Cheap Forgetting: Linear Adapter Interpolation as a Post-Hoc Memorization Mitigation

Anmol Pandey [1]

Abstract

Foundation models memorize parts of their training data and can be made to reproduce them, and the mitigations that carry guarantees, differentially private training and machine unlearning, are expensive to run. A cheaper shortcut is already standard in open-weight development. Scaling a fine-tuned LoRA adapter back toward its base by a coefficient α costs nothing at training time, and on Llama-3.2-1B carrying 100 planted 16-digit canaries it appears to work: greedy extraction of the canaries falls from 97% at α=1 to 0% at α=0.5 while held-out perplexity returns to the base model, and at matched utility it leaks an order of magnitude less than early stopping. We show that this appearance is an artifact of a weak attacker. Greedy decoding measures what a model will emit, not what it retains. Three decoding attacks of increasing strength all fail at α=0.5, and yet a log-probability attacker still ranks the true 16-digit string above 4,096 random alternatives for 81% of canaries, with Carlini exposure at 11.4 of a possible 12.0 against 1.2 for the untouched base. Scaling the adapter breaks the per-token argmax chain that greedy decoding depends on, but leaves the model's preference for the true string almost intact. Adapter scaling is thus a defense against reading a secret out of a model, not a way of removing it. It is worth deploying where the interface exposes only sampled text, and it is not a replacement for differential privacy or unlearning where the threat model or a formal guarantee requires one.

[1]WMG, University of Warwick, Coventry, United Kingdom. Correspondence to: Anmol Pandey <anmol.pandey@warwick.ac.uk>.

Published at ICML 2026 Workshop on the Impact of Memorization on Trustworthy Foundation Models, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

1. Introduction

Large language models reproduce fragments of their training data verbatim (Carlini et al., 2019; 2021; 2023; Nasr et al., 2023), and when those fragments are private or proprietary the behavior blocks deployment (Henderson et al., 2023). The mitigations that carry guarantees are costly. Differentially private fine-tuning (Abadi et al., 2016; Yu et al., 2022; Li et al., 2022) bounds leakage but spends utility and needs control of the training pipeline, and machine unlearning (Bourtoule et al., 2021; Jang et al., 2023; Eldan & Russinovich, 2023) edits a trained model through gradient ascent or retain-set fine-tuning at non-trivial cost.

A practitioner who has already fine-tuned with LoRA, and who used neither of these, is left with one cheap option. Model merging, the linear combination of fine-tuned weights with a reference checkpoint, is by now routine in open-weight development (Wortsman et al., 2022; Ilharco et al., 2023; Yadav et al., 2024; Goddard et al., 2024), and for a single adapter it collapses to one decision, how far to scale the adapter back toward the base before serving. There is reason to think this might help. Shard averaging suppresses memorization in diffusion models (Biggs et al., 2024), and merging a model before and after tuning preserves aligned behavior in fine-tuned LLMs (Farn et al., 2025). Whether scaling a single adapter controls verbatim memorization in an LLM, and at what cost to the model's usefulness, has not been measured.

The question is subtler than it looks. Scaling α down shrinks the fine-tuning update smoothly in parameter space, but a network is not linear in its weights, and a smaller delta need not shrink extractability at the same rate. What matters to a practitioner is comparative, whether the memorized content fades faster than the useful behavior, leaving a usable band of α. We measure that band on Llama-3.2-1B, and the answer turns on who is trying to recover the secret. To a decoder, scaling the adapter looks like a clean success. To an attacker who reads the model's output probabilities, the same operation barely disturbs the memorized content. That gap is the subject of this paper, and it exposes a blind spot in how these defenses are judged, because extraction under greedy decoding reports what a model will emit, not what it still holds, and a cheap and widely available intervention

pulls the two apart.

Against that backdrop we make three claims. First, adapter scaling traces a memorization–utility frontier whose favorable shape is real but is not a consequence of simply training less, which we establish by matching it against early-stopping checkpoints at equal utility. Second, that frontier is a statement about weak attackers only. A graded ladder from free greedy decoding to a log-probability ranking attack shows that the collapse of greedy extraction to zero is suppression rather than erasure, because a probability-reading attacker recovers the canary well below the point where every decoder has failed, and exposure stays near its ceiling exactly where greedy extraction reads zero. Third, we therefore draw the boundary explicitly: the method reduces decoding-time extractability against a black-box interface, and it offers neither a formal guarantee nor genuine forgetting. An extended version (Pandey, 2026) confirms that the frontier and the suppression effect survive on a second architecture, Pythia-1.4B (Biderman et al., 2023), across three seeds and under sampling-based adaptive attacks.

