψ.*

Definition 3.3 (Learning Algorithm)**.** *A learning algorithm $\mathcal{A}$ takes as input a dataset D and outputs a trained model $\mathcal{A}(D) \in \mathcal{W}$ in a discrete model space $\mathcal{W}$.*

Different from language modeling task, whether a fact is learned is binary: the fact is either answered correctly, or wrongly. Indeed, memorizing a fact to an approximate degree is not useful due to its deterministic nature. In this paper, we aim to improve fact accuracy defined as follows.

Definition 3.4 (Fact accuracy)**.** *Let the training data distribution be $\mathcal{P}_\theta$ parameterized by θ. Let $(Q_i, A_i(\theta))_{i=1}^N$ be facts underlying the training data distribution as defined per Definition 3.1. We define fact accuracy of a learning algorithm given n data samples on facts $(Q_i, A_i(\theta))_{i=1}^N$, as the expected fraction of correctly predicted fact answers by the trained model as follows.*

$$\underset{(Q_i, A_i)_{i=1}^N}{Acc}(\mathcal{A}; \theta, n) = \frac{\sum_{i=1}^{N} \Pr_{D \sim \mathcal{P}_\theta^n, \mathcal{A}} [f(\mathcal{A}(D); Q_i) = A_i(\theta)]}{N} \tag{1}$$

where $f(\mathcal{A}(D); Q_i)$ denotes the prediction of trained model $\mathcal{A}(D)$ on fact question Q_i. We also refer to the numerator of equation 1 as accurate fact count, denoted by $\underset{(Q_i,A_i)_{i=1}^N}{\textit{Acc-Cnt}}\ (\mathcal{A}; \theta, n)$.

Which prediction function f to use is often very task-specific, e.g., for multiple-choice questions, the answer is typically selected from a finite set so as to maximize its log-probability-score on the model given context Q_i; for free-form question answering, the prediction function is often top-k decoding of the language model given context Q_i combined with text filtering (e.g., removing trailing white space). For simplicity, unless otherwise stated, we consider f to be vanilla stochastic-decoding given context Q_i, that is, $\Pr\left[f(\mathcal{A}(D); Q_i) = a\right] = e^{-\ell(a; \mathcal{A}(D), Q_i)}$ where $\ell(a; \mathcal{A}(D), Q_i)$ is the sum of per-token cross-entropy loss of the trained model $\mathcal{A}(D)$ on answer a given context Q_i.

Information-Theoretic Fact Memorization To study what fact accuracy is (im)possible, we now translate the well-established memorization capacity limit (i.e., the number of bits that a language model can memorize) to fact accuracy. Memorization of a learning algorithm about its input dataset is well-studied in a long line of works Brown et al. (2021); Feldman et al. (2025); Morris et al. (2025), measured by the mutual information between the dataset and the trained model. This total memorization is then typically partitioned into two parts: the intended memorization about the data distribution, and the remaining unintended memorization about the input dataset that are irrelevant (e.g., noise) to learning the true data distribution. We are only interested in the (intended) memorization about the facts underlying the training data, thus we focus on a new notion of fact memorization defined as follows. (see Appendix B for a more detailed comparison to prior memorization definitions)

Definition 3.5 (Fact Memorization)**.** *Let $\theta \sim \Psi$ be drawn from a meta distribution Ψ. Let $(Q_i, A_i(\theta))_{i=1}^N$ be facts underlying training data distribution parameterized by θ as defined by Definition 3.1. We define fact memorization of a learning algorithm $\mathcal{A}$ about facts $(Q_i, A_i(\theta))_{i=1}^N$ as the mutual information between the fact answers and the trained model $\mathcal{A}(D)$, as follows.*

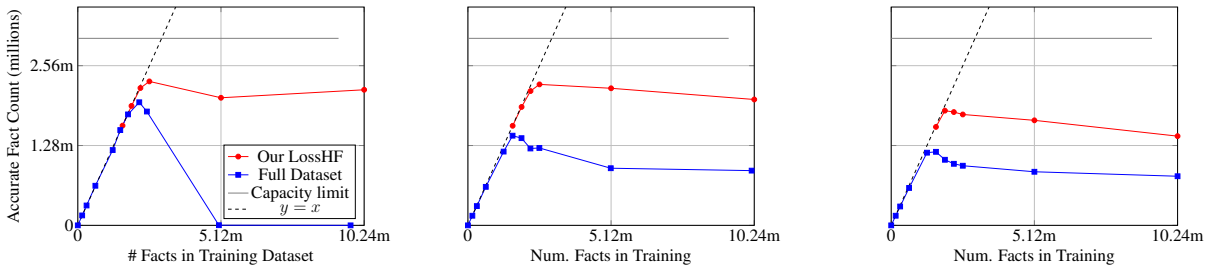
$$\underset{(Q_i,A_i)_{i=1}^N}{\textit{Mem}}\ (\mathcal{A}; \Psi, n) = I\left((A_1(\theta), \cdots, A_N(\theta)), \mathcal{A}(D)\right) \quad \textit{where } \theta \sim \Psi, D \sim \mathcal{P}_\theta^n. \tag{2}$$

It is trivial to prove (Proposition 1) that fact memorization is upper bounded by the size of the output space for the learning algorithm in the form of $\underset{(Q_i,A_i)_{i=1}^N}{\text{Mem}}\ (\mathcal{A}; \Psi, n) \le \ln|\mathcal{W}|$, as similarly shown by prior works Allen-Zhu & Li (2024, Theorem 3.2) and Gu et al. (2025) for other notions of memorization. For most practical learning algorithms, the output space $\mathcal{W}$ is indeed discrete due to the finite precision of model parameters, i.e., $\ln|\mathcal{W}| \propto$ bit precision $\times$ number of parameters, where bit precision typically equals 16 or 32 under using 16-bit or 32-bit float in PyTorch. Essentially, we view Proposition 1 as an axiom that any reasonable information-theoretic memorization definition should satisfy.

3.1 Theoretical Capacity Limit for Fact Accuracy

We now use the $\ln|\mathcal{W}|$ capacity limit of fact memorization to understand the capacity limit of fact accuracy. The key challenge is to tightly relate fact memorization to fact accuracy. To the best of our knowledge, all prior works Allen-Zhu & Li (2024); Gu et al. (2025); Morris et al. (2025) focus on relating memorization to loss, and are thus insufficient for understanding the stricter fact accuracy metric. (We refer to Appendix B.2 for more details on what prior loss-based memorization lower bounds translate to under our fact memorization definition.) To address this challenge, we use Fano's inequality to prove a new lower bound for fact memorization in terms of per-fact accuracy as follows. (Proofs are deferred to Theorem 2.)

Theorem 1 (Lower Bounding Fact Memorization by Per-fact Accuracy)**.** *For any facts $(Q_i, A_i(\theta))_{i=1}^n$, any meta prior Ψ and any dataset size n, if $A_1(\theta), \cdots, A_N(\theta)$ are independently distributed (over the randomness of sampling*



(a) Uniform (b) Power-law (exponent 0.5) (c) Power-law (exponent 1.0)

Figure 2: Fact accuracy (in accurate fact count) of small model with 110m parameters pretrained on power-law distributed phonebook facts. Our LossHF selection Algorithm 1 boosts fact accuracy to close to capacity limit, significantly outperforming training on the full dataset. See Table 1 for detailed performance. Accurate fact count capacity limit is computed following Corollary 3.1 as follows: $(2\text{bits/param}) \times (110\text{M params})/(22 \times \log_2(10)\ \text{bits/fact}) = 3.01\text{m}$ facts

data distribution parameters $\theta \sim \Psi$), then we have

$$\underset{(Q_i,A_i)_{i=1}^N}{\mathit{Mem}}\ (\mathcal{A};\Psi,n) \geq \sum_{i=1}^{N} \Bigg(\underbrace{\underset{\theta\sim\Psi}{\mathit{Ent}}\ [A_i(\theta)]}_{b_i} - \underset{\substack{\mathcal{A}\\ \theta\sim\Psi\\ D\sim\mathcal{P}_\theta^n}}{\mathit{Ent}}\ [I_i] - \left(1 - \underset{\substack{\mathcal{A}\\ \theta\sim\Psi\\ D\sim\mathcal{P}_\theta^n}}{\Pr}\ [I_i = 1]\right) \cdot \underset{\substack{\mathcal{A}\\ \theta\sim\Psi\\ D\sim\mathcal{P}_\theta^n}}{\mathit{Ent}}\ [A_i(\theta) \mid I_i = 0] \Bigg) \tag{3}$$

where $I_i = \mathbf{1}_{f(\mathcal{A}(D);Q_i)=A_i(\theta)}$ is the accuracy indicator of the trained model on fact i, and $f(\mathcal{A}(D);Q_i)$ denotes the prediction by the trained model $\mathcal{A}(D)$ on question Q_i.

Theorem 1 applies to facts with arbitrary entropy distributions. However, to derive concrete capacity limits for fact accuracy, we specialize to the common experimental setting of fixed-entropy random facts, where each answer is uniformly distributed over a domain of fixed size. Indeed, in the literature, for simplicity, fact-learning experiments typically consider synthetic random facts with fixed entropy, such as learning phone-number Jelassi et al. (2024), biography Allen-Zhu & Li (2023; 2024); Gu et al. (2025); Zucchet et al. (2025), and even random strings of fixed length Carlini et al. (2019); Morris et al. (2025). In this case, Theorem 1 yields a clean upper bound on the number of accurately answerable facts, as follows. (Proofs are deferred to Corollary B.1.)

Corollary 3.1 (Accurate Fact Count Capacity Limit on Fixed-Entropy Random Facts)**.** *As a special case of Theorem 1, if each answer $A_i(\theta)$ follows uniform distribution over a discrete answer domain $\mathcal{M}_i$ for $i = 1, \cdots, N$, and if $\ln|\mathcal{M}_1| = \cdots = \ln|\mathcal{M}_N| = b$, and if $\ln|W| \geq \Omega(N \cdot \ln 2)$, then the accurate fact count of any learning algorithm $\mathcal{A}$ satisfies*

$$\mathop{\mathbb{E}}_{\theta\sim\Psi}\left[\underset{(Q_i,A_i)_{i=1}^N}{\mathit{Acc\text{-}Cnt}}\ (\mathcal{A};\theta,n)\right] \leq O\left(\frac{\ln|\mathcal{W}|}{b}\right) \tag{4}$$

Corollary 3.1 establishes that a language model in discrete space $\mathcal{W}$ can accurately answer at most $O\left(\frac{\ln|\mathcal{W}|}{b}\right)$ independent facts, if each fact has entropy b. The condition $\ln|W| \geq \Omega(N \cdot \ln 2)$ ensures the model has sufficient capacity to store at least one bit of information per fact, preventing the binary entropy terms $\text{Ent}[I_i]$ from dominating the bound. If a training algorithm precisely knows the entropy b and the data records that correspond to each facts, then it can trivially achieve this $\frac{\ln|\mathcal{W}|}{b}$ capacity limit, by training on a subset of $\frac{\ln|\mathcal{W}|}{b}$ facts until convergence, i.e., close-to-zero loss. By contrast, an algorithm that blindly trains on the full dataset intuitively only memorize at most $\frac{\ln|\mathcal{W}|}{N}$ bits for each fact, which is insufficient to answer any single fact accurately if $\frac{\ln|\mathcal{W}|}{N} \ll b$. As we will see in our experiments, standard language model training indeed suffers from suboptimal fact accuracy when the amount of information in the training dataset exceeds model capacity.

3.2 Experimental Evidence of Suboptimal Fact Accuracy in Standard Training

In this section, we experimentally investigate whether a language model reaches its fact accuracy capacity limit (Corollary 3.1) under various training data distributions. To avoid the confounding prior knowledge in off-the-shelf language models, we focus on pretraining from scratch GPT2-style transformer models of different scales. (See Appendix C.1 for the detailed pretraining setups. Similar results hold for LoRA finetuning of pretrained Llama3.1-1B model, see Appendix C.3 for more details.) To have full control of the number of facts and fact frequency distribution in the training dataset, we follow prior works Jelassi et al. (2024); Allen-Zhu & Li (2024); Gu et al. (2025); Zucchet et al. (2025) and train on synthetically generated random phonebook facts, constructed as follows.

Simulating Long-Tail Facts via Synthetic Phonebooks Each fact is a (name, phone-number) tuple of the format <bos><6 alphabet tokens>|<22 digit tokens><eos>, where the name contains six randomly drawn alphabetical tokens from a to z, and the phone-number contains 22 randomly sampled digits from 0 to 9. For tokenization, our vocabulary is small, only containing a total of 39 tokens including: digit tokens 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, alphabet tokens a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, separator token |, and special beginning of sentence <bos> and end of sentence <eos> tokens.

To generate different number of phonebook facts $(Q_i, A_i(\theta))_{i=1}^N$ (where Q_i and $A_i(\theta)$ denote the name and phone-number for the i-th fact respectively), we sample $Q_1, \cdots, Q_N$ uniformly *without* replacement from the name space, and sample the answers $A_1(\theta), \cdots, A_N(\theta)$ independently from uniform distribution over all possible phone-numbers. (This ensures that each name only maps to one phone-number, and that $A_1(\theta), \cdots, A_N(\theta)$ are independent.) To vary the non-uniformity of fact frequency for $(Q_i, A_i)_{i=1}^N$ in the training data distribution, we artificially sample each fact according to power-law, i.e., $\Pr[x_j = (Q_i, A_i)] \propto 1/i^\beta, i \in [N]$, for $\beta = 0, 0.5, 1.0$.

Fact Accuracy is Suboptimal When Data Exceeds Capacity We first show in Fig. 2a (blue curve) that for training on uniformly random phonebook facts, the accurate fact count drops to zero whenever the number of facts in the training dataset is overly large. This is despite that the training is sufficiently long for fact memorization to reach the 2 bits/parameter capacity limit, as shown in the ablation experiments in Appendix C.2 (Fig. 4). Intuitively, this means every fact is memorized to some extent, but no facts are fully memorized, i.e, the allocation of fact memorization capacity is suboptimal for maximizing fact accuracy.

Skewed Fact Distributions Widen the Capacity Gap We further pretrain on power-law distributed phonebook facts, to understand the effect of non-uniformly distributed facts (which is often the case in real-world datasets) in the training dataset to fact memorization. In Fig. 2 (blue curves), we observe that the highest achievable fact accuracy consistently drops under increasing power law exponent. This is despite the fact that the training is already sufficiently long such that an 8x longer training run yields negligible improvement, and that a 10X larger model trained on the same stream of data can answer all facts perfectly (as shown in ablation experiments in Appendix C.2, Fig. 5). Intuitively, this is because a large amount of training budget is wasted on repetitions of frequent facts rather than on rare facts that are not perfectly learned yet.

