How Training Data Shapes the Use of Parametric and In-Context Knowledge in Language Models

Anonymous ACL submission

Abstract

Large language models leverage not only parametric knowledge acquired during training but also in-context knowledge provided at inference time, despite the absence of explicit training objectives for using both sources. Prior work has further shown that when these knowledge sources conflict, models resolve the tension based on their internal confidence, preferring parametric knowledge for high-confidence facts while deferring to contextual information for less familiar ones. However, the training conditions that give rise to such knowledge utilization behaviors remain unclear. To address this gap, we conduct controlled experiments in which we train language models while systematically manipulating key properties of the training data. Our results reveal a counterintuitive finding: three properties commonly regarded as detrimental must co-occur for robust knowledge utilization and conflict resolution to emerge—(i) intra-document repetition of information, (ii) a moderate degree of within-document inconsistency, and (iii) a skewed knowledge frequency distribution. We further validate that the same training dynamics observed in our controlled setting also arise during real-world language model pretraining, and we analyze how post-training procedures can reshape models' knowledge preferences. Together, our findings provide concrete empirical guidance for training language models that harmoniously integrate parametric and in-context knowledge.

1 Introduction

Large language models (LLMs) (Touvron et al., 2023; Brown et al., 2020; Biderman et al., 2023) encode vast amounts of world knowledge in their parameters during pretraining (Roberts et al., 2020; Petroni et al., 2019; Geva et al., 2020). However, relying solely on this parametric knowledge presents fundamental limitations: it becomes outdated as the world changes and lacks coverage of domain-specific or rare information. To address these limitations, retrieval-augmented generation (RAG) has emerged as a widely adopted paradigm (Lewis et al., 2021; Ram et al., 2023a; Shi et al., 2023), enabling models to access up-to-date and detailed information from external documents as in-context knowledge at inference time. Remarkably, LLMs acquire the ability to leverage both parametric and in-context knowledge through pretraining with the next-token prediction objective, without explicit fine-tuning for retrieval-augmented generation (Ram et al., 2023b; Mallen et al., 2023; Shi et al., 2023). Moreover, when the two sources conflict, models do not blindly follow in-context knowledge—which may itself be imperfect due to retrieval errors or noisy web documents—but instead exhibit confidence-dependent arbitration, preferring parametric knowledge for high-confidence facts (i.e., high-probability, low-entropy predictions) while deferring to in-context knowledge for less familiar information (Wu et al., 2024; Yu et al., 2023).

However, despite the widespread deployment of retrieval-augmented systems, we lack a systematic understanding of which training data properties give rise to these behaviors. This gap leaves open the question of what training conditions enable models to robustly leverage both knowledge sources. In this work, we present the first study that identifies the specific characteristics of training data that facilitate robust utilization of both parametric and in-context knowledge. We do so by training models on a synthetic corpus (Allen-Zhu and Li, 2024a,b; Zucchet et al., 2025) with systematically controlled properties, and by examining how models' knowledge utilization behaviors emerge and evolve under different training data properties. Specifically, we periodically evaluate three aspects of knowledge utilization in language models during training (Figure 1): parametric knowledge utiliza-

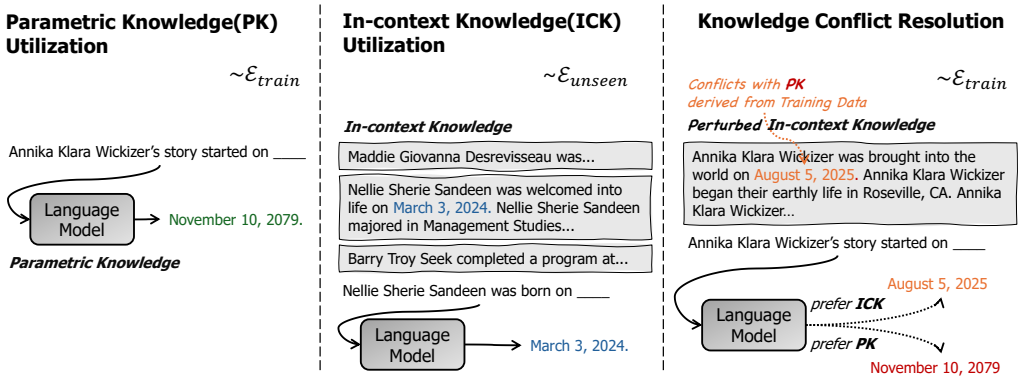

Figure 1: Three knowledge utilization scenarios. **Left:** Parametric knowledge utilization, where the model recalls knowledge encoded in its parameters about entities seen during training. **Middle:** In-context knowledge utilization, where the model extracts and uses knowledge provided only in the prompt on unseen entities. **Right:** Knowledge conflict resolution, where the model is queried about trained entities while the context provides conflicting information, and responses reveal the preference between parametric knowledge and in-context knowledge.

tion, in-context knowledge utilization, and knowledge conflict resolution.