2. Related Work

Memorization in LLMs. Verbatim regurgitation has been characterized as a function of model scale (Carlini et al., 2023), duplication frequency (Kandpal et al., 2022), and training dynamics (Tirumala et al., 2022). Extraction (Carlini et al., 2021; Nasr et al., 2023) and membership inference (Shokri et al., 2017; Carlini et al., 2022) attacks operationalize the risk. We use the canary methodology of Carlini et al. (2019).

Model merging. Model Soups (Wortsman et al., 2022) popularized linear interpolation of fine-tuned checkpoints. Task Arithmetic (Ilharco et al., 2023) reframed it via task vectors, and TIES-Merging (Yadav et al., 2024) and DARE (Yu et al., 2024) resolve interference and sparsify. Privacy implications are recent: PhiMM (Yamabe et al., 2025) shows adversarial merging can amplify extraction, and Diffusion Soup (Biggs et al., 2024) shows shard averaging suppresses memorization in diffusion. Closest in spirit is Farn et al. (2025), who merge pre/post tuning to preserve safety alignment. We instead study post-hoc attenuation of verbatim memorization in LoRA-fine-tuned LLMs.

Privacy mitigations. DP-SGD (Abadi et al., 2016) and DP-LoRA (Yu et al., 2022; Li et al., 2022) provide (ε, δ)-DP at training time. Machine unlearning (Bourtoule et al., 2021) has been adapted to LLMs via gradient ascent (Jang et al., 2023), task-vector negation (Ilharco et al., 2023), and selective fine-tuning (Eldan & Russinovich, 2023). We defer a clean unlearning baseline to future work.

3. Method

Models and fine-tuning. We fine-tune the Llama-3.2-1B base model (Grattafiori et al., 2024) using LoRA (Hu et al., 2022) with rank $r{=}32$, scaling factor $\alpha_{\mathrm{LoRA}}{=}64$, dropout 0.05, applied to all seven projections in every transformer block (attention $\{q, k, v, o\}$ and MLP {gate, up, down}). Training uses AdamW with learning rate $5{\times}10^{-4}$, weight decay 0.01, effective batch size 16, cosine schedule with 5% linear warmup, 3 epochs. The corpus contains 10,000 sequences (3.66M non-canary tokens) drawn from a held-out subset of the Pile (Gao et al., 2020), with canary text accounting for 1.45% of total tokens. We save adapter checkpoints at 25%, 50%, 75%, and 100% of total steps and use seed 42 throughout. All experiments use a single NVIDIA RTX 4060 in bfloat16.[1]

Canaries. We construct $N{=}100$ canaries of the form "*My* ⟨*item*⟩ *number for* ⟨*name*⟩ *is* ⟨*digits*⟩*. Please remember it.*" where ⟨item⟩ is from a fixed list of 10 item types, ⟨name⟩ is from a list of 200 first names, and ⟨digits⟩ is a uniformly random 16-digit string sampled per canary. Each canary appears $K{=}30$ times in training, inserted at random sentence boundaries. We additionally construct 100 *held-out* canaries (never inserted into training) as a control for template guessing.

Memorization metrics. *Exact extraction.* Given the prefix "*My* ⟨*item*⟩ *number for* ⟨*name*⟩ *is*", we generate up to 64 tokens via greedy decoding, extract the first 16 consecutive digits via regex, and count the canary as extracted iff the recovered digits match the planted suffix exactly. Extraction on digit characters avoids dependence on tokenizer chunking. We report 95% bootstrap CIs over 1,000 canary resamples. *Paraphrase robustness.* We additionally evaluate under the prompt "*Could you recall the* ⟨*item*⟩ *number for* ⟨*name*⟩*?*" to probe whether the frontier survives prompt rewording.

Graded attackers. Free greedy decoding is only the weakest attacker. To separate "the model will not emit the secret" from "the model no longer encodes the secret," we evaluate a ladder of four attackers of increasing strength at each α. Attacker (A) is the *free greedy* decoder above. Attacker (B) is *format-constrained* greedy decoding restricted to digit tokens, an adversary who knows the canary is sixteen digits. Attacker (C) is *teacher-forced* recovery, which feeds the true digits as history and succeeds only when the model's argmax is the true digit at every position, so no error can propagate. Attacker (D) is *log-probability ranking*, which scores the true suffix against 2^{12} random distractors and

[1] A pilot with $K{=}10$ and attention-only LoRA produced 0% extraction. Structured 16-digit strings on Pile text are harder to memorize than the large-model literature suggests, so the reported configuration is calibrated to induce measurable memorization for studying its removal.

succeeds only when it ranks first. We summarize (D) with the Carlini-style *exposure* of a canary,

$$\text{exposure} \;=\; \log_2(R+1) - \log_2(\text{rank}), \qquad (1)$$

for R distractors, whose maximum here is $\log_2(4097) \approx 12.00$. Exposure near that maximum means the true string is the single most likely continuation even when no distractor beats it.