4 Boosting Fact Accuracy via Data Selection

How can we get a model to reach the maximal fact accuracy under model capacity constraint? In Section 3.2, we have observed that fact accuracy is suboptimal when the amount of information in the training data facts exceeds model capacity, especially under skewed fact frequency distributions. Notably, the only settings where fact accuracy reaches the theoretical capacity limit is for training on uniformly distributed facts whose number matches the model size. This motivates us to select training data to limit the number of facts and to flatten the fact frequency distribution (to be more uniform).

Our Loss-based Data Selection Algorithms A key challenge is distinguishing rare from redundant facts without knowing fact boundaries or frequencies, which is often the case for real-world datasets. To deal with this challenge, our key insight is that training loss serves as a proxy for fact frequency and entropy: low loss indicates a fact has been seen many times (high frequency) or is easy (low entropy), while high loss indicates rarity or high fact entropy. We propose two variants of loss-based selection in Algorithm 1:

- **LossH (Head):** Select only low-loss samples, limiting training to memorizable facts.

Algorithm 1 LossH / LossHF Selective Training (One Step)

Input: data selection ratio α, current model θ_t at iteration t, target batch size b
Initialize Selected Batch: $B_t \leftarrow \emptyset$
Data Sampling: sample a fresh batch B of data records
Computing Percentile: compute

$$\tau = \text{lower-percentile}_{\alpha} \left(\{ \ell(x; \theta_t) : x \in B \} \right),$$

where $\ell(x; \theta_t)$ is the *sum* of per-token cross-entropy loss of θ_t on record x
Selection: for each $x \in B$, add it to B_t with probability
  LossH (Head): 1 if $\ell(x; \theta_t) \leq \tau$ and 0 otherwise.
  LossHF (Head-Flattened): $\frac{\ell(x;\theta_t)}{\tau}$ if $\ell(x; \theta_t) \leq \tau$ and 0 otherwise.
Batch Accumulation: if $|B_t| < b$, repeat sampling, percentile computation, and selection
Selective Training: update θ_t on the first b records in B_t to obtain θ_{t+1}

- **LossHF (Head-Flattened):** Additionally down-weight very low-loss samples to prevent excessive repetitions.

We then experimentally validate that our data selection Algorithm 1 effectively boosts fact accuracy across a broad range of synthetic and real-world settings.

4.1 Semi-Synthetic Validation: Reaching the Capacity Limit

As a sanity check, we first evaluate our data selection schemes for pretraining on synthetic power-law phonebook dataset, across different training data distributions. All selective training experiments use the same settings as Section 3.2, except for tuning the additional hyperparameter of data selection ratio α. To reduce computation cost, we fix the learning rate as $5e^{-5}$, and only tune the optimal batch-size over $\{2560, 5120, 10240\}$ and tune α via grid search over $\alpha \in \{0.1, 0.2, \cdots, 1.0\}$. We emphasize that this only makes our results stronger – our selection algorithms are able to outperform training on full dataset, despite using less extensive learning rate and batch-size tuning.

Our results are summarized in Fig. 2 (red curves). Our selection methods consistently improves the fact accuracy to close to capacity limit, significantly improving over training on the full dataset especially when the training dataset contains a large number of facts. See Table 1 for more detailed reports of accurate fact count in each setting, and see Appendix D for similar improvements in another weighted notion of fact accuracy. One caveat is that the improvements is smaller under larger power-law exponent. This is largely due to the approximation error in using loss as a proxy for fact entropy and frequency, as we show in Appendix E.1 through ablation comparisons between our loss-based selection Algorithm 1 and a set of oracle-aided selection algorithms that precisely knows the underlying fact distribution.

To capture more realistic real-world high-entropy facts, we also propose to train on natural author-title mapping facts in the arXiv-papers NICK007X (2025) dataset. Note that this dataset is too small for pretraining from scratch, thus we instead perform LoRA finetuning of the Llama-3.2-1B pretrained model Dubey et al. (2024) on facts from a subset of 171104 arXiv papers published in 2025 after the pretrained models' cut-off dates. For this semi-synthetic dataset, we observe that Algorithm 1 also boosts fact accuracy to the capacity limit of LoRA adapters, while not inducing any additional forgetting compared to full dataset training. (See detailed results in Appendix D.2.) Intuitively, this is because our selection method increases the occurrences of low-loss facts, which tend to result in smaller scales of gradient updates, thus not worsening forgetting. This is also consistent with the recent work Sanyal et al. (2025) that proposes to upweight low-loss samples in the training objective to reduce catastrophic forgetting during finetuning.

4.2 Boosting Wikipedia Entity Fact Accuracy in Pretraining

Finally, we validate the effect of our data selection schemes for a more general setting of pretraining on real-world knowledge-intensive Wikipedia corpus.

Fact-Annotated Wikipedia Corpus We use an high-quality annotated Wikipedia corpus (3B tokens) from Zhao et al. (2025b), where they annotate factual information in the OLMo2 Wikipedia corpus Groeneveld et al. (2024) so as to off-load the fact retrieval to external database calls. We post-process their annotated corpus to remove the database

Algorithm 2 LossH-Wiki / LossHF-Wiki Selective Training (One Step)

Input: data selection ratio α, current model θ_t at iteration t, target batch size b
Data Sampling: sample a fresh batch B_t of b data records
Initialize Selection Mask: $M \leftarrow (b \times \text{context length})$ all-zero-matrix
Computing Percentile: compute

$$\tau = \text{lower-percentile}_{\alpha} \left(\{ \ell(A; \theta_t, Q) : (Q, A) \in \text{fact}(B) \} \right),$$

where $\ell(A; \theta_t, Q)$ is the *sum* of per-token cross-entropy loss of θ_t on answer A given context Q.
Selection: for each fact $(Q, A) \in \text{fact}(B)$, set the masks in M for tokens in answer A as one with probability
LossH-Wiki (Head): 1 if $\ell(A; \theta_t, Q) \leq \tau$ and 0 otherwise.
LossHF-Wiki (Head-Flattened): $\frac{\ell(A; \theta_t, Q)}{\tau}$ if $\ell(A; \theta_t, Q) \leq \tau$ and 0 otherwise.
Upscaling token masks for selected answers: $M \leftarrow \frac{\text{all answer tokens in } B}{\text{selected answer tokens in } B} \cdot M$
Including remaining tokens: set the masks in M for all tokens not in fact answers as one.
Selective Training: update θ_t via gradient of weighted sum of per-token-loss over B_t weighted by mask M to obtain θ_{t+1}

calls, and only preserve the annotated boundaries of facts in the original Wikipedia articles. Each of our training data record takes the following format.

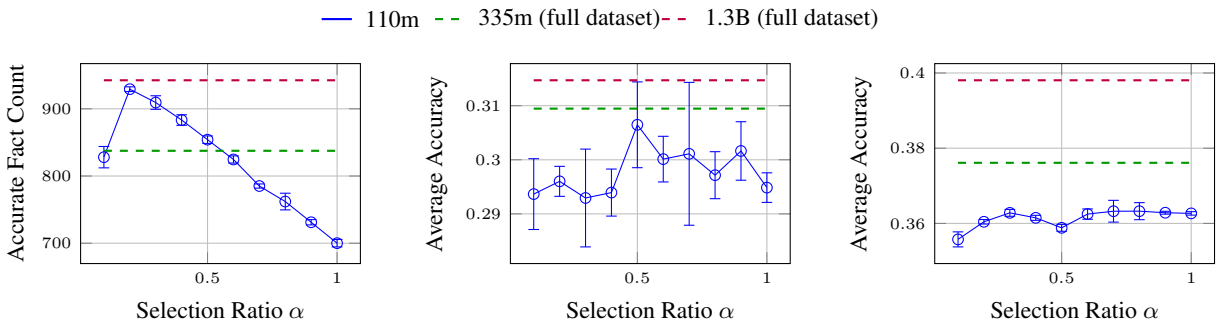
```
Pterostylis stricta was first described in <|start_of_fact|>1972<|end_of_fact|> by <|
    start_of_fact|>Stephen Clemesha and Bruce Gray<|end_of_fact|> and the description
    ...
```

We view each (context, fact tokens) pair in this dataset as one fact. E.g., in the above record, the first fact is (Q_1, A_1) where Q_1 is the context string "Pterostylis stricta was first described in " and the answer A_1 is the string "<|start_of_fact|>1972<|end_of_fact|>". We use the same train, test and validation splits as Zhao et al. (2025b), and show the number of records and facts in each split in Table 4, where on average each record contains around 10 facts.

Adapting Our Selection Algorithm to General Corpus Our selection algorithm Algorithm 1 operates at the unit of fact, i.e., it implicitly assumes each record x corresponds to one fact. In the general pretraining corpus, however, facts and non-fact natural language are mixed in each record (Wikipedia article), and thus requires modifying Algorithm 1 to correctly operate at the fact-level, while not affecting the learning of the remaining non-fact parts of the data record. To this end, we use a variant of Algorithm 1 for our Wikipedia experiments, presented in Algorithm 2. Notably, to select at the fact-level, we compute per-fact loss and select at the unit of fact answers. To avoid adversely affecting the learning of non-fact content, we keep all tokens that are not fact answers, while increasing the weight of tokens in selected fact answers to keep the relative weight between fact and non-fact tokens unchanged before and after selection.

Evaluation metrics We evaluate three setsof performance metrics: (1) fact accuracy on the test split of Wikipedia corpus, measured under stochastic decoding following Definition 3.4, i.e., by the probability of generating A_i given Q_i as context (under stochastic decoding) averaged over all annotated facts $(Q_i, A_i)_{i=1,2,\ldots}$ in the test split. High test fact accuracy shows that the model not only memorizes the facts in the training dataset, but also successfully uses that knowledge to answer facts accurately in fresh test cases. As a sanity check, we also evaluate the train fact accuracy in Fig. 8, where the trend is similar. (2) Knowledge-MMLU accuracy measured as the average accuracy over a subset of MMLU tasks that target world knowledge, including high-schoool-us-history, high-school-european-history, world-religions, clinical-knowledge, global-facts, human-aging, medical-genetics, nutrition, virology, high-school-geography, and human-sexuality. (3) General capability accuracy measured by the average accuracy over a set of standard NLU tasks following Zhao et al. (2025b), including CommonsenseQA, HellaSwag, PIQA, Social IQA, ARC-Easy.

Results We pretrain small language models with 110m parameters from scratch on the annotated Wikipedia Corpus for around 8 epochs, using the same hyperparameters as Zhao et al. (2025b). Our results for 110m model under full



(a) Fact accuracy (in accurate fact count) on Test Split (containing 7135 facts)

(b) MMLU-Knowledge Accuracy

(c) NLU-Accuracy (CommonsenseQA, HellaSwag, PIQA, SIQA, ARC-Easy)

Figure 3: LM lmlmwiki loss experiments across different selection ratio α values. Error bars show standard deviation across 3 repeated training runs.

data selection ($\alpha = 1$) matches the performance reported for 124m model in Zhao et al. (2025b, Table 12). See Appendix D.3 for the detailed setups.

Our main observation is that that our LossH-Wiki selection Algorithm 2 improves fact accuracy (Fig. 3a) and downstream MMLU-Knowledge accuracy (Fig. 3b) without harming general natural language understanding (Fig. 3c). Namely, Fig. 3a shows that our LossH-Wiki selection significantly improves the test fact accuracy compared to full dataset training ($\alpha = 1.0$, with the optimal selection ratio being $\alpha = 0.2$. In Fig. 3b, we observe similar improvement in MMLU-Knowledge accuracy under dataset selection ratio $\alpha = 0.5$ compared to full dataset training ($\alpha = 1.0$), although the improvement is more noisy and smaller in scale, potentially due to the limited number of evaluation questions in MMLU subtasks that target world knowledge. As a sanity check, we also evaluate on the full MMLU benchmark in Fig. 8, where the signal becomes even more noisy, potentially because the remaining subtasks in MMLU require a mix of knowledge and reasoning capabilities, thus diffusing the fact accuracy improvements coming from our selection algorithms. We comment that high variation of Q&A accuracy is widely observed in the literature, and it is a long-standing challenge to evaluate knowledge capabilities of language model in an accurate and stable manner. Lastly, for natural language understanding tasks, Fig. 3c validates that our data selection does not harm the general capabilities of trained model: NLU accuracy remains stable across different selection ratio, except for exceedingly small selection ratio $\alpha = 0.1$ that are not used in practical training.

To understand the scale of improvement, we additionally pretrain two larger models with 335m and 1.3B parameters on the full annotated Wikipedia Corpus using the same hyperparameters, and show their performances in Fig. 3 (dashed lines). As expected, these larger models generally have stronger performances than the small 110m model that is also trained on the full Wikipedia Corpus (i.e., $\alpha = 1.0$). However, this gap shrinks significantly under our data selection schemes: the test fact accuracy of 110m models trained with selection ratio $\alpha = 0.2$ matches the test fact accuracy of a 10X larger 1.3B model trained on full dataset, and the MMLU-Knowledge accuracy of 110m model trained with selection ratio $\alpha = 0.5$ is within standard deviation to that of a 3X larger 335m model trained on full dataset. This shows that the improvements coming from our data selection algorithms are significant.

5 Conclusion

We study how to increase the number of facts that language model can memorize. We prove new connections between fact memorization and fact accuracy, and use it to establish the capacity limit of fact accuracy for language models. We propose loss-based data selection algorithms that boosts fact accuracy to the capacity limit, even when standard training on the full dataset yields sub-optimal (as low as zero) fact accuracy. Our findings challenge the prevailing dogma of indiscriminate data scaling. By demonstrating that a 110M parameter model can match the fact accuracy of a 1.3B model through targeted training data pruning, we suggest that future pretraining laws must account for data efficiency and redundancy, not just data quantity. This opens new avenues for parameter-efficient pretraining via capacity-aware training data selection.

Our work suggests several directions for future research. In gradient-based training, memorization of new information often harms the memorization of old information, i.e., cause catastrophic forgetting. It is conceivable that by adapting our selection schemes to alleviate forgetting in training, e.g., via data replaying Buzzega et al. (2020); Verwimp et al. (2021); Li et al. (2025), the fact memorization speed could be further improved. We also only consider standalone dense transformer model throughout the paper, it is another interesting direction to boost fact memorization capacity and efficiency via adapting model architectures, e.g., via mixture-of-expert (MOE) Jiang et al. (2024); Jelassi et al. (2024) or specialized memory architectures Cheng et al. (2026); Pouransari et al. (2025); Weston et al. (2014). Finally, our selection operates at the fact-level, and requires knowledge for the fact boundaries (such as annotation for fact tokens in the Wikipedia Corpus provided by Zhao et al. (2025b)), while many other real-world datasets may not have clear fact formats or boundaries. It remains a challenge to design the right selection unit for boosting fact memorization on more general real-world datasets.