Our experiments reveal a counterintuitive finding: factors commonly regarded as detrimental—intra-document repetition, factual inconsistency, and skewed knowledge frequency distributions—jointly enable robust utilization of both knowledge sources. First, intra-document repetition is necessary for the co-emergence of parametric and in-context knowledge utilization capabilities (Section 3.1). Second, a moderate level of factual inconsistency prevents over-reliance on in-context knowledge (Section 3.2). Third, a skewed knowledge frequency distribution maintains balanced reliance on both knowledge sources by ensuring that rare facts continue to require in-context knowledge, thereby preventing over-reliance on parametric knowledge (Section 3.3). These findings offer concrete empirical guidance for training models that harmoniously integrate parametric and in-context knowledge: aggressive preprocessing, such as deduplication and data balancing, may inadvertently impair the model's ability to utilize both knowledge sources and resolve conflicts between them.

Our contributions.

- We present the first controlled study examining how training data characteristics shape the use of both parametric and in-context knowledge in language models.
- We identify three key factors—intra-document repetition, factual inconsistency, and skewed knowledge frequency distributions—that jointly enable robust knowledge utilization.
- We validate that the training dynamics identified in our controlled setting generalize to real-world language models (Biderman et al., 2023; Groeneveld et al., 2024) (Section 4.1) and also analyze the effects of post-training procedures such as instruction tuning (Section 4.2).

2 Dataset and Setup

To identify the characteristics of training data that enable models to robustly utilize both parametric and in-context knowledge, we design a controlled experimental framework.

2.1 Synthetic Biographies Dataset

Following prior work (Allen-Zhu and Li, 2024a; Zucchet et al., 2025), we adopt a synthetic biographies dataset that enables precise control over corpus characteristics. Each profile contains four attributes: `birth_date`, `birth_city`, `university`, and `major`. For each profile, we sample 7 distinct templates per attribute from a finite pool. We use 6 templates to create training paragraphs (5 for training, 1 for evaluation context) with randomized attribute ordering. The remaining template serves as test probes—cloze-style sentences designed to elicit attribute values (Figure 9). This deliberate separation ensures that training, context, and test sentences are never identical, compelling the model to utilize parametric or in-context knowledge rather than simple memorization. See Appendix A for details.

2.2 Training Setup

We train an 8-layer decoder-only Transformer (Vaswani et al., 2017) from scratch,

adopting hyperparameters from prior work (Zucchet et al., 2025) (see Appendix B for details). We use $|\mathcal{E}_{\text{train}}| = 50\text{k}$ entities for training and hold out a separate set of $|\mathcal{E}_{\text{unseen}}| = 50\text{k}$ entities for evaluating in-context knowledge utilization on unseen entities. The model is trained on a corpus of training paragraphs from these entities using the next-token prediction objective (Radford et al., 2018).

2.3 Evaluation Protocol

We periodically evaluate models on three capabilities described below (Figure 1), using exact-match accuracy over 200 randomly sampled entities every 100 steps during training.

Parametric Knowledge Utilization (PKU). This metric measures the model's ability to recall learned facts without contextual support. Given an entity $e \in \mathcal{E}_{\text{train}}$ and a test probe p_a for attribute a, the model must generate the correct value v_a solely from its parameters:

$$\text{Acc}_{\text{PKU}} = \mathbb{E}_{e \sim \mathcal{E}_{\text{train}}} \left[\frac{1}{|A_e|} \sum_{a \in A_e} \mathbf{1}\{M(p_a) = v_a\} \right]$$

where A_e is the set of attributes for entity e, $M(\cdot)$ denotes the model output, and $\mathbf{1}\{\cdot\}$ is the indicator function.

In-Context Knowledge Utilization (ICKU). This metric evaluates the model's ability to extract and utilize information provided in context for entities never seen during training. For $e \in \mathcal{E}_{\text{unseen}}$, we construct a context C containing e's paragraph along with paragraphs from two other unseen entities as distractors:

$$\text{Acc}_{\text{ICKU}} = \mathbb{E}_{e \sim \mathcal{E}_{\text{unseen}}} \left[\frac{1}{|A_e|} \sum_{a \in A_e} \mathbf{1}\{M(C, p_a) = v_a\} \right]$$