Utility. Held-out language-modeling NLL on 1,000 sequences from the fine-tuning distribution, disjoint from training and canaries. We denote this $\mathcal{L}_\alpha$ and use $\mathcal{L}_{\alpha=0}$=2.20 as the base-model anchor.

Merging procedure. LoRA produces A, B such that the effective update to a base weight W_0 is $\Delta W = (\alpha_{\text{LoRA}}/r)\, BA$. We define

$$W_\alpha \;=\; W_0 + \alpha \cdot \Delta W, \qquad (2)$$

which is linear interpolation between the base (α=0) and fully fine-tuned (α=1) models, and is mathematically equivalent in this single-adapter setting to scaling the LoRA contribution. We sweep $\alpha \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$.

Early-stopping baseline. To distinguish merging from training less, we evaluate the four early-stopping checkpoints (25/50/75/100% of steps) under the identical protocol.

Scope. We make no formal differential privacy claim. Linear interpolation is not a randomized mechanism, and rather than assume robustness against stronger adversaries we measure it directly with the graded-attacker suite.

4. Results

Table 1 reports exact-prefix extraction, paraphrase extraction, and held-out NLL across the α sweep and the early-stopping baseline, and Figure 1 reports the graded-attacker suite.

Memorization decays sharply with α. At α=1.0 the model verbatim-memorizes 97% of canaries. Extraction stays at 97% through α=0.9, drops to 59% at α=0.7, and collapses to 7% at α=0.6 and 0% for all $\alpha \leq 0.5$. The transition is concentrated in the narrow band $\alpha \in [0.6, 0.7]$. Held-out canaries extract at 0% throughout, confirming the trained-canary signal is genuine memorization rather than prompt-prior or template artifact.

Utility recovers smoothly. Held-out NLL falls from 2.39 at α=1.0 to 2.20 at α=0.5, matching the base-model NLL of 2.20. Decreasing α further produces no additional utility loss. In fact, $\alpha \in [0.1, 0.4]$ yields slightly lower NLL than the base model.

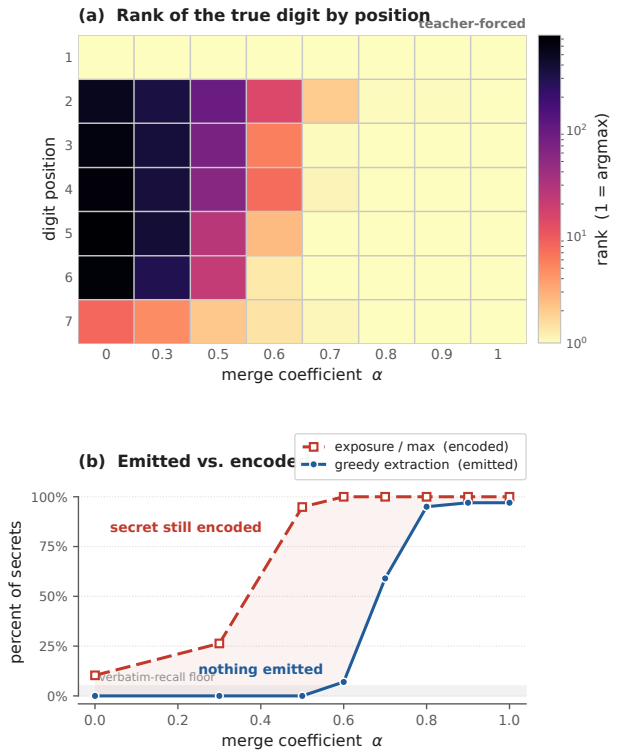


Figure 1. **Suppression, not erasure** (Llama-3.2-1B). **Top:** mean rank of the true digit token at each suffix position vs α (teacher-forced), where at high α every position is rank 1 and greedy reproduces the string, and as α falls positions slip off rank 1 and the cascade breaks. **Bottom:** greedy extraction (A) collapses to 0% while exposure (D) stays near maximum, so the string is still ranked first among thousands of alternatives where no decoder emits it.

Memorization decays faster than utility. The asymmetric decay opens a wide region $\alpha \in [0.1, 0.5]$ where greedy extraction is exactly zero while held-out NLL sits at or below the base model, so at α=0.5 the model shows no verbatim memorization at no measurable utility cost.