REFERENCES

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

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.3, knowledge capacity scaling laws. *arXiv preprint arXiv:2404.05405*, 2024.

Idan Attias, Gintare Karolina Dziugaite, Mahdi Haghifam, Roi Livni, and Daniel M Roy. Information complexity of stochastic convex optimization: Applications to generalization and memorization. *arXiv preprint arXiv:2402.09327*, 2024.

BIG bench authors. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1533–1544, 2013.

Stella Biderman, Usvsn Prashanth, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raff. Emergent and predictable memorization in large language models. *Advances in Neural Information Processing Systems*, 36:28072–28090, 2023a.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023b.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, 2020.

Gavin Brown, Mark Bun, Vitaly Feldman, Adam Smith, and Kunal Talwar. When is memorization of irrelevant training data necessary for high-accuracy learning? In *Proceedings of the 53rd annual ACM SIGACT symposium on theory of computing*, pp. 123–132, 2021.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX security symposium (USENIX security 19)*, pp. 267–284, 2019.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *2022 IEEE symposium on security and privacy (SP)*, pp. 1897–1914. IEEE, 2022a.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*, 2022b.

Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX security symposium (USENIX Security 23)*, pp. 5253–5270, 2023.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*, 2017.

Xin Cheng, Wangding Zeng, Damai Dai, Qinyu Chen, Bingxuan Wang, Zhenda Xie, Kezhao Huang, Xingkai Yu, Zhewen Hao, Yukun Li, et al. Conditional memory via scalable lookup: A new axis of sparsity for large language models. *arXiv preprint arXiv:2601.07372*, 2026.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*, 2018.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. *arXiv preprint arXiv:2010.00904*, 2020.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pp. arXiv–2407, 2024.

Logan Engstrom, Axel Feldmann, and Aleksander Madry. Dsdm: Model-aware dataset selection with datamodels. *arXiv preprint arXiv:2401.12926*, 2024.

Simin Fan, Matteo Pagliardini, and Martin Jaggi. Doge: Domain reweighting with generalization estimation. *arXiv preprint arXiv:2310.15393*, 2023.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd annual ACM SIGACT symposium on theory of computing*, pp. 954–959, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891, 2020.

Vitaly Feldman, Guy Kornowski, and Xin Lyu. Trade-offs in data memorization via strong data processing inequalities. *arXiv preprint arXiv:2506.01855*, 2025.

Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. Entities as experts: Sparse memory access with entity supervision. *arXiv preprint arXiv:2004.07202*, 2020.

Jonas Geiping, Alex Stein, Manli Shu, Khalid Saifullah, Yuxin Wen, and Tom Goldstein. Coercing llms to do and reveal (almost) anything. *arXiv preprint arXiv:2402.14020*, 2024.

David Grangier, Simin Fan, Skyler Seto, and Pierre Ablin. Task-adaptive pretrained language models via clustered-importance sampling. *arXiv preprint arXiv:2410.03735*, 2024.

Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. Olmo: Accelerating the science of language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 15789–15809, 2024.

Xinran Gu, Kaifeng Lyu, Jiazheng Li, and Jingzhao Zhang. Data mixing can induce phase transitions in knowledge acquisition. *arXiv preprint arXiv:2505.18091*, 2025.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Few-shot learning with retrieval augmented language models. *arXiv preprint arXiv:2208.03299*, 1(2):4, 2022.

Samy Jelassi, Clara Mohri, David Brandfonbrener, Alex Gu, Nikhil Vyas, Nikhil Anand, David Alvarez-Melis, Yuanzhi Li, Sham M Kakade, and Eran Malach. Mixture of parrots: Experts improve memorization more than reasoning. *arXiv preprint arXiv:2410.19034*, 2024.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

Jaehun Jung, Seungju Han, Ximing Lu, Skyler Hallinan, David Acuna, Shrimai Prabhumoye, Mostafa Patwary, Mohammad Shoeybi, Bryan Catanzaro, and Yejin Choi. Prismatic synthesis: Gradient-based data diversification boosts generalization in llm reasoning. *arXiv preprint arXiv:2505.20161*, 2025.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pp. 10697–10707. PMLR, 2022.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In *International conference on machine learning*, pp. 15696–15707. PMLR, 2023.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pp. 6769–6781, 2020.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*, 2019.

Andrei N Kolmogorov. Three approaches to the quantitative definition ofinformation'. *Problems of information transmission*, 1(1):1–7, 1965.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8424–8445, 2022.

Jeffrey Li, Mohammadreza Armandpour, Seyed Iman Mirzadeh, Sachin Mehta, Vaishaal Shankar, Raviteja Vemulapalli, Samy Bengio, Oncel Tuzel, Mehrdad Farajtabar, Hadi Pouransari, et al. Tic-lm: A web-scale benchmark for time-continual llm pretraining. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32231–32273, 2025.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. Halueval: A large-scale hallucination evaluation benchmark for large language models. *arXiv preprint arXiv:2305.11747*, 2023.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7602–7635, 2024.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, Weizhu Chen, et al. Not all tokens are what you need for pretraining. *Advances in Neural Information Processing Systems*, 37:29029–29063, 2024.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. *arXiv preprint arXiv:2312.15685*, 2023.

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.

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 *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9802–9822, 2023.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*, 2018.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100, 2023a.

Sewon Min, Weijia Shi, Mike Lewis, Xilun Chen, Wen-tau Yih, Hannaneh Hajishirzi, and Luke Zettlemoyer. Nonparametric masked language modeling. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 2097–2118, 2023b.

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pp. 15630–15649. PMLR, 2022.

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? *arXiv preprint arXiv:2505.24832*, 2025.

Sasi Kumar Murakonda, Reza Shokri, and George Theodorakopoulos. Quantifying the privacy risks of learning high-dimensional graphical models. In *International Conference on Artificial Intelligence and Statistics*, pp. 2287–2295. PMLR, 2021.

NICK007X. arxiv-papers. `huggingface.co/datasets/nick007x/arxiv-papers`, 2025. URL `huggingface.co/datasets/nick007x/arxiv-papers`. Accessed: Dec 9 2025.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics. *arXiv preprint arXiv:2104.13346*, 2021.

Aaron Parisi, Yao Zhao, and Noah Fiedel. Talm: Tool augmented language models. *arXiv preprint arXiv:2205.12255*, 2022.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 2463–2473, 2019.

Hadi Pouransari, David Grangier, C Thomas, Michael Kirchhof, and Oncel Tuzel. Pretraining with hierarchical memories: separating long-tail and common knowledge. *arXiv preprint arXiv:2510.02375*, 2025.

Ronak Pradeep, Kai Hui, Jai Gupta, Adam Lelkes, Honglei Zhuang, Jimmy Lin, Donald Metzler, and Vinh Tran. How does generative retrieval scale to millions of passages? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 1305–1321, 2023.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*, 2020.

Noveen Sachdeva, Benjamin Coleman, Wang-Cheng Kang, Jianmo Ni, Lichan Hong, Ed H Chi, James Caverlee, Julian McAuley, and Derek Zhiyuan Cheng. How to train data-efficient llms. *arXiv preprint arXiv:2402.09668*, 2024.

Sunny Sanyal, Hayden Prairie, Rudrajit Das, Ali Kavis, and Sujay Sanghavi. Upweighting easy samples in fine-tuning mitigates forgetting. *arXiv preprint arXiv:2502.02797*, 2025.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551, 2023.

Avi Schwarzschild, Zhili Feng, Pratyush Maini, Zachary Lipton, and J Zico Kolter. Rethinking llm memorization through the lens of adversarial compression. *Advances in Neural Information Processing Systems*, 37:56244–56267, 2024.

Agam Shah, Liqin Ye, Sebastian Jaskowski, Wei Xu, and Sudheer Chava. Beyond the reported cutoff: Where large language models fall short on financial knowledge. *arXiv preprint arXiv:2504.00042*, 2025.

Nazar Shmatko, Alex Adam, and Paul Esau. Gptzero finds 100 new hallucinations in neurips 2025 accepted papers, January 2026. URL `https://gptzero.me/news/neurips/`. Accessed: 2026-01-26.

Thomas Steinke and Lydia Zakynthinou. Reasoning about generalization via conditional mutual information. In *Conference on Learning Theory*, pp. 3437–3452. PMLR, 2020.

Jiashu Tao and Reza Shokri. (token-level) informia: Stronger membership inference and memorization assessment for llms. *arXiv e-prints*, pp. arXiv–2510, 2025.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems*, 35:21831–21843, 2022.

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.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. D4: Improving llm pretraining via document de-duplication and diversification. *Advances in Neural Information Processing Systems*, 36:53983–53995, 2023.

Grant Van Horn and Pietro Perona. The devil is in the tails: Fine-grained classification in the wild. *arXiv preprint arXiv:1709.01450*, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Eli Verwimp, Matthias De Lange, and Tinne Tuytelaars. Rehearsal revealed: The limits and merits of revisiting samples in continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9385–9394, 2021.

Jiachen Tianhao Wang, Tong Wu, Dawn Song, Prateek Mittal, and Ruoxi Jia. Greats: Online selection of high-quality data for llm training in every iteration. *Advances in Neural Information Processing Systems*, 37:131197–131223, 2024a.

Peiqi Wang, Yikang Shen, Zhen Guo, Matthew Stallone, Yoon Kim, Polina Golland, and Rameswar Panda. Diversity measurement and subset selection for instruction tuning datasets. *arXiv preprint arXiv:2402.02318*, 2024b.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *arXiv preprint arXiv:1410.3916*, 2014.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36:34201–34227, 2023.

Jiayuan Ye, Aadyaa Maddi, Sasi Kumar Murakonda, Vincent Bindschaedler, and Reza Shokri. Enhanced membership inference attacks against machine learning models. In *Proceedings of the 2022 ACM SIGSAC conference on computer and communications security*, pp. 3093–3106, 2022.

Jiayuan Ye, Anastasia Borovykh, Soufiane Hayou, and Reza Shokri. Leave-one-out distinguishability in machine learning. *arXiv preprint arXiv:2309.17310*, 2023.

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. Generate rather than retrieve: Large language models are strong context generators. *arXiv preprint arXiv:2209.10063*, 2022.

Zichun Yu, Spandan Das, and Chenyan Xiong. Mates: Model-aware data selection for efficient pretraining with data influence models. *Advances in Neural Information Processing Systems*, 37:108735–108759, 2024.

Sajjad Zarifzadeh, Philippe Liu, and Reza Shokri. Low-cost high-power membership inference attacks. *arXiv preprint arXiv:2312.03262*, 2023.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual memorization in neural language models. *Advances in Neural Information Processing Systems*, 36:39321–39362, 2023.

Renfei Zhang, Manasa Kaniselvan, and Niloofar Mireshghallah. Reinforcement learning improves traversal of hierarchical knowledge in llms. *arXiv preprint arXiv:2511.05933*, 2025.

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

Linxi Zhao, Sofian Zalouk, Christian K Belardi, Justin Lovelace, Jin Peng Zhou, Kilian Q Weinberger, Yoav Artzi, and Jennifer J Sun. Pre-training large memory language models with internal and external knowledge. *arXiv preprint arXiv:2505.15962*, 2025b.

Xiangxin Zhu, Dragomir Anguelov, and Deva Ramanan. Capturing long-tail distributions of object subcategories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 915–922, 2014.

Nicolas Zucchet, Jörg Bornschein, Stephanie Chan, Andrew Lampinen, Razvan Pascanu, and Soham De. How do language models learn facts? dynamics, curricula and hallucinations. *arXiv preprint arXiv:2503.21676*, 2025.

A Related works

Memorization Definition There is a long line of literature on how to meaningfully define memorization for general machine learning algorithms, i.e., the information that a trained model contains about its training dataset. In the natural language modeling literature, such memorization is often intuitively defined by the probability of generating the complete training data given its prefix tokens Carlini et al. (2019; 2022b); Tirumala et al. (2022); Biderman et al. (2023b;a). However, such memorization definitions are often criticized by its overly small coverage, i.e., only capturing the model's prediction behavior on the exact prefix tokens of training data. To address this limitation, recent works Schwarzschild et al. (2024); Morris et al. (2025) relaxes the definition and define memorization by the shortest context required to generate a training data, relating to data compression and Kolmogorov complexity Kolmogorov (1965). However, such memorization definitions still suffer from lack of interpretability due to the fact that a language model can essentially predict any target string, given sufficient manipulations to the context Geiping et al. (2024). Consequently, such measurements are typically only considered meaningful when performed on training data that contains unique information, e.g., personal identifiable information Carlini et al. (2019; 2023); Lukas et al. (2023); Biderman et al. (2023b), or when they are compared to measurements on leave-one-out models Feldman (2020); Feldman & Zhang (2020); Zhang et al. (2023); Ye et al. (2023) trained on the remaining training dataset (to calibrate for shared common knowledge among data, in essence equivalent to measuring the success of membership inference attacks Murakonda et al. (2021); Carlini et al. (2022a); Ye et al. (2022); Zarifzadeh et al. (2023); Tao & Shokri (2025) in controlled data sampling setups).

Recently, a growing line of works Steinke & Zakynthinou (2020); Brown et al. (2021); Attias et al. (2024); Feldman et al. (2025); Morris et al. (2025) focus on information-theoretic definitions for memorization via the (conditional) mutual information between the training dataset and the trained model, partially due to its intuitive interpretation in terms of the bits of information. Independently, Allen-Zhu & Li (2023; 2024) derives information-theoretical connections between a loss-based memorization definition and the bit precision and number of parameters of language models. Morris et al. (2025); Gu et al. (2025) further prove similar connections between the mutual-information-based memorization definition and the bit precision and number of parameters of language models. Our work builds on this line of works, but go beyond (total) memorization to establish a new definition of fact memorization, so as to analyze capacity limit of fact accuracy for a language model, i.e., how many facts can a language model correctly answer.