Knowledge Conflict Resolution. This metric reveals the model's preference when parametric and in-context knowledge conflict. For $e \in \mathcal{E}_{\text{train}}$, we construct a perturbed context C'_e by replacing attribute values with randomly sampled alternatives, then measure how often the model follows each source:

$$\text{Pref}_{\text{PK}} = \mathbb{E}_{e \sim \mathcal{E}_{\text{train}}} \left[\frac{1}{|A_e|} \sum_{a \in A_e} \mathbf{1}\{M(C'_e, p_a) = v_a\} \right]$$

$$\text{Pref}_{\text{ICK}} = \mathbb{E}_{e \sim \mathcal{E}_{\text{train}}} \left[\frac{1}{|A_e|} \sum_{a \in A_e} \mathbf{1}\{M(C'_e, p_a) = v'_a\} \right]$$

where v_a denotes the original (parametric) value and v'_a denotes the perturbed (in-context) value. Higher Pref_{PK} indicates stronger reliance on parametric knowledge; higher Pref_{ICK} indicates stronger reliance on in-context knowledge.

3 Experiments

We investigate how training data enable a model (i) to utilize both parametric and in-context knowledge, and (ii) to develop a robust arbitration strategy for resolving conflicts between these two sources. Specifically, we examine which characteristics of natural text corpora found on the web contribute to these capabilities and how they give rise to them.

3.1 Intra-Document Repetition Enables Co-Emergence

Motivation and hypothesis. We first examine which factors enable models to utilize both parametric and in-context knowledge. We hypothesize that *intra-document repetition*—a common property of natural text in which some information is restated within the same document (Figure 2)—plays a critical role. During next-token prediction, the first mention of a fact requires parametric recall (Geva et al., 2023; Meng et al., 2022), whereas later mentions allow the model to leverage earlier context. We hypothesize that this learning signal naturally enables the emergence of in-context knowledge utilization.

Design. To test this hypothesis, we construct two corpus variants that differ in whether attributes repeat within documents:

- SINGLE: Each document contains one paragraph per entity, so attributes appear only once.
- REPEATED: Each document contains two paraphrased paragraphs per entity. The first mention necessarily relies on parametric knowledge, while the second mention provides an opportunity for the model to use either parametric knowledge or in-context knowledge. To avoid trivial copying based solely on previously mentioned attribute types regardless of the subject, we mix multiple entities within each document. Specifically, we sample two paraphrased paragraphs for each of three distinct entities and shuffle all six paragraphs to form a single training document.

Finding 1: Repetition yields co-emergence, with in-context knowledge utilization emerging first. Figure 3 (left) shows that models trained on SINGLE develop only parametric knowledge utilization, whereas models trained on REPEATED acquire both capabilities. Moreover, in-context knowledge utilization emerges earlier than parametric knowledge

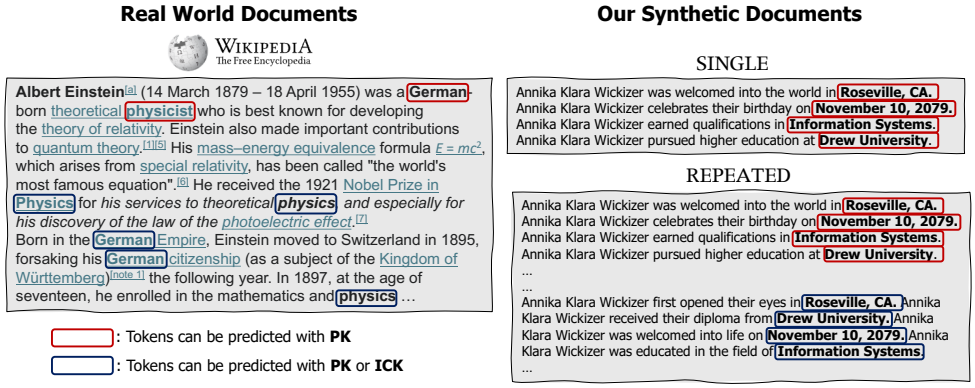

Figure 2: Intra-document repetition in real-world Wikipedia text (left) and our synthetic corpus variants (right). SINGLE contains one paragraph per entity; REPEATED contains two paraphrased paragraphs per entity, enabling in-context knowledge utilization on later mentions.