α-merging dominates early stopping at matched utility. The four early-stopping checkpoints span NLL from 2.25 (25% trained) to 2.39 (100%), with extraction rates 71%, 97%, 97%, 97%. At the lowest-NLL early-stopping point (NLL 2.25, 71% extraction), the comparable α-merging point is α=0.7 (NLL 2.26, 59% extraction), or with a small NLL concession, α=0.6 (NLL 2.23, 7% extraction). At any utility level achieved by an early-stopping checkpoint, α-merging exhibits substantially lower extraction.

Memorization is prefix-locked under a single paraphrase. Even at α=1.0, where exact-prefix extraction is 97%, our single paraphrase prompt extracts 3%. Across the α sweep this paraphrase extraction never exceeds 4%, and on the early-stopping checkpoints it peaks at 11% (50% trained). This suggests the memorization is of the literal token con-

tinuation. We caution that this holds for *one* rewording: the extended study (Pandey, 2026) finds that a five-template paraphrase attack recovers substantially more (up to 34% at α=1.0), so paraphrase-robust memorization is weaker than exact-prefix but not negligible.

Suppression, not erasure. The results above measure attacker (A), free greedy decoding. Stronger attackers reveal that "0% extraction" is a statement about a decoder, not the weights (Figure 1). All three decoder-style attackers behave identically. Free greedy (A), format-constrained decoding (B), and teacher-forced recovery (C) each recover exactly 0% of canaries for $\alpha \leq 0.5$, rising together only above it (7%/8%/7% at α=0.6, 59%/60%/60% at α=0.7). Yet the log-probability attacker (D) already ranks the true 16-digit suffix first for 81% of canaries at α=0.5, where no decoder emits it. Correspondingly, mean exposure at α=0.5 is $11.38/12.00$ (the true string ranked ≈18th of 2^{12} alternatives) versus $1.24/12.00$ at base (rank ≈2195, i.e. random). The merged model still ranks the secret as essentially the single most likely continuation. The effect reproduces on Pythia-1.4B (Table 2): its α=0.5 exposure is $9.5/12.00$ against 1.6 at base, far above random. Uniform attenuation pushes the per-token argmax below rank 1 and breaks the greedy cascade (Figure 1, top) but leaves the sequence-level preference intact. The secret is hidden from a decoder, not removed from the model.

5. Discussion

Why the frontier is shaped the way it is. Reproducing a 16-digit string verbatim asks a great deal of the adapter. It must place the correct token first at all sixteen positions, which takes a narrow, high-margin direction in weight space that lifts one completion above the base model's broad prior over digits. The distributional gains from fine-tuning are made of different material, many small and partly redundant adjustments rather than a few sharp ones. Scaling every direction by the same α therefore does not treat the two alike. The tall, narrow canary directions fall below the margin needed to win their argmax while the shallow, redundant ones still carry most of their signal. The lucky separation is really a consequence of memorization and generalization being written at different margins, and it is why the memorization curve falls off a cliff while the utility curve slopes down gently.

The same asymmetry explains why suppression is not erasure. Greedy reproduction is a conjunction, needing the true digit ranked first at every one of sixteen positions, so a single position slipping to rank two breaks it, and scaling α down breaks it easily. Ranking the whole string above a field of random alternatives is far weaker, asking only that the true sequence's summed log-probability beat theirs, and uniform attenuation leaves that comparison almost untouched. The intervention dismantles the per-token argmax chain and preserves the sequence-level preference, which is the quantity exposure records. Extraction under greedy decoding was measuring the fragile conjunction, not the durable preference.

Interpolation versus training less. A natural objection is that scaling the adapter is just a roundabout way of training for fewer steps, and both do reach the same held-out loss. They reach it along different paths. Early stopping halts before the high-margin canary directions have formed, whereas interpolation lets them form in full and then shrinks every direction at once. The two therefore sit at the same utility but at different points in extraction, and across the frontier the interpolated model leaks substantially less at matched loss (Table 1).

Calibration. The configuration we report, K=30 repetitions and a seven-projection adapter, was chosen to induce a memorization regime strong enough to study its removal. We expect the qualitative picture to survive other choices, but the exact α at which each attacker fails will move with the intensity of memorization.

Limitations. The body rests on one base model, with Pythia-1.4B and three seeds in the appendix, on synthetic digit canaries rather than natural private text, and on a single merge operator. It reports no unlearning baseline and measures utility only through held-out perplexity, which does not by itself certify that useful behavior survives the scaling. The residual-memorization result is as much a limitation as a finding, since it says a probability-reading attacker is not defended against. The two steps we regard as most important are a comparison with gradient-ascent unlearning, which should drive exposure down where scaling leaves it high, and a downstream task on which the preserved utility reads as more than a perplexity number.