Memorization Capacity Limit Analysis Our work theoretically analyzes and experimentally validates the fact accuracy capacity limit of language model via controlled experiments on synthetic fact memorization benchmarks. We directly build up on a recent line of works that analyzes and measures memorization capacity for language model training on randomly constructed synthetic datasets Allen-Zhu & Li (2024); Zucchet et al. (2025); Gu et al. (2025); Morris et al. (2025); Jelassi et al. (2024). To the best of our knowledge, all of the aforementioned works focus on relating memorization to training loss, while we prove new connections between (fact) memorization and the stricter metric of fact accuracy that is more suitable for measuring performance of fact-learning. Additionally, prior works focus on random facts with uniform frequency distribution Allen-Zhu & Li (2024); Morris et al. (2025); Jelassi et al. (2024), and only perform passive analysis of memorization speed under power-law distributed facts Gu et al. (2025); Zucchet et al. (2025). By contrast, we propose to *actively* manipulate the training data distribution via loss-based data selection (that limits the number of facts in the training data and flattens their frequency distribution). Notably, we observe that data selection can boost fact accuracy to the (newly derived) capacity limit for training language models across a range of synthetic data distributions. Moreover, our data selection schemes apply to pretraining on general Wikipedia corpus, and effectively boosts fact accuracy and knowledge-intensive downstream task performances without harming natural language understanding. We also comment that although Gu et al. (2025); Jelassi et al. (2024) propose other data paraphrasing/augmentation/mixing techniques to improve fact accuracy, their techniques are heavily dependent on the precise knowledge of the fact formats in their synthetic setups. By contrast, our propose data selection technique is solely based on loss, thus more generally applying to facts in real-world datasets such as arXiv papers and Wikipedia datasets, as shown in our experiments.

Parametric Memory of Language Models Parametric memory Roberts et al. (2020); Févry et al. (2020); De Cao et al. (2020); Yu et al. (2022); Kandpal et al. (2023); Meng et al. (2022) refers to the practice of using a language model to answer questions in a closed-book manner, without access to any external information. This intuitively, requires a language model to "encode knowledge in its parameters", and has been a long-standing topic in understanding the knowledge capabilities of language models Petroni et al. (2019); Jiang et al. (2020); Chowdhery et al. (2023); Shah et al. (2025). Such parametric memory, is often evaluated by closed-book Q&A performance on knowledge-based tasks for short-form generations Wei et al. (2024); Kwiatkowski et al. (2019); Berant et al. (2013); Joshi et al. (2017); Mallen et al. (2023); bench authors (2023), or by factuality performance for long-form generations Min et al. (2023a); Li et al. (2023). It has long been observed Kandpal et al. (2023); Mallen et al. (2023) that the parametric memory of language model is limited when trained on long-tailed data distributions, especially in its knowledge of rare facts. However, to the best of our knowledge, most of the prior investigations are experimentally conducted under fixed training algorithms and real-world datasets, where there are many confounding factors that hinders understanding whether the limited parametric memory is theoretically inevitable, or is only a consequence of suboptimal training algorithms or suboptimal training dataset. By contrast, we theoretically and experimentally analyze the capacity limit of fact accuracy under the more controlled settings of synthetic long-tailed fact datasets. Our controlled investigations show that the reason for limited fact accuracy is largely due to suboptimal training data selection. Importantly, we show that simple loss-based data selection schemes can boost fact accuracy to the capacity limit even when standard training on the full data distribution is high suboptimal and only achieves close-to-zero fact accuracy.

Contextual Memory of Language Models Although our paper focus on boosting the parametric memory of language model, we comment that an orthogonal line of effort to address limited parametric memory is to use contextual memory Berant et al. (2013); Chen et al. (2017); Radford et al. (2019), where the information is fed to the model as input context. To this end, contextual memory predominantly requires a language model's reading comprehension capabilities, thus largely alleviating the burden of memorizing all knowledge in model parameters. The most prevalent examples of contextual memory include retrieval augmented generation (RAG) Khandelwal et al. (2019); Karpukhin et al. (2020); Izacard et al. (2022); Min et al. (2023b) and tool-calling Schick et al. (2023); Jin et al. (2025); Parisi et al. (2022). The performance and efficiency of using contextual memory crucially relies on the quality and quantity of external information sources, besides the language model's knowledge as well as reading comprehension capabilities. Due to the numerous confounding factors, it is often tricky to separate out different factors that affect contextual memory. In this paper, we solely focus on boosting language model's parametric memory, which is a an orthogonal direction to improving usability of contextual memory – it is often observed Jin et al. (2025); Zhang et al. (2025) that the performance and efficiency of knowledge-intensive Q&A often improves as the parametric memory of the base model becomes stronger. We show through experiments that our insights generalize to real-world dataset, e.g., boosting the fact accuracy and accuracy of knowledge downstream tasks for pretraining from scratch on Wikipedia corpus.

Data Selection for Language Model One key ingredient of our data selection technique is to select and flatten the head of the distribution. Intuitively, our head selection step has a similar effect to a long line of works in the literature for selecting data that are the most aligned to end tasks Xie et al. (2023); Fan et al. (2023); Xia et al. (2024); Grangier et al. (2024); Wang et al. (2024a), and our flattening operation is intuitively similar to data de-duplication Lee et al. (2022); Kandpal et al. (2022) and diversification Tirumala et al. (2023); Jung et al. (2025); Wang et al. (2024b); Sachdeva et al. (2024); Liu et al. (2023). However, the aforementioned works largely neglects the question of the right amount of data to subselect by treating it as an hyperparameter tuning problem in experiments. By contrast, one key contribution of our paper is to provide understanding on the right amount of data to select from the model capacity limit perspective. In other words, the optimal selection for small model is intrinsically different to that of large model, which is a perspective missing in the prior data selection literature.

In terms of selection score, we use loss as proxy to approximate the weights and bits of information in training data. The signal of loss by itself, is also widely used in the data selection literature to approximate sample alignment, learnability, difficulty, and diversity Lin et al. (2024); Mindermann et al. (2022); Yu et al. (2024); Engstrom et al. (2024); Li et al. (2024); Sanyal et al. (2025). Different from the prior data selection objectives, our data selection algorithms maximizes fact accuracy within the fact memorization capacity limit. Importantly, for training on synthetic power-law distributed random phonebook facts, we are able to experimentally show that our data selection scheme is near-optimal for the fact accuracy objective, effectively boosting fact accuracy to close to the (2 bits/parameters) capacity limit.

Lastly, we comment that interestingly, our selection strategy of prioritizing low loss facts is intuitively opposite to the seminal Rho-1 Lin et al. (2024) and Rho-Loss Mindermann et al. (2022) algorithms (that selects data with high excess loss relative to a reference model). As our selection is experimentally near-optimal for memorizing independent facts, this suggests that the benefits of Rho-1 Lin et al. (2024) and Rho-Loss Mindermann et al. (2022) are likely not due to better fact memorization. Indeed, these works typically evaluate more complex learning tasks, which not only requires fact memorization but also requires generalization and even reasoning, making selecting low-loss samples not necessarily ideal due to over-fitting. We leave it as an interesting open question, as to what data selection schemes achieve the ideal trade-offs between fact accuracy and other model capabilities.

B Additional Discussions on Memorization Definitions

B.1 Deferred Proof for Fact Accuracy Capacity Limit

We first show how to prove the following proposition that says memorization is upper bounded by the size of the algorithm's output space.

Proposition 1 (Fact Memorization Capacity Limit under Fixed Model Capacity)**.** *For any facts $(Q_i, A_i(\theta))_{i=1}^n$, any meta prior Ψ and any dataset size n, the fact memorization of a learning algorithm $\mathcal{A}$ with discrete output model parameters space $\mathcal{W}$ satifies the following upper bound.*

$$\underset{(Q_i,A_i)_{i=1}^N}{Mem}\ (\mathcal{A};\Psi,n) \le \ln|\mathcal{W}| \tag{5}$$

where $|\mathcal{W}|$ is the cardinality of discrete model parameters space $\mathcal{W}$.

Proof. By definition,

$$\begin{aligned}\underset{(Q_i,A_i)_{i=1}^N}{\text{Mem}}\ (\mathcal{A};\Psi,n) =& I((A_1(\theta),\cdots,A_N(\theta)),\mathcal{A}(D))\\ =&\text{Ent}(\mathcal{A}(D)) - \text{Ent}(\mathcal{A}(D)|A_1(\theta),\cdots,A_N(\theta))\\ \le&\text{Ent}(\mathcal{A}(D)) \le \ln|\mathcal{W}|\end{aligned}$$

where the last equality is by the fact that that the maximal entropy distribution over discrete space $\mathcal{W}$ is the uniform distribution with entropy $\ln|\mathcal{W}|$. □

We now prove Theorem 1 for relating fact memorization to per-fact accuracy.

Theorem 2 (Fact Memorization Lower Bound by Per-fact accuracy)**.** *For any facts $(Q_i, A_i(\theta))_{i=1}^n$, any meta prior Ψ and any dataset size n, if $A_1(\theta),\cdots,A_N(\theta)$ are independently distributed (over the randomness of sampling data distribution parameters $\theta\sim\Psi$), then we have*

$$\underset{(Q_i,A_i)_{i=1}^N}{Mem}\ (\mathcal{A};\Psi,n) \ge \sum_{i=1}^N \Bigg(\underbrace{\underset{\theta\sim\Psi}{Ent}\left[A_i(\theta)\right]}_{b_i} - \underset{\substack{\mathcal{A}\\ \theta\sim\Psi\\ D\sim\mathcal{P}_\theta^n}}{Ent}\left[I_i\right] - \left(1 - \Pr_{\substack{\mathcal{A}\\ \theta\sim\Psi\\ D\sim\mathcal{P}_\theta^n}}\left[I_i=1\right]\right)\cdot \underset{\substack{\mathcal{A}\\ \theta\sim\Psi\\ D\sim\mathcal{P}_\theta^n}}{Ent}\left[A_i(\theta)\mid I_i=0\right]\Bigg) \tag{6}$$

where $I_i = \mathbf{1}_{f(\mathcal{A}(D);Q_i)=A_i(\theta)}$ is the accuracy indicator of the trained model on fact i, and $f(\mathcal{A}(D);Q_i)$ denotes the prediction by the trained model $\mathcal{A}(D)$ on question Q_i.

Proof. By definition, we have

$$\begin{aligned}\underset{(Q_i,A_i)_{i=1}^N}{\text{Mem}}\ (\mathcal{A};\Psi,n) =& I((A_1(\theta),\cdots,A_N(\theta)),\mathcal{A}(D))\\ =&\left(\sum_{i=1}^N \text{Ent}\left[A_i(\theta)\right]\right) - \text{Ent}\left[A_1(\theta),\cdots,A_N(\theta)|\mathcal{A}(D)\right] &(7)\\ \ge&\sum_{i=1}^N\left(\text{Ent}\left[A_i(\theta)\right] - \text{Ent}(A_i(\theta)|\mathcal{A}(D))\right) &(8)\end{aligned}$$

where equation 7 is by the independence between $A_1, \cdots, A_n$, and equation 8 is by the sub-additivity of entropy, i.e., $\text{Ent}(X_1, \cdots, X_n|Y) \leq \sum_{i=1}^{n} \text{Ent}(X_i|Y)$ for any random variables $X_1, \cdots, X_n, Y$. By further applying Fano's inequality to equation 8, we prove that

$$\underset{(Q_i,A_i)_{i=1}^N}{\text{Mem}}\left(\mathcal{A};\Psi,n\right) \geq \sum_{i=1}^{N}\left(\underbrace{\text{Ent}\left[A_i\right]}_{b_i} - \text{Ent}\left[I_i\right] - \left(1-\Pr\left[I_i=1\right]\right)\cdot \text{Ent}\left[A_i(\theta)|I_i=0\right]\right) \tag{9}$$

where $I_i = \mathbf{1}_{f(\mathcal{A}(D);Q_i)=A_i(\theta)}$ is the accuracy indicator of the trained model in predicting fact i. □

We finally provide proof for Corollary 3.1.

Corollary B.1 (Fact Accuracy Capacity Limit on Fixed-Entropy Random Facts)**.** *As a special case of Theorem 1, if each answer $A_i(\theta)$ follows uniform distribution over a discrete answer domain $\mathcal{M}_i$ for $i = 1, \cdots, N$, and if $\ln|\mathcal{M}_1| = \cdots = \ln|\mathcal{M}_N| = b$, and if $\ln|W| \geq \Omega\left(N \cdot \ln 2\right)$, then the accurate fact count of any learning algorithm $\mathcal{A}$ satisfies*

$$\mathop{\mathbb{E}}_{\theta\sim\Psi}\left[\underset{(Q_i,A_i)_{i=1}^N}{\textit{Acc-Cnt}}\left(\mathcal{A};\theta,n\right)\right] \leq O\left(\frac{\ln|\mathcal{W}|}{b}\right) \tag{10}$$

Proof. By substituting the constant $b_1 = \cdots = b_N = b$ in equation 3 we prove that

$$\begin{aligned}
\underset{(Q_i,A_i)_{i=1}^N}{\text{Mem}}\left(\mathcal{A};\Psi,n\right) \geq& \sum_{i=1}^{N}\left(b\cdot\Pr\left[I_i=1\right] - \text{Ent}\left[I_i\right]\right)\\
\geq& \sum_{i=1}^{N}\left(b\cdot\Pr\left[I_i=1\right] - \ln 2\right)\\
=& b\cdot\mathop{\mathbb{E}}_{\theta\sim\Psi}\left[\underset{(Q_i,A_i)_{i=1}^N}{\text{Acc-Cnt}}\left(\mathcal{A};\theta,n\right)\right] - \ln 2\cdot N
\end{aligned} \tag{11}$$

where the second-to-last inequality is by observing that $\text{Ent}\left[I_i\right] \leq \ln 2$ and $\text{Ent}\left(A_i|I_i=0\right) \leq b_i$ due to the fact that the maximum entropy distribution over a discrete space is the uniform distribution, and the last equality is by observing that $\mathop{\mathbb{E}}_{\theta\sim\Psi}\left[\underset{(Q_i,A_i)_{i=1}^N}{\text{Acc-Cnt}}\left(\mathcal{A};\theta,n\right)\right] = \sum_{i=1}^{N}\Pr\left[I_i=1\right]$ by Definition 3.4 for fact accuracy. Combining Eq. (11) and Eq. (5) suffice to prove the bound equation B.1 in the statement. □

B.2 Prior Loss-based Lower Bounds for (Fact) Memorization

We now discuss prior memorization definitions and loss-based memorization lower bounds.

Prior Loss-based Unintentional Memorization Lower Bound A long line of prior worksBrown et al. (2021); Feldman et al. (2025); Morris et al. (2025) define memorization of a learning algorithm about is input dataset as follows.

Definition B.1 (Memorization)**.** *The memorization of a learning algorithm $\mathcal{A}$ (Definition 3.3) about its input dataset D (Definition 3.2) is defined by the mutual information between the dataset D and the learning algorithm's output $\mathcal{A}(D)$ as follows.*

$$\textit{Mem}_{n,\mathcal{W},\Psi}\left(\mathcal{A}\right) = I(\mathcal{A}(D), D) \tag{12}$$

where the expectation is over random sampling of dataset $D \sim \mathcal{P}_\theta$ from training distribution parameterized by θ, the sampling of data distribution parameters θ from the meta prior Ψ, and the randomness of the learning algorithm $\mathcal{A}$.

Through a clever decomposition technique, prior works prove that if we know the training data $x_1, \cdots, x_n$ of the target model, lower bounding the unintended memorization of a trained model $\hat{\theta}$ becomes as easy as summing its unintended memorization over individual training data x_i. This is the quantity estimated in many prior works Feldman & Zhang (2020); Ye et al. (2023); Morris et al. (2025) up to translations, and is also by definition a lower bound for total memorization, written as follows.

Proposition 2 ((Morris et al., 2025, Proposition 1, Proposition 4, Section 2.3))**.**

$$\mathit{Mem}_{n,\mathcal{W},\Psi}(\mathcal{A}) \gtrapprox \sum_{x\in D}\Big(\ell(\theta_r;x)-\ell(\hat{\theta};x)\Big) \tag{13}$$

where $\gtrapprox$ denotes approximately greater than or equal to, θ_r is a reference model trained on freshly sampled n i.i.d. samples from a data distribution specified by the same underlying parameters θ as dataset D, and ℓ denotes the sum of per-token cross-entropy prediction loss.

However, this lower bound is very small when there is no unintentional memorization, i.e., all memorization are necessary for learning the true data distribution parameters (such as retrieval or in general memory-intensive Q&A task), and is thus insufficient for our problem of fact memorization. For example, for the synthetic power-law phonebook experiments in Section 3.2, the model trained on the input dataset D would have similar loss as the reference model trained on fresh i.i.d. samples from the same fact distribution, as the phone numbers are fixed by the training data distribution and remain unchanged under data resampling.

Prior Loss-based Total Memorization Lower Bound Another set of prior works Allen-Zhu & Li (2024); Gu et al. (2025) propose loss-based lower bounds for total memorization, which more naturally translates to our fact-learning setting. For completeness, below we present the derivations along with translated statements.

Theorem 3 (Memorization Lower Bound by Negative Log Likelihood (Similar to Allen-Zhu & Li (2024); Gu et al. (2025)))**.** *Let $(Q_i, A_i(\theta))_{i=1}^N$ be facts encoded by the data distribution parameterized by θ and the function ϕ, as defined by Definition 3.1. For any fixed $n, \mathcal{W}, \Psi$, if answers $A_1(\theta), \cdots, A_N(\theta)$ are independently distributed over $\theta \sim \Psi$, then we have the below loss-based memorization lower bound for any learning algorithm $\mathcal{A}$*

$$\underset{(Q_i,A_i)_{i=1}^N}{\mathit{Mem}}(\mathcal{A};\Psi,n) \geq \sum_{i=1}^{N}\left(\underbrace{\underset{\theta\sim\Psi}{\mathit{Ent}}[A_i(\theta)]}_{b_i} - \mathop{\mathbb{E}}_{\theta\sim\Psi, D\sim\mathcal{P}_\theta^n, \hat{\theta}\sim\mathcal{A}(D)}\left[-\ln\left(\Pr_f[f(\hat{\theta};Q_i)=A_i(\theta)]\right)\right]\right) \tag{14}$$

where $f(\hat{\theta};Q_i)$ denotes the (possibly randomized) prediction of the trained model $\hat{\theta}$ given prefix Q_i.

Proof. By definition,

$$\begin{aligned}
\underset{(Q_i,A_i)_{i=1}^N}{\mathrm{Mem}}(\mathcal{A};\Psi,n) =& I((A_1(\theta),\cdots,A_N(\theta)),\mathcal{A}(D)) && (15)\\
=& \left(\sum_{i=1}^{N}\underset{\theta\sim\Psi}{\mathrm{Ent}}[A_i(\theta)]\right) - \underset{\theta\sim\Psi, D\sim\mathcal{P}_\theta^n,\hat{\theta}\sim\mathcal{A}(D)}{\mathrm{Ent}}\left[A_1(\theta),\cdots,A_N(\theta)|\hat{\theta}\right] && (16)\\
\geq& \sum_{i=1}^{N}\left(\underset{\theta\sim\Psi}{\mathrm{Ent}}[A_i(\theta)] - \underset{\theta\sim\Psi, D\sim\mathcal{P}_\theta^n,\hat{\theta}\sim\mathcal{A}(D)}{\mathrm{Ent}}\left[A_i(\theta)|\hat{\theta}\right]\right) && (17)\\
=& \sum_{i=1}^{N}\left(\underset{\theta\sim\Psi}{\mathrm{Ent}}[A_i(\theta)] + \sum_a \mathop{\mathbb{E}}_{\theta,\hat{\theta}}\left[\Pr[A_i(\theta)=a|\hat{\theta}]\cdot\ln\left(\Pr[A_i(\theta)=a|\hat{\theta}]\right)\right]\right) && (18)\\
\geq& \sum_{i=1}^{N}\left(\underset{\theta\sim\Psi}{\mathrm{Ent}}[A_i(\theta)] + \sum_a \mathop{\mathbb{E}}_{\theta,\hat{\theta}}\left[\Pr[A_i(\theta)=a|\hat{\theta}]\cdot\ln\left(\Pr_f[f(\hat{\theta};Q_i)=a]\right)\right]\right) && (19)
\end{aligned}$$

where equation 16 is by the independence between the answers $A_1(\theta),\cdots,A_N(\theta)$ over $\theta\sim\Psi$; equation 17 is by $\mathrm{Ent}(X_1,\cdots,X_n|Y) \leq \sum_{i=1}^n \mathrm{Ent}(X_i|Y)$ for any random variables $X_1,\cdots,X_n,Y$; equation 18 is by the definition of conditional entropy; and equation 19 is by using the Gibbs inequality which ensures $\sum_a \Pr[X=a]\cdot\ln\Pr[X=a] \geq \sum_a \Pr[X=a]\cdot\ln(\Pr[Y=a])$ for any (discrete) random variables X and Y. (This is intuitively saying for describing random variable X, the Huffman code optimized for X has the shortest length.)

□

This suggests that we can lower bound memorization via the difference between the entropy of answers generated from random prior, versus the average negative log probability of predicting the right answer for each question on top

of observing the trained model. The prediction function f is specified by the decoding and answer matching process when using the trained language model. In the special case of f given by vanilla stochastic-decoding answering function, equation 3 simplifies to the following loss-based lower bound for total memorization.

Corollary B.2 (Memorization Lower Bound by Loss)**.** *Let $\ell(A_i; \hat{\theta}, Q_i)$ be the sum of per-token cross-entropy loss for using trained model $\hat{\theta}$ to predict answer A_i given prefix Q_i. If the same condition of Theorem 3, then under stochastic decoding, we have the below loss-based memorization lower bound*

$$\underset{(Q_i,A_i)_{i=1}^N}{\mathit{Mem}}(\mathcal{A}; \Psi, n) \geq \sum_{i=1}^{N} \left(\underbrace{\underset{\theta\sim\Psi}{\mathit{Ent}}\left[A_i(\theta)\right]}_{b_i} - \mathop{\mathbb{E}}_{\theta\sim\Psi, D\sim\mathcal{P}_\theta^n, \hat{\theta}\sim\mathcal{A}(D)} \left[\ell\left(A_i; \hat{\theta}, Q_i\right)\right] \right) \tag{20}$$

Estimating the first term b_i, i.e., the entropy of the fact answer $A_i(\theta)$ over $\theta \sim \Psi$, requires accurately approximations for the meta prior Ψ. For synthetically constructed dataset, we have control over the meta distribution ψ and can compute b_i exactly. For real dataset, however, we often do not have precise knowledge of fact entropy, nor about the (existence of) independently distributed facts. In our experiments, we use the **median of model's training loss across first epoch as approximations for the average of b_i over all facts $i = 1, \cdots, N$ under meta prior**. (Besides this heuristic, we also tried other heuristic choices of meta distribution, such as the loss of model at initialization, the loss of a reference model trained on disjoint facts. However, we found these choices severely overestimate memorization, as discussed in Appendix B.3.)

B.3 More Discussions on Meta Prior Choices

Below we discuss two other choices of meta prior, that we found to be overestimating memorization lower bound in our experiments. We identify them as overestimation because the estimated memorization lower bound grows indefinitely with regard to training dataset size despite constrained model size, thus contradicting Theorem 3.

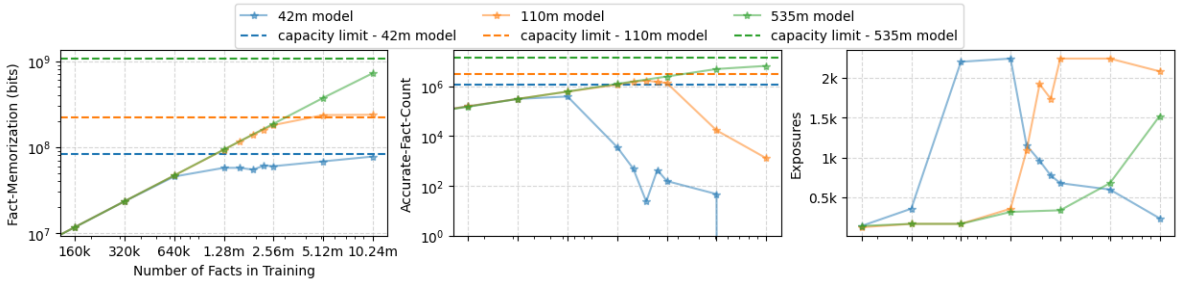
Using Reference Model trained on Disjoint Facts as Meta Prior May Overestimate Memorization due to Overfitting This is similar to the approach in Morris et al. (2025), yet we found that it overestimates memorization in our experiments. This is because reference model trained to memorize a disjoint set of facts could overfit to those facts, resulting in increasing loss on the facts in the target model's training dataset, thus serving as a baseline that overestimates the fact-entropy and memorization about training facts in the target model.

Using Model Initialization as Meta Prior May Overestimate Memorization due to Correlated Format Knowledge Another intuitive approximation in practice, is to heuristically choose the meta distribution Ψ as the pretrained model at random (re)initializations. However, this has the risk of breaking the independence assumption among answers $A_1(\theta), \cdots, A_N(\theta)$ (as one can imagine that the predictions given similar prefix are correlated, even at initialization), and thus overestimating memorization lower bound. For example, for synthetically constructed phonebook dataset that consists of (name, phone-number) tuples (with name being six randomly drawn alphabetical characters from A to Z, and with phone-number being 22 randomly drawn digits from 0 to 9), the average per-token loss at initialization is as high as 8 when the underlying true average per-token entropy of each (name, phone-number) tuple is $(6 \cdot \ln(26) + 22 \cdot \ln(10))/(6+22) = 2.507$. This overestimation is intuitively because there exists shared knowledge among answers for different questions, e.g., about the format.

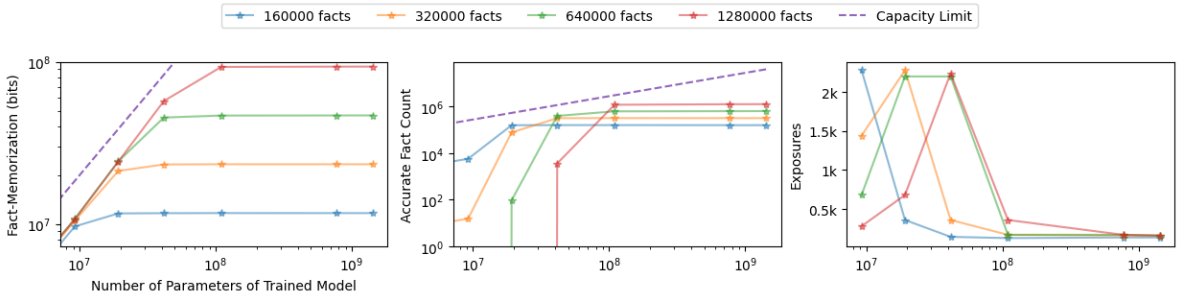
C Deferred Details for Suboptimal fact accuracy Results

C.1 Experiment Setups for Pretraining on Synthetic Phonebook

Pretrained Model Architecture We consider variants of the standard GPT2-style decoder-only transformer models Vaswani et al. (2017) with context length 1024, sinusoidal positional embeddings, ReLU activation, and post-output layer norm. Each model has L layers, H heads, hidden dimension D, and MLP dimension $4D$ in bfloat16 precision. To control model size, we follow Pythia Biderman et al. (2023b) and vary (L, D, H) across (6, 512, 8), (12, 768, 12), (24, 1024, 16), (16, 2048, 8), (24, 2048, 16), (32, 2560, 32) to create a family of model with size ranging from 42m to 1.4B parameters.



(a) varying number of facts in the training dataset



(b) varying number of parameters of the trained model

Figure 4: Fact memorization (in bits), fact accuracy (in accurate fact count), and exposures (needed till convergence) for pretraining on synthetic phonebook dataset. Fact memorization is measured by loss-based lower bound equation B.2. Fact accuracy is measured by accurate fact count defined in Definition 3.4, i.e., the sum of probability of generating the correct answer for each fact under stochastic decoding. Exposure is measured by the number of epochs needed till convergence for training under tuned hyperparameters (Appendix C.1).

Hyperparameter Tuning to Ensure Sufficiently Long Training For all experiments, we train with auto-regressive next-token-prediction cross-entropy loss, and follow prior works Brown et al. (2020); Hoffmann et al. (2022) to use AdamW optimizer with weight decay 0.1 and cosine learning rate scheduler: the learning rate linearly increases from 0 to the maximal learning rate in warm-up steps (2.5% of the training steps), and then decrease to 0.1 times of the maximal learning rate in the remaining training steps following cosine decay. We fix the number of training steps as 800000, and perform extensive grid search for the optimal batch-size over $\{32, 80, 160, 320, 640, 1280, 2560, 5120, 10240\}$, for the optimal learning rate over $\{5e^{-4}, 1e^{-5}, 5e^{-5}, 1e^{-4}\}$. We set the gradient clipping norm as 1.0. To identify the best training run, for each setting, we first find runs that reaches close-to-best performance (within 2% multiplicative difference or 0.01 additive gap to the smallest loss over all runs), and then identify one run among them with the fastest convergences (in terms of smallest number of training tokens seen). We define convergence as reaching within 2% multiplicative difference or 0.01 additive gap to the final performance.

C.2 Additional Fact Memorization Analysis for Pretraining on Synthetic Phonebook

In Fig. 4 (left), we first perform ablation experiments to validate that the language model trained on sufficiently many uniformly distributed facts consistently reach the 2 bit/parameter fact memorization capacity limit (Proposition 1), confirming the observations in prior works Allen-Zhu & Li (2024); Morris et al. (2025); Gu et al. (2025) and validating the effectiveness of our pretraining recipe.