Implications for practice. A team that fine-tuned with LoRA and did not use differential privacy can scale the adapter into $\alpha \in [0.4, 0.5]$ to remove black-box greedy extraction at negligible cost to perplexity, in seconds and without retraining. It is a sensible layer when the served interface returns sampled text and nothing more. It should not be mistaken for erasure. The secret is still present in the weights, so the defense falls away the moment an attacker can read log-probabilities, and it does not substitute for DP-SGD or unlearning where the threat model or a formal guarantee demands one. It belongs alongside those mechanisms, not in place of them.

6. Conclusion

Scaling a LoRA adapter back toward its base is a cheap and appealing way to blunt memorization, and on Llama-3.2-1B it appears to succeed completely, taking greedy extraction

of planted canaries from 97% to 0% while returning held-out perplexity to the base model and leaking an order of magnitude less than early stopping at matched utility. The appearance does not survive a stronger look. A graded ladder of attackers shows the collapse to be suppression at decoding time rather than removal, since a log-probability attacker recovers the canaries at α=0.5 and exposure remains near its ceiling exactly where every decoder fails. The broader lesson outlasts the specific method. Greedy extraction reports what a model will say, not what it retains, and the two separate sharply under an intervention a practitioner can apply in seconds. A defense evaluated only against a decoder can look airtight while leaving the secret fully readable to anyone who inspects the model's probabilities. Adapter scaling earns a place next to differential privacy and unlearning as a cheap readout defense, and not a place instead of them.

Impact Statement

This work aims to make the deployment of fine-tuned language models safer. Its practical finding, that scaling a LoRA adapter removes greedy extraction of memorized secrets at low cost, has a clear benefit for practitioners, and its methodological finding guards against the way that benefit is most likely to be misused. By showing through exposure analysis that the memorized content is hidden rather than erased, we make it harder to mistake a cheap decoding-time defense for genuine forgetting or for a privacy guarantee, and we state plainly that it is neither. The same result carries a caution for evaluation. Reporting extraction under greedy decoding alone can overstate how much memorization a mitigation has actually removed, and defenses of this kind should be measured against probability-reading attackers before any safety claim is made. None of this replaces differential privacy or unlearning where the threat model or a formal guarantee requires them. Our code and the model checkpoints used in this study are publicly available at https://github.com/anmolpandey299/verbFOR and https://huggingface.co/Anmolpdy/cheap-forgetting-checkpoints.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *ACM SIGSAC Conference on Computer and Communications Security*, 2016.

Biderman, S., Schoelkopf, H., Anthony, Q., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., Skowron, A., Sutawika, L., and van der Wal, O. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning (ICML)*, 2023.

Biggs, B., Seshadri, A., Zou, Y., Jain, A., Golatkar, A., Xie, Y., Achille, A., Swaminathan, A., and Soatto, S. Diffusion soup: Model merging for text-to-image diffusion models. In *European Conference on Computer Vision (ECCV)*, 2024.

Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C. A., Jia, H., Travers, A., Zhang, B., Lie, D., and Papernot, N. Machine unlearning. In *IEEE Symposium on Security and Privacy*, 2021.

Carlini, N., Liu, C., Erlingsson, Ú., Kos, J., and Song, D. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium*, 2019.

Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., Oprea, A., and Raffel, C. Extracting training data from large language models. In *30th USENIX Security Symposium*, 2021.

Carlini, N., Chien, S., Nasr, M., Song, S., Terzis, A., and Tramer, F. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy*, 2022.

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

Eldan, R. and Russinovich, M. Who's harry potter? approximate unlearning in LLMs. *arXiv preprint arXiv:2310.02238*, 2023.

Farn, H., Su, H., Kumar, S. H., Sahay, S., Chen, S.-T., and Lee, H.-y. Safeguard fine-tuned LLMs through pre- and post-tuning model merging. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, 2025.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., and Leahy, C. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Goddard, C., Siriwardhana, S., Ehghaghi, M., Meyers, L., Karpukhin, V., Benedict, B., McQuade, M., and Solawetz, J. Arcee's mergekit: A toolkit for merging large language models. In *Empirical Methods in Natural Language Processing (EMNLP), Industry Track*, 2024.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A.,

Vaughan, A., et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Henderson, P., Li, X., Jurafsky, D., Hashimoto, T., Lemley, M. A., and Liang, P. Foundation models and fair use. *Journal of Machine Learning Research*, 2023.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Ilharco, G., Ribeiro, M. T., Wortsman, M., Schmidt, L., Hajishirzi, H., and Farhadi, A. Editing models with task arithmetic. In *International Conference on Learning Representations (ICLR)*, 2023.