Exposure Needed for Convergence Peaks at Capacity Threshold Conditioned on the effectiveness of our pretraining strategy, we now investigate whether the fact memorization speed is a function of ratio between model size and data complexity. We perform two sets of experiments: in Fig. 4 (a) we fix the model size and vary the training dataset size, and in Fig. 4 (b) we fix the training dataset and vary the model size. We observe in both cases, the exposures (i.e., the number of times that the training passes each fact) needed for convergence peaks at capacity threshold, i.e., when the model size matches the dataset size. On the one hand, this means that when model is too small to fit all training data, training longer would not help the model to converge to better optima. On the other hand, this means

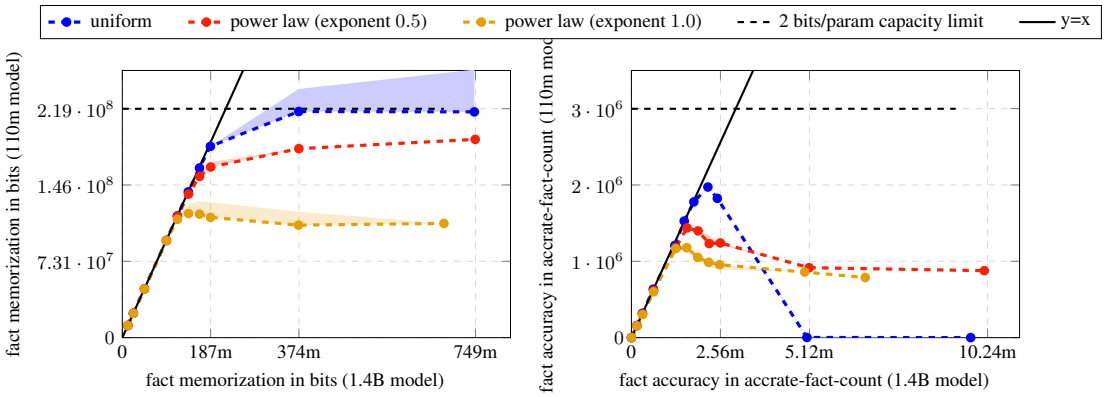


Figure 5: Gap in fact memorization in bits (left) and fact accuracy in accurate fact count (right) between sufficiently trained small model (110m parameters) and 10X larger model (1.3B parameters). Dashed lines show results for the small model trained for 800k steps, and shaded areas (extending above the dashed lines) show the improvement obtained by training the small model for $8\times$ longer (6.4M steps). Each point in the curve shows result for one training dataset, and curves are plotted over increasingly large training datasets containing $\{160000, 320000, 640000, 1280000, 1600000, 1920000, 2240000, 2560000\}$ facts, following different frequency distributions, including uniform, power law with exponent 0.5, and power law with exponent 1.0. All results are for the optimal run after hyperparameter tuning as discussed in Appendix C.1.

that larger models need fewer passes to memorize individual facts. This is intuitively because larger model are easier to optimize and has more capacity, thus reducing the interference during memorizing different facts and leading to faster convergence. standard auto-regressive next-token-prediction cross-entropy loss.

Understanding the Exacerbating Suboptimality of Fact Accuracy under Power-law Training Data In Fig. 5, we observe that the maximal fact memorization and fact accuracy drop significantly under increasing power law exponent. To understand the reason, we conduct two ablation experiments in Fig. 5: **(1) we train a 10x bigger model** (1.4B parameters) for the same number of steps on exactly the same stream of training data; and **(2) we train a small model for 8x longer** (via increasing the number of training steps). We observe that training a 10x larger model significantly improves fact memorization and fact accuracy to near-perfect (Fig. 5 x-axis), while 8x longer training only yields negligible improvements (Fig. 5 shaded area). This implies that larger models may need fewer exposures to memorize each fact compared to small models, which is the key reason for the exacerbating fact memorization and fact accuracy gap between small and large model as the data becomes more non-uniform (i.e., as the power-law exponent increases). We comment that this also evidenced by our experiments in Fig. 4 right plots, and has also observed for other definitions of memorization on real-world datasets in prior works Tirumala et al. (2022).

C.3 Additional Results for Suboptimal Fact Accuracy in LoRA Finetuning

LoRA Finetuning has Similar Memorization Capacity to Pretraining From Scratch We now turn to LoRA finetuning, and investigate its capacity limit for fact memorization and fact accuracy. We first repeat the experiments for synthetic phonebook dataset in the LoRA finetuning settings. As shown in Fig. 6a, the memorized bits is also close to the 2 bits/parameter capacity limit, matching the capacity of pretraining from scratch. This shows that the representation power of LoRA is strong enough to match full transformer model in terms of fact memorization capacity.

Results for Real-World Author-Title Mapping Facts from Arxiv Papers To capture more realistic real-world high-entropy facts, we perform LoRA finetuning of the Llama-3.2-1B pretrained model Dubey et al. (2024) on natural author-title mapping facts in the arXiv-papers NICK007X (2025) dataset (subselecting 171104 articles published in 2025 after the pretrained models' cut-off dates). This is to simulate the setting of teaching a pretrained language model new knowledge that are *not* in its pretraining dataset. We train on data of format "title: ___ | authors: ___" for learning the author-title mapping facts. We choose this type of facts as frontier language models tend to hallucinate author names or paper titles, which is a well-known issue in the scientific community. (E.g., GPTZero finds 100+ confirmed hallucinations in a subset of evaluated 300 NeurIPS 2025 accepted papers Shmatko et al. (2026).) We choose this

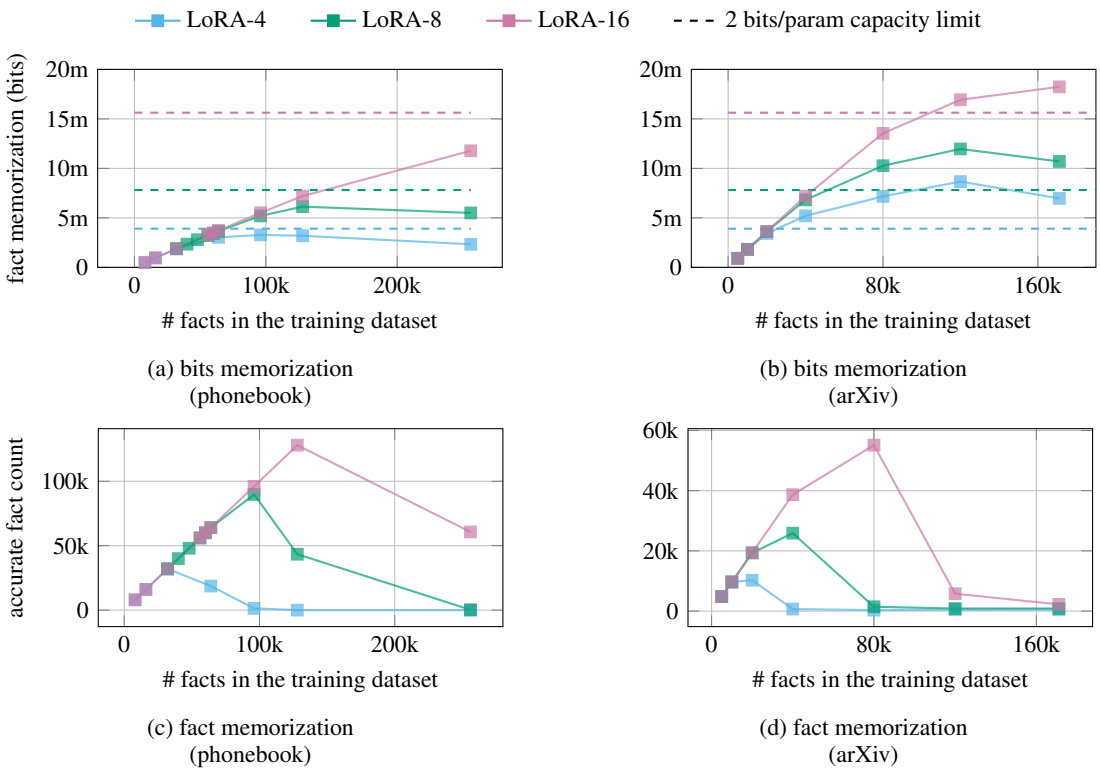


Figure 6: Fact memorization and fact accuracy capacity of LoRA finetuning. See settings in Section 4.1.

close-to-Q&A format because Q&A is observed to be the most effective data format for knowledge injection during finetuning in prior works Zhao et al. (2025a). Note that this dataset is not suitable for pretraining due to its limited size. All LoRA training runs consider a fixed context length of 64 and for hyperparameter tuning, we perform grid search for the number of training steps over $\{2000, 4000\}$, learning rate in $\{2e^{-4}, 5e^{-4}, 1e^{-3}, 2e^{-3}, 5e^{-3}\}$, batch-size in $\{4000, 16000, 64000\}$.

We estimate memorized bits via loss-based memorization lower bound Corollary B.2 and heuristic approximations for the average fact-entropy via median of training loss in the first epoch as discussed after Corollary B.2. We observe in Fig. 6b that our estimates of memorized bits for arXiv-paper dataset is higher than that for the synthetic phonebook dataset. This is potentially due to overestimation of memorized bits under heuristic loss-based approximations for fact-entropy, as discussed in details after Corollary B.2 and in Appendix B.3. We comment that accurately estimating memorized bits for real dataset (where the meta distribution is unknown or even non-existent) is a long-standing challenge in the literature Allen-Zhu & Li (2024); Morris et al. (2025). Nevertheless, our result still shows the promise of reaching the capacity limit of LoRA adapters for memorizing real-world high-entropy facts.

Fact Accuracy Drops to Close-to-Zero when Training Data Exceeds Model Capacity Our final observation is that similar to the results for pretraining (Fig. 4), fact accuracy of LoRA finetuning (on both the synthetic phonebook dataset and the more realistic aXiv-papers datset) also drops significantly to as low as zero, as the number of facts in the training dataset increases to exceeding the model's fitting power. In Appendix D.2, we also further investigate whether fact accuracy of LoRA finetuning can be similarly boosted by our data selection Algorithm 1.

# Facts in Train Data	Full Dataset		Oracle-Aided Selection			Loss-based Selection	
	1x steps	8x steps	Flattened	Head	Head-Flattened	LossH (Ours)	LossHF (Ours)
Power-law Exponent $\beta = 0$							
1.60M	1.53M	1.53M	1.50M	1.55M	1.55M	1.55M	**1.60M**
1.92M	1.78M	1.78M	1.80M	1.81M	1.76M	1.81M	**1.91M**
2.24M	1.90M (457k)	1.90M (457k)	1.94M	1.90M	1.93M (141k)	1.88M	**2.05M**
2.56M	0.99M (635k)	1.85M (136k)	1.10M (645k)	1.83M (356k)	1.93M (525k)	1.88M	**2.04M (914k)**
5.12M	0.01M (26k)	0.01M (5k)	0.09M	**2.06M**	1.95M (471k)	1.79M	1.94M
10.24M	0.00M	0.00M	0.00M	1.97M	1.87M	1.87M	**2.11M**
Power-law Exponent $\beta = 0.5$							
1.60M	1.44M	1.44M	1.55M	1.47M	1.53M	1.47M	**1.60M**
1.92M	1.39M	1.39M	1.80M	1.47M	1.79M	1.52M	**1.90M**
2.24M	1.22M (9k)	1.36M (9k)	1.98M	1.52M	1.76M (706k)	1.57M	**2.15M**
2.56M	1.23M (20k)	1.20M (8k)	1.44M (678k)	1.52M (18k)	1.90M (440k)	1.54M (20k)	**2.24M (44k)**
5.12M	0.93M (20k)	0.93M (20k)	1.02M	1.53M	0.32M (620k)	1.46M	**2.16M**
10.24M	0.82M	0.82M	0.96M	1.51M	**1.64M**	1.32M	1.57M
Power-law Exponent $\beta = 1.0$							
1.60M	1.18M	1.17M	1.54M	1.19M	1.52M	1.13M	**1.58M**
1.92M	1.05M	0.99M	1.80M	1.18M	1.47M	1.10M	**1.82M**
2.24M	0.99M (9k)	0.93M (2k)	**1.95M**	1.18M	1.91M (520k)	1.06M	1.80M
2.56M	0.95M (8k)	0.90M (3k)	1.40M (645k)	1.18M (15k)	**1.83M (489k)**	1.05M (5k)	1.74M (23k)
5.12M	0.86M (10k)	0.85M	0.10M	1.19M	**1.87M (230k)**	1.00M	1.72M
10.24M	0.78M	0.78M	0.70M	1.19M	**1.87M**	0.97M	1.27M

Model Size: 110M parameters
Accurate Fact Count Capacity Limit: $(2\text{bits/param}) \times (110\text{M params})/(22 \times \log_2(10)\ \text{bits/fact}) = 3.01\text{M facts}$

Table 1: Best **fact accuracy** (in accurate fact count, i.e., the expected number of correctly answered facts) for pretraining from scratch on power-law distributed phonebook facts under different data selection schemes. We bold the best result in each row, and show standard deviation across 10 runs in brackets for certain settings where the performances show more variances. All settings consider sufficiently trained model with 110m parameters that trains for 800k steps, except for the 8x longer training runs which train for 6.4m steps. See Appendix C.1 for hyperparameter tuning setups for training on full dataset, and see Section 4.1 for hyperparameter tuning setups for training with data selection.

D Additional Results for Our Selective Training

D.1 Additional Results for Pretraining

In Table 1, we show the detailed performance numbers for how our data selection boosts fact accuracy for pretraining on synthetic power-law phonebook datasets. Observe that fact memorization can be seen as an unweighted average of prediction accuracy on individual facts, it is natural to ask, does data selection also boosts weighted fact accuracy? Indeed in practice, certain facts are more "important" to memorize due to their frequent occurrences in the training dataset. In our synthetic phonebook experiments, such weights are naturally the probability of each fact in the underlying power law distributions. Thus below in Table 2, we present additional results for weighted fact accuracy performance, and show that it similarly improves under data selection.

D.2 Additional Results for LoRA-Finetuning: Boosting Fact Accuracy without Additional Forgetting

We now perform data selection for LoRA finetuning on the semi-synthetic title-author mapping facts in the arXiv-papers dataset NICK007X (2025). Besides the finetuning setups mentioned in Appendix C.3, we additionally tune the selection ratio α grid search over $\alpha \in \{0.1, 0.2, \cdots, 1.0\}$ for our selection algorithms. Our results are summarized in Fig. 7. We observe that compared to training on full dataset, our LossHF selection Algorithm 1 significantly improves the fact accuracy (in accurate fact count) on the arXiv-papers dataset, while achieving similar or better general capability performance (as measured by average accuracy across a set of standard Commonsense Zellers et al. (2019); Mihaylov et al. (2018); Bisk et al. (2020), MMLU Hendrycks et al. (2020), and ARC Clark et al. (2018)

# Facts in Train Data	Full Dataset 1x steps	Full Dataset 8x steps	Oracle-Aided Selection Flattened	Oracle-Aided Selection Head	Oracle-Aided Selection Head-Flattened	Loss-based Selection LossH (Ours)	Loss-based Selection LossHF (Ours)
Power-law Exponent $\beta = 0$							
1.60M	0.942	0.942	0.936	0.962	0.966 (0.004)	0.396	**0.999**
1.92M	0.929	0.929	0.935	0.918	0.917 (0.115)	0.931	**0.996 (0.002)**
2.24M	0.872	0.872	0.864	0.849	0.834 (0.075)	0.841	**0.914 (0.366)**
2.56M	0.677	0.726	0.739	0.735	0.739 (0.262)	0.740	**0.798 (0.340)**
5.12M	0.003	0.002	0.017	**0.402**	0.380 (0.013)	0.349	0.379 (0.117)
10.24M	0.000	0.000	0.000	0.192	0.183 (0.015)	0.183	**0.204 (0.102)**
Power-law Exponent $\beta = 0.5$							
1.60M	0.932	0.932	0.971	0.946	0.958 (0.034)	0.944	**0.998 (0.000)**
1.92M	0.814	0.814	0.935	0.863	0.935 (0.005)	0.855	**0.992 (0.000)**
2.24M	0.692	0.727	0.885	0.796	0.861 (0.141)	0.797	**0.971 (0.001)**
2.56M	0.660	0.629	0.600	0.755	0.785 (0.247)	0.724	**0.905 (0.011)**
5.12M	0.397	0.397	0.421	0.529	**0.569 (0.021)**	0.460	0.566 (0.018)
10.24M	0.276	0.276	0.289	0.373	**0.409 (0.007)**	0.290	0.272 (0.092)
Power-law Exponent $\beta = 1.0$							
1.60M	0.971	0.972	0.969	0.973	0.953 (0.015)	0.970	**0.998 (0.000)**
1.92M	0.952	0.940	0.934	0.960	0.962 (0.007)	0.955	**0.995**
2.24M	0.938	0.927	0.854	0.951	0.948 (0.008)	0.942	**0.980 (0.001)**
2.56M	0.928	0.916	0.684	0.943	0.904	0.933	**0.967 (0.001)**
5.12M	0.883	0.876	0.034	0.902	0.899 (0.007)	0.889	**0.916**
10.24M	0.840	0.840	0.333	**0.865**	0.853 (0.027)	0.850	0.858
Model Size: 110M parameters							

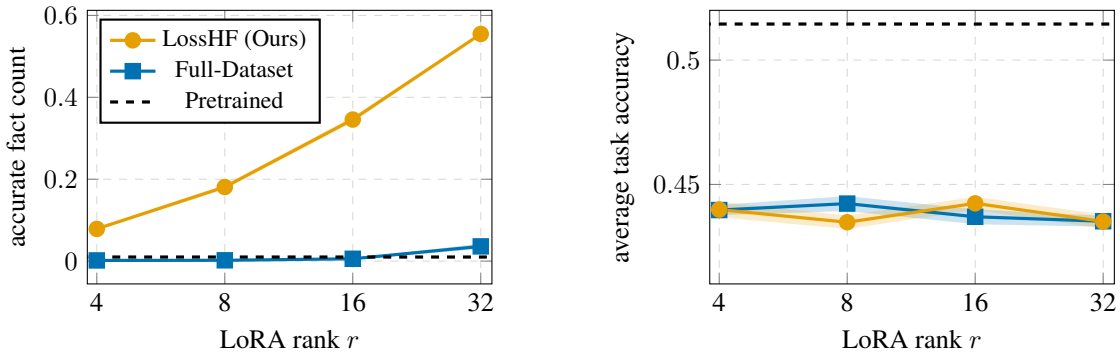
Table 2: Best **weighted** fact accuracy (combined weight of accurately answered facts in the underlying data distribution) for pretraining from scratch on power-law distributed phonebook facts under different data selection schemes. We bold the best result in each row, and show standard deviation across 10 runs in brackets for certain settings where the performances show more variances. All settings consider sufficiently trained model with 110m parameters that trains for 800k steps, except for the 8x longer training runs which train for 6.4m steps. See Appendix C.1 for more details on the setup for training on full dataset, and see Section 4.1 for more details on the setup for training with data selection.

Q&A tasks following Sanyal et al. (2025)). In Appendix D.2, we further show the detailed performance for each task besides their average. The trend remains roughly the same for individual task performance. This shows that our loss-based selection improves fact accuracy without worsening forgetting during finetuning. Intuitively, this is because our selection method increases the occurrences of low-loss facts, which tend to result in smaller scales of gradient updates, thus not worsening forgetting. This is also consistent with the recent work Sanyal et al. (2025) that proposes to upweight low-loss samples in the training objective to reduce catastrophic forgetting during finetuning. However, we comment that the forgetting is still severe in Fig. 7, and even slightly worsens as LoRA rank increases. It remains an interesting open question to alleviate forgetting as much as possible during the finetuning that is needed for memorizing new facts.

In Table 3, we further report the detailed performance for various general capabilities tasks. The trend remains roughly the same for each individual task performance, i.e., our loss-based selection improves fact accuracy without worsening forgetting, when compared to LoRA finetuning on the full dataset.

D.3 Additional Results for Wikipedia Pretraining

Additional Dataset Details and Training Settings We use the same train, test and validation splits as Zhao et al. (2025b), and show the number of records and facts in each split in Table 4, where on average each record contains around 10 facts. We pretrain variants of the GPT2-small (110m parameters), GPT2-medium (335m parameters) and GPT2-large (1.3B parameters) models with context length 1024, sinusoidal positional embeddings, ReLU activation, and post-output layer norm. We follow the hyperparameters in Zhao et al. (2025b) and train for 66000 steps (around 8 epochs) with batch-size 320, using AdamW optimizer with weight decay 0.1 and cosine learning rate scheduler with fixed learning rate 5e-4, warmup steps 2000, and gradient clipping norm 5.0. For our selection Algorithm 2, we tune



(a) Fact Accuracy

(b) General Capabilities Performance

Figure 7: Performance comparison between Full-Dataset Training and our LossHF Selection Algorithm 1 for LoRA finetuning across different ranks on title-authors mapping facts in the arXiv-papers dataset. All performances are averaged across three runs. See Section 4.1 for more details on the settings, and see Table 3 for detailed performances of each general capability task.

Method	**General Capability Accuracy**						**Target Accuracy**
		Commonsense		MMLU	ARC		Fact Accuracy (%)
	Hellaswag	OpenbookQA	PiQA		Easy	Challenge	on arXiv Papers
Pre-trained	63.1 (0.10)	36.7 (0.14)	74.8 (0.09)	30.7 (0.24)	65.6 (0.19)	36.8 (0.10)	0.1
LoRA $r = 4$							
Full Data	54.2 (0.16)	32.7 (0.79)	67.8 (0.36)	24.6 (0.23)	53.6 (0.82)	31.0 (0.37)	0.2 (0.03)
LossHF (Ours)	55.2 (0.17)	34.3 (1.35)	67.7 (0.80)	26.0	50.5 (0.81)	30.3 (0.65)	7.9 (0.41)
LoRA $r = 8$							
Full Data	55.3 (0.39)	34.2 (1.17)	65.9 (0.35)	27.0 (0.78)	51.6 (0.24)	31.4 (0.90)	0.2 (0.06)
LossHF (Ours)	56.2 (0.43)	33.7 (0.74)	67.2 (0.33)	25.0 (0.86)	48.3 (0.83)	30.4 (0.56)	18.1 (0.57)
LoRA $r = 16$							
Full Data	56.3 (0.20)	34.4 (1.25)	65.5 (0.73)	25.8 (0.56)	49.0 (0.81)	31.2 (0.45)	0.6 (0.03)
LossHF (Ours)	56.5 (0.03)	33.7 (0.98)	66.7 (0.95)	26.4 (0.59)	50.7 (0.17)	31.5 (0.36)	34.6 (0.72)
LoRA $r = 32$							
Full Data	55.9 (0.10)	32.9 (0.71)	64.4 (0.25)	26.8 (0.25)	49.5 (0.83)	31.6 (0.67)	3.6 (0.69)
LossHF (Ours)	55.9 (0.17)	33.8 (1.22)	65.1 (0.36)	26.9 (0.58)	48.5 (0.59)	30.9 (0.75)	55.5 (0.20)

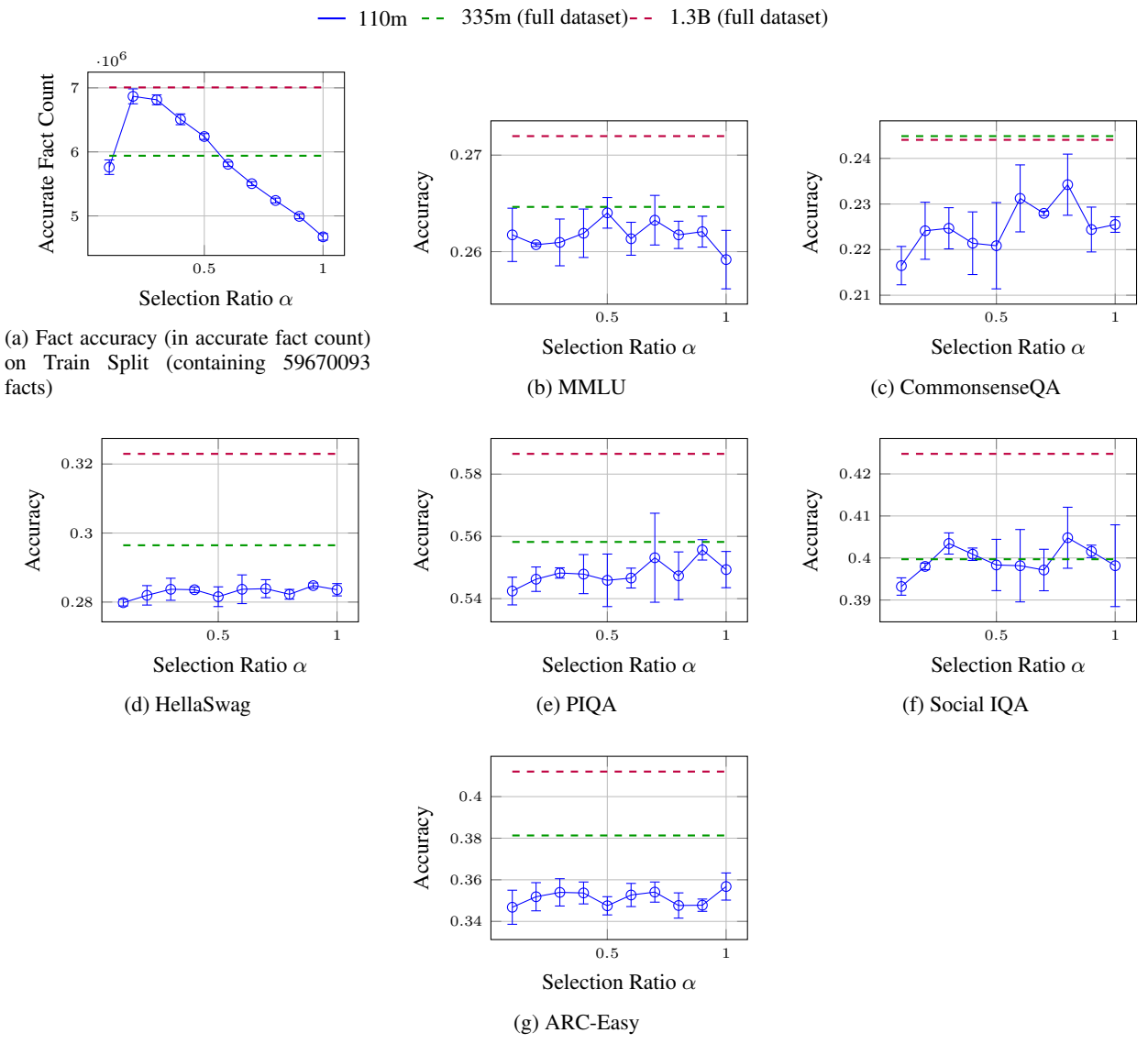
Table 3: Our selective training improves fact accuracy without worsening forgetting during LoRA finetuning on title-authors mapping facts in the arXiv-papers dataset. We show mean and standard deviation (in brackets) across three runs.

the selection ratio to maximize train fact accuracy by grid search over $\alpha \in \{0.1, 0.2, \cdots, 1.0\}$. For small model with 110m parameters, all results are over three training runs for statistical significance.

Detailed performance for Wikipedia Pretraining under our Data Selection Besides the metrics presented in Section 4.2, we now show in Fig. 8 the detailed performance trend for fact accuracy on training split, general MMLU task, as well individual NLU tasks over different selection ratios. Observe that fact accuracy on the training split is significantly improved by data selection, similar to test fact accuracy in Fig. 3a; and MMLU performance is similar in trend to Knowledge-MMLU performance in Fig. 3b, despite being more noisy due to the inclusion of more reasoning or conprehension related tasks besides world knowledge; and similar to the trend of average NLU accuracy in Fig. 3c, the performances of individual NLU tasks (c) to (g) in Fig. 8 remain roughly the same across different selection ratio, except for exceedingly small selection ratio $\alpha = 1$ that are never used in practical training.

Table 4: Fact counts in processed Wikipedia corpus

Split	Facts	Records
train	59670093	6245642
validation	733	75
test	7135	830
Total	59677961	6246547



(a) Fact accuracy (in accurate fact count) on Train Split (containing 59670093 facts)

(b) MMLU

(c) CommonsenseQA

(d) HellaSwag

(e) PIQA

(f) Social IQA

(g) ARC-Easy

Figure 8: Performance details for fact accuracy on Training split, full MMLU, and individual NLU tasks for using our data selection Algorithm 2 in pretraining on annotated Wikipedia Corpus (3B tokens) with 66k steps and batch-size 320 (roughly 8 epochs). We show standard deviation error bars across 3 runs for model with 110m parameters.

E Ablation Experiments and Discussions

In this section, we further perform ablation experiments to understand the inner working of our selection algorithms, as well as discuss design choices and computation cost of our selection algorithms.

E.1 Comparing to Oracle-Aided Head Selection and Flattening

To isolate the benefits of the head selection and the flattening step in our data selection schemes, as well as to understand how accurate is loss as a proxy for the underlying fact frequency, in this section, we compare our selection Algorithm 1 with the following oracle-aided baselines that have precise knowledge of what fact each training record corresponds to, and of the frequency of each fact in the training dataset distribution.