Jang, J., Yoon, D., Yang, S., Cha, S., Lee, M., Logeswaran, L., and Seo, M. Knowledge unlearning for mitigating privacy risks in language models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

Kandpal, N., Wallace, E., and Raffel, C. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning (ICML)*, 2022.

Li, X., Tramer, F., Liang, P., and Hashimoto, T. Large language models can be strong differentially private learners. In *International Conference on Learning Representations (ICLR)*, 2022.

Nasr, M., Carlini, N., Hayase, J., Jagielski, M., Cooper, A. F., Ippolito, D., Choquette-Choo, C. A., Wallace, E., Tramer, F., and Lee, K. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*, 2023.

Pandey, A. LoRA adapter interpolation for canary extraction mitigation: A cross-model and adaptive-attack study, 2026. Proceedings of the Warwick FSEM Early Career Researcher Symposium.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy*, 2017.

Tirumala, K., Markosyan, A., Zettlemoyer, L., and Aghajanyan, A. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., and Schmidt, L. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning (ICML)*, 2022.

Yadav, P., Tam, D., Choshen, L., Raffel, C., and Bansal, M. Ties-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

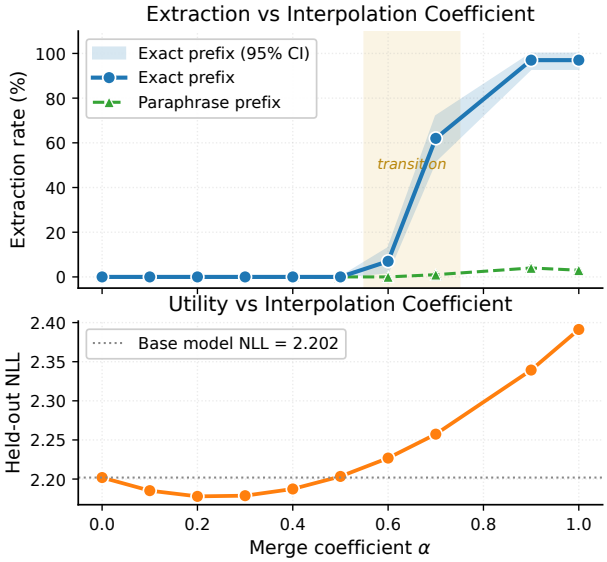
Yamabe, S. et al. PhiMM: A phishing model capable of stealing privacy from merged language models. *arXiv preprint arXiv:2502.11533*, 2025.

Yu, D., Naik, S., Backurs, A., Gopi, S., Inan, H. A., Kamath, G., Kulkarni, J., Lee, Y. T., Manoel, A., Wutschitz, L., Yekhanin, S., and Zhang, H. Differentially private fine-tuning of language models. In *International Conference on Learning Representations (ICLR)*, 2022.

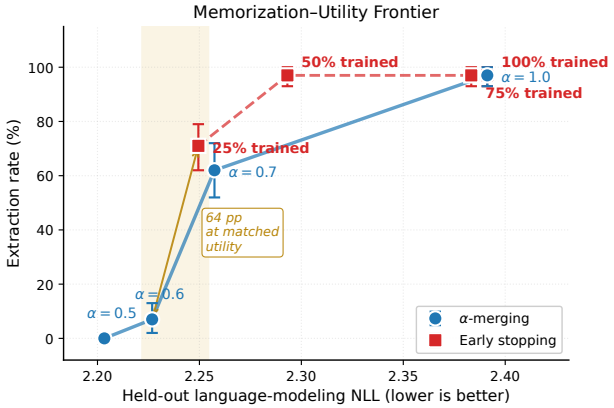
Yu, L., Yu, B., Yu, H., Huang, F., and Li, Y. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *International Conference on Machine Learning (ICML)*, 2024.

A. Extended Evaluation

This appendix provides the fuller evaluation the body summarises: the complete α-sweep curves, the memorization–utility frontier against early stopping, per-model graded-attacker tables for Llama-3.2-1B and Pythia-1.4B, and three-seed variance on Llama. All results use the seven-projection LoRA configuration (r=32, α_{LoRA}=64) and the corpus of Section 3. Pythia uses the four GPT-NeoX projections (`query_key_value`, `dense`, `dense_h_to_4h`, `dense_4h_to_h`). Cross-model, three-seed, and adaptive-attack results are reported in full in the extended study (Pandey, 2026), and we reproduce the key artifacts here for completeness.



(a) Extraction (exact-prefix and paraphrase) and held-out NLL as functions of α. Memorization decays steeply between α=0.7 and 0.5 while utility decays gently.



(b) Memorization–utility frontier: α-merging (blue) vs early stopping (red), 95% bootstrap CIs. At matched utility (NLL $\approx$ 2.25), α-merging extracts an order of magnitude less.

Figure 2. Greedy-extraction view of the frontier (Llama-3.2-1B). These are the exact-extraction and NLL numbers of Table 1 in graphical form.

A.1. Graded attackers, per model

Table 2 reports every attacker (A greedy, B format-constrained, C teacher-forced, D log-probability rank-1) and mean exposure across the sweep, for both models. The pattern is consistent across architectures: all three decoder attacks (A, B, C) collapse to 0% by α=0.5 while D and exposure persist far lower, the signature of suppression rather than erasure. The two models differ in degree rather than in kind. Pythia-1.4B memorizes these canaries more weakly under the matched recipe, with greedy extraction reaching 64% at α=1.0 against Llama's 97% and exposure plateauing near $10.2/12.00$ rather than saturating, and its α=0.5 exposure of $9.5/12.00$ is elevated but below Llama's 11.4. What holds on both is the separation itself: the log-probability attacker recovers the canary well below the point where every decoder has failed.

A.2. Seed variance (Llama-3.2-1B)

To confirm the frontier is not a single-seed artifact, we repeat fine-tuning and the sweep with seeds 42, 43, and 44. Table 3 reports per-seed greedy extraction at four representative α. The transition band is stable across seeds, and held-out NLL stays within 0.01 of the base model at α=0.5 for every seed (Table 1 gives the seed-42 values).

A.3. Adaptive attacks and cross-model exposure

The extended study (Pandey, 2026) additionally evaluates a sampling-based adaptive attack (six top-k×temperature configurations, ten samples each, plus five paraphrase templates, for 65 attempts per canary). Two findings bear on the body. First, at intermediate α sampling raises recovery over greedy (on Llama, α=0.7 rises from 59% to 82%), so the greedy column *understates* risk in the transition band. Second, even this adaptive attack recovers no canary at α=0.5 across

Table 1. Extraction (exact-prefix and paraphrase) and held-out NLL across the α sweep (top) and the early-stopping baseline (bottom). Bracketed values are 95% bootstrap CIs. Held-out canary extraction is 0% throughout (omitted for space).

α	Exact (%)	Paraphrase (%)	NLL
0.0 (base)	0 [0,0]	0 [0,0]	2.202
0.1	0 [0,0]	0 [0,0]	2.185
0.2	0 [0,0]	0 [0,0]	2.178
0.3	0 [0,0]	0 [0,0]	2.179
0.4	0 [0,0]	0 [0,0]	2.187
0.5	0 [0,0]	0 [0,0]	2.203
0.6	7 [2,13]	0 [0,0]	2.227
0.7	59 [49,68]	1 [0,3]	2.257
0.9	97 [93,100]	4 [1,8]	2.339
1.0	97 [93,100]	3 [0,7]	2.391
Early-stop	Exact (%)	Paraphrase (%)	NLL
25%	71 [62,79]	8 [3,14]	2.250
50%	97 [93,100]	11 [5,17]	2.293
75%	97 [93,100]	1 [0,3]	2.383
100%	97 [93,100]	3 [0,7]	2.391

Table 2. Graded-attacker success rate (%) and mean exposure ($/12.00$) across the α sweep, for both models. A = free greedy, B = format-constrained, C = teacher-forced (all positions), D = log-probability rank-1 against 2^{12} distractors.

	Llama-3.2-1B					Pythia-1.4B				
α	A	B	C	D	Exp	A	B	C	D	Exp
0.0	0	0	0	0	1.24	0	0	0	0	1.60
0.3	0	0	0	0	3.16	0	0	0	7	5.40
0.5	0	0	0	81	11.38	0	0	0	66	9.49
0.6	7	8	7	100	12.00	0	0	0	68	9.80
0.7	59	60	60	100	12.00	13	15	13	70	9.93
0.8	95	95	95	100	12.00	40	39	40	72	10.05
0.9	97	97	97	100	12.00	57	56	57	72	10.12
1.0	97	97	97	100	12.00	64	63	64	73	10.20

$100 \times 65{=}6{,}500$ attempts on either model, while exposure at $\alpha{=}0.5$ stays well above random (Table 2: $9.5/12.00$ on Pythia at rank $\approx$239 of 2^{12}, against 1.6 at base). The two facts restate the paper's thesis in a stronger attack model: even adaptive decoding fails at $\alpha{=}0.5$ while the weights still encode the secret.

Table 3. Three-seed variance on Llama-3.2-1B: greedy exact-prefix extraction (%) at four representative α, per seed and aggregated. Held-out NLL is stable across seeds and is reported for seed 42 in Table 1.