1. **Head**: select $O\left(\frac{\ln|\mathcal{W}|}{b}\right)$ facts in the training dataset with the highest frequency, which is one way to reach the optimal fact accuracy within capacity limit as discussed after Corollary 3.1. This is equivalent to LossH selection in our Algorithm 1, after replacing loss with the inverse of ground-truth frequency of the fact for record x.
2. **Head-Flattened**: on top of Head selection, decreasing the sampling probability for facts with high weights. This is equivalent to LossHF selection in Algorithm 1, after replacing loss with the inverse of ground-truth frequency of the fact for record x.
3. **Flattened:** only reduce the sampling probability for facts with high weights, but does not throw away any tail facts with low weights. This is an baseline algorithm to understand the effect of flattening without head selection, and is equivalent to LossHF selection in Algorithm 1, after replacing loss with the inverse of ground-truth frequency of the fact for record x, and after setting the sampling probability for low-frequency facts to be one instead of zero.

Our results are summarized in Table 1, where for most entries, we show the optimal run after tuning training hyperparameters following Section 3.2 and tuning the selection ratio via grid search over $\alpha \in \{0.1, 0.2, \cdots, 1.0\}$; for settings where the model and training dataset are at capacity of each other, we observe high variance of results across repeated runs with the same hyperparameters (intuitively due to an edge-of-stabilithy phenomenon near capacity threshold), and thus report the median (standard deviation) across 10 repeated runs. Our first observation is that among the three oracle-aided methods, Head-Flattened consistently performs the best, significantly outperforming other oracle-aided baselines including Flattened and Head. The comparison between Flattened and Head shows some nuances: Flattened outperforms Head when the model is sufficiently large to fit all facts in the training dataset (facts$\leq$2560000); by contrast Head outperforms Flattened when the number of facts in the training dataset exceeds model fitting power (facts$\geq$2560000). This validates that both the flattening step (which down-samples facts that appear too frequently) and the head selection step (which removes rare facts that exceed the model's fitting power) are necessary for boosting fact memorization to the capacity limit.

Our second observation is that our LossHF and LossH selection Algorithm 1, despite solely using loss for selection (and not having any prior knowledge on the fact frequency or entropy), consistently performs on par with the oracle-aided Head-Flattened selection and Head selection respectively. LossHF (LossH) are only worse than Head-Flattened (Head) when the training dataset contains an extremely large number of facts that are distributed as a power low with high exponent ($\beta = 1.0$). This validates the ***effectiveness of using loss alone*** to approximately distinguish rare facts versus redundant facts, as also illustrated in the data usage histograms Fig. 9 for various selection methods. In Appendix E.2, we also discuss more nuances in the design of loss-based selection score.

E.2 On Design Choices of Loss-based Selection Score

There are many confounding factors that may affect the quality of loss-based approximation for fact weight, such as the training time, sequence length, and the hardness of each training data sequence. Below we discuss important design choices in our loss-based selection score to adapt to these factors.

Why not select by token-level loss Many data selection methods for language model training operate at the token level, including the celebrated Rho-1 Lin et al. (2024) method. In this paper, we choose to perform selection based on per-record loss (Algorithm 1) or per-fact loss (Algorithm 2) rather than per-token loss, to preserve the boundary of

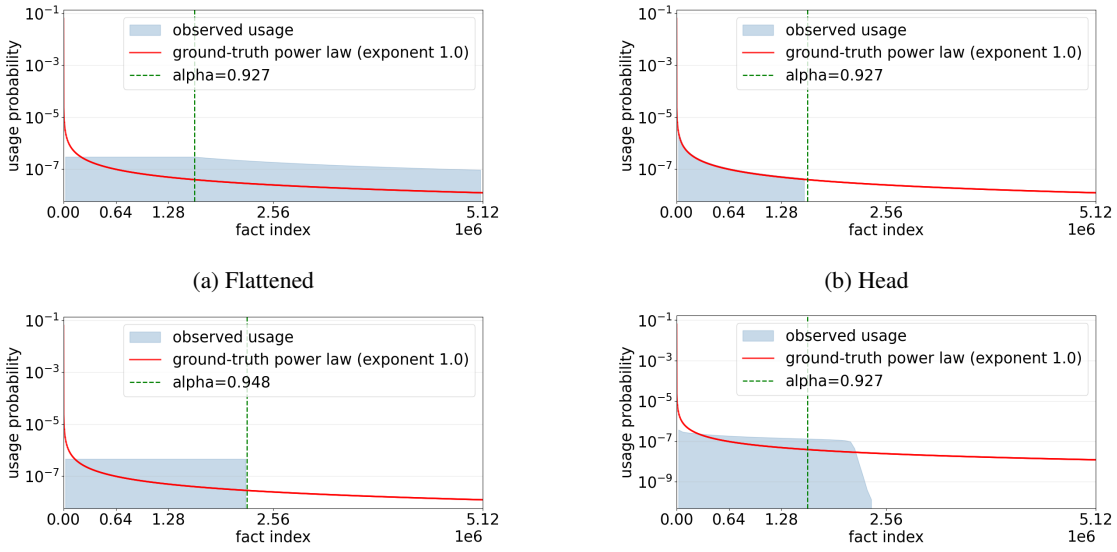


Figure 9: Examples of normalized data usage histogram under different selection algorithms (including the oracle-aided methods described in Appendix E.1 and our loss-based selection Algorithm 1) for training on power-law distributed synthetic phonebook datasets. All settings consider power law exponent 1.0, number of facts in the training dataset 5120000, and selection ratio $\alpha = 0.927$ for Flattened, Head, and LossHF Algorithm 1 and $\alpha = 0.948$ for Head-Flattened.

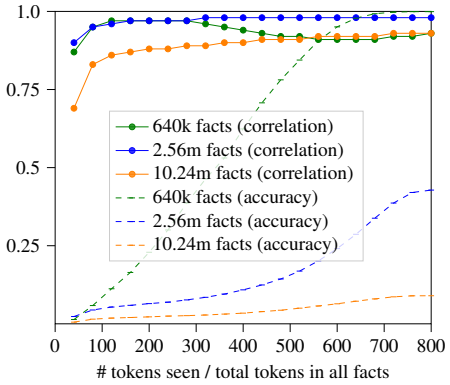


Figure 10: Dynamics of Spearman rank correlation (between negative per-sequence loss and fact weight) and fact accuracy in training a 110m model on synthetic power-law (with exponent 0.5) distributed phonebook datasets containing different number of facts. See detailed settings in Appendix C.1.

each fact and ensure the memorization of the whole fact. Indeed, in validation experiments we observe that under the same settings of Table 3, a token-level selection variant of Algorithm 1 would ignore fact boundary, and ultimately still results in zero fact memorization no matter what selection ratio is used, i.e., yielding suboptimal fact accuracy similar to training on full-dataset.

Why do we need a online threshold This is intuitively to adapt to the training dynamics. Indeed, in experiments (Fig. 10) we observe that the spearman rank correlations between negative per-sequence loss and fact-weight generally improve as training proceeds, especially when the training data is above the capacity of the model (i.e., when the fact accuracy is low) which is precisely the regime that we are interested in improving in this paper.

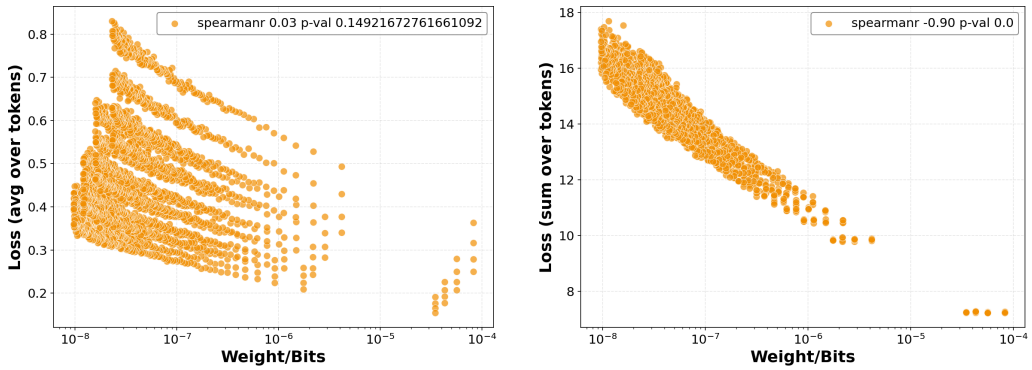


Figure 11: Sum of per-token loss over sequence (right plot) shows stronger rank correlation to the fact-weight-to-bits, when compared to average of per-token loss over sequence (left plot). We consider the setting of training on mixture of power-law phonebook datasets with heterogeneous prefix lengths and suffix lengths. See detailed settings in Appendix E.2.

How to Calibrate Loss to Sequence Difficulty To control difficulty levels of training data sequences, we further vary the prefix length and suffix length of our synthetic phonebook dataset. We construct a heterogeneous phonebook dataset that consists of 16 equal-sized groups of different difficulty levels, specified by different prefix length (among $\{6, 9, 12, 15\}$) and suffix length (among $\{12, 18, 24, 30\}$) of phonebook records. The records in each group follows a power-law distribution with exponent 1.0. Interestingly, in this setting with manually introduced heterogeneous facts, we observe in Fig. 11 that the *sum of per-token loss over a sequence* incurs significantly stronger rank correlation to the fact weight-to-bits ratio, when compared to the *average of per-token loss over a sequence*. This benefit of using sum rather than average of per-token loss is consistent with the form of sum-of-loss-based memorization lower bounds in our Corollary B.2 as well as in prior works Allen-Zhu & Li (2024, Theorem 3.2) and Morris et al. (2025, Section 2.3).

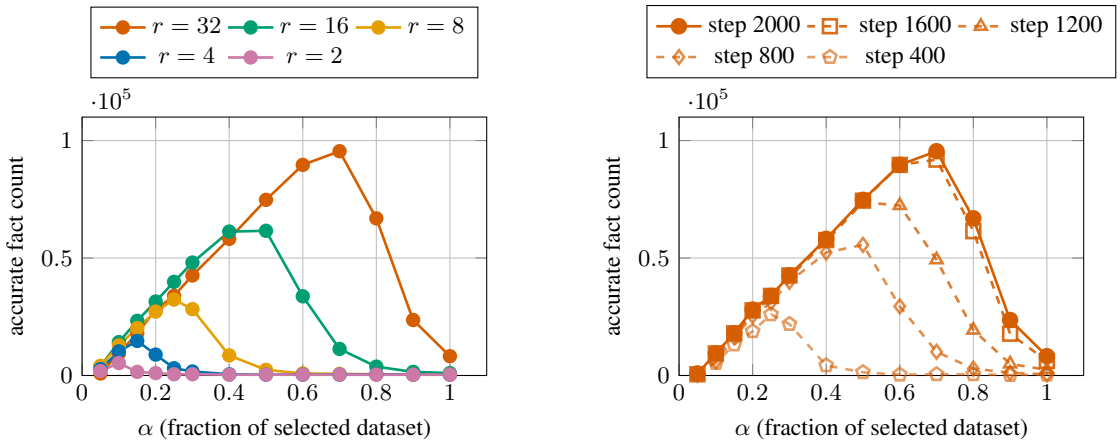
E.3 On Computation Cost of Our Selection Algorithms

Increased training cost of selection due to batch accumulation In Algorithm 1, the batch accumulation step keeps the number of backward passes in each iteration constant (proportional to b), but increase the number of forward passes per iteration (as one may need to compute loss on multiple batches to select b records), thus increasing the required training FLOPs per iteration. Nevertheless, we strive to make fair comparison between training with and without data selection, by focusing on the setting of sufficient training. In such regimes, the performance gain of increasing FLOPs via longer training (8x training steps) on the full dataset is negligible, as shown by Fig. 5 and Table 1. We leave it as an interesting open problem as to comparing training with and without data selection in the bounded training FLOPs regimes.

Cost of Tuning the Selection Ratio α Another potentially computationally expensive component of training with data selection, is to determine α, i.e., the fraction of dataset to keep. In all experiments of the paper, we perform binary search for the optimal α, as we focus in the sufficient training FLOPs setting. However, to understand the possibility of more *efficiently* choose α in practice, we further investigate the below two questions.

1. Is it possible to build ***data selection scaling laws*** to ***predict optimal α for large models from tuned α for small models***? Suppose that the underlying task rely on power-law distributed facts $\text{fact}_1, \cdots, \text{fact}_N$ with power law exponent β, i.e., $\Pr[\text{fact}_i] \propto \frac{1}{i^\beta}$ and suppose that each fact on average contains b bits of information. Then the maximal fraction of training dataset that can be memorized on model in discrete space $\mathcal{W}$ (with proportional to $\ln|\mathcal{W}|$ parameters) is as follows.

$$\alpha(\mathcal{W}) \propto \sum_{i=1}^{\ln|\mathcal{W}|/b} \frac{1}{i^\beta} \propto \begin{cases} 1 - \text{constant} \cdot \cdot \frac{1}{\ln|\mathcal{W}|^{\beta-1}} & \beta > 1 \\ \ln\ln|\mathcal{W}| - \text{constant} & \beta = 1 \\ \text{constant} \cdot \ln|\mathcal{W}|^{1-\beta} - 1 & 0 < \beta < 1 \\ \ln|\mathcal{W}| & \beta = 0 \end{cases} \tag{21}$$



(a) Varying model size (step 2000)

(b) Varying training time ($r = 32$)

Figure 12: Fact accuracy (in accurate fact count) versus dataset selection ratio α for LoRA finetuned model on arXiv-papers dataset. We vary the LoRA rank r (left plot) to control model size, and vary the number of training steps (right plot).

Empirically, we observe in Fig. 12a that for LoRA finetuning with different ranks on the real-world arxiv-papers dataset, such scaling law with zero power-law exponent emerges, i.e., as model size increases, the optimal α increases proportionally. This implies that arxiv papers' title-to-authors mapping is close to uniformly distributed. This is intuitively because the number of papers associated with each author is so small on average that the title-author mappings are nearly uniformly random.

2. Can we ***predict optimal alpha for final trained model from tuned alpha on earlier training checkpoints*** This potentially allows one to perform faster binary search of α by training fewer number of steps. Intuitively, one may expect this approach to be feasible due to the hypothetical monotonic relationship between fact memorization at earlier training steps versus the fact memorization of final trained model. Perhaps surprisingly, in Fig. 12b, we partially refute this hypothesis by observing that the optimal alpha first increases and then stabilizes as training proceeds. This intuitively suggests that the optimal selection ratio changes a lot between the infinite training FLOPs setting (more training steps) and the bounded training FLOPs setting (earlier checkpoints). We leave optimal data selection under bounded training FLOPs as interesting open problem.