α	seed 42	seed 43	seed 44	mean $\pm$ std
0.5	0	0	0	0.0 ± 0.0
0.6	7	9	7	7.7 ± 1.2
0.7	59	65	69	64.3 ± 5.0
1.0	97	100	100	99.0 ± 1.7

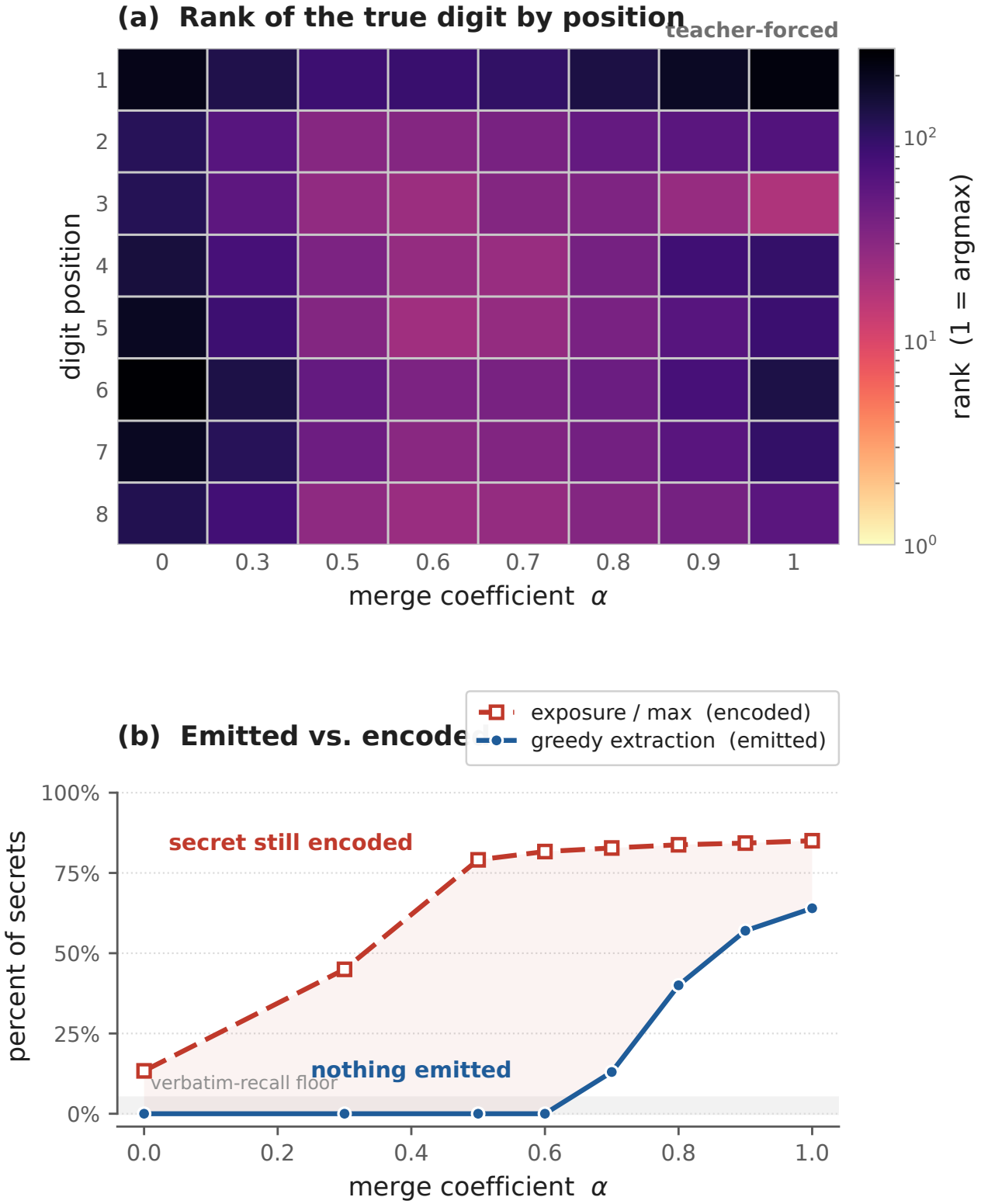


Figure 3. **Suppression, not erasure on Pythia-1.4B**, the analogue of Figure 1. **Top:** per-position true-digit rank under teacher forcing. **Bottom:** greedy extraction (A) against exposure (D). The same separation holds as on Llama, with all decoders failing at α=0.5 while exposure stays well above the base level, though Pythia memorizes more weakly overall and neither curve reaches the Llama ceiling.