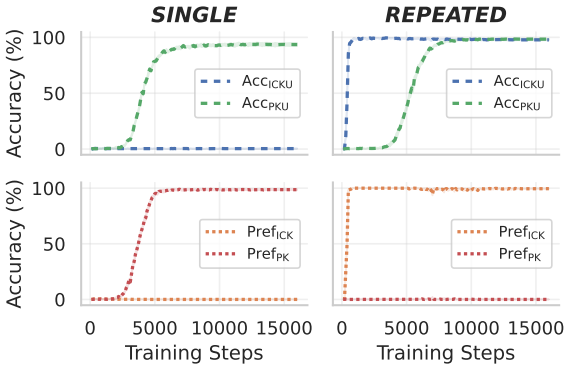

Figure 3: Evaluation results during training on SINGLE versus REPEATED. SINGLE develops only parametric knowledge utilization and always prefers parametric knowledge under conflict. REPEATED yields both capabilities, with in-context knowledge utilization emerging first, but consistently prefers in-context knowledge under conflict.

utilization. One possible explanation for this ordering is a structural asymmetry: in-context knowledge utilization can be implemented via general copying mechanisms (Olsson et al., 2022), whereas parametric knowledge utilization requires jointly learning entity-specific knowledge and a parametric recall mechanism—a combination that develops more gradually, as observed in prior work (Zucchet et al., 2025).

Finding 2: Clean REPEATED corpus induces over-reliance on context. Models trained on SINGLE cannot utilize in-context knowledge and therefore trivially prefer parametric knowledge under conflict. In contrast, models trained on REPEATED, despite possessing both capabilities, consistently prefer in-context knowledge under conflict (Figure 3, right), even when their parametric knowledge is highly confident. This is evidenced by significantly lower entropy and higher target probability for training entities (see Appendix D). Such over-reliance on context deviates from the behavior of real-world language models, which tend to prefer parametric knowledge for high-confidence facts (Yu et al., 2023; Wu et al., 2024).

3.2 Factual Inconsistency Induces Preference Transition

Motivation and hypothesis. The previous section showed that models trained on clean REPEATED data over-rely on in-context knowledge, even when their parametric knowledge is highly confident. This observation raises the question of which properties of natural web corpora discourage such unconditional reliance on in-context knowledge.

We hypothesize that a moderate degree of *within-document inconsistency* plays this role. Real-world corpora inevitably contain noise (e.g., typos, outdated statements, or paraphrased terms), making in-context evidence an imperfect signal. When contextual information is occasionally incorrect, the model may learn that parametric knowledge is more reliable for high-confidence facts. To test this hypothesis, we introduce controlled factual inconsistency into the training corpus.

Design. Starting from REPEATED, we inject inconsistency by perturbing the values of entity attributes in the *leading* paragraph with probability $p \in \{1\%, 5\%, 10\}$, replacing them with randomly sampled alternative values, while leaving the later paragraph unchanged (Figure 10).

Finding 1: Within-document inconsistency induces a preference transition. Figure 4 reveals

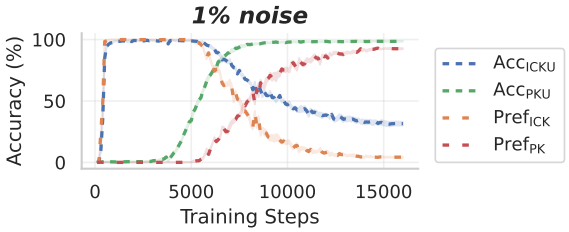

Figure 4: Effects of within-document inconsistency. With 1% inconsistency, preference shifts from in-context knowledge to parametric knowledge as training progresses.

a consistent two-stage pattern. Early in training, in-context knowledge utilization emerges first, and the model prefers in-context knowledge under conflict. As parametric knowledge utilization stabilizes, however, the model's preference gradually shifts toward parametric knowledge. Remarkably, even 1% inconsistency is sufficient to induce this transition. This behavior suggests that inconsistency imposes an effective ceiling on the reliability of context-based copying; once parametric accuracy exceeds this ceiling, the model increasingly favors parametric knowledge under conflict.

Finding 2: Inconsistency degrades in-context knowledge utilization. As the model increasingly relies on parametric knowledge, in-context knowledge utilization degrades at convergence. One plausible explanation is the gradual forgetting of in-context circuits due to reduced usage (Olsson et al., 2022). As parametric knowledge becomes more advantageous for frequently observed entities during training, in-context knowledge utilization circuits receive diminishing learning signal and gradually deteriorate. Attention analysis in Appendix E supports this interpretation: when evaluating on unseen entities, attention initially concentrates on context tokens but progressively shifts toward subject name tokens, indicating that the circuits responsible for in-context knowledge retrieval no longer function reliably.

3.3 Skewed Knowledge Distribution Preserves Balanced Reliance on Both Knowledge Sources

Motivation and hypothesis. To prevent the degradation of in-context knowledge utilization observed in the previous section, we hypothesize that the model must be continuously exposed to predictions that cannot be resolved by parametric knowledge alone during training. In natural web corpora, a vast amount of information exists where some knowledge appears very frequently while most knowledge appears only occasionally (long-tailed knowledge) (Mallen et al., 2023). We hypothesize that this skewed distribution of knowledge is key to harmoniously utilizing both in-context and parametric knowledge.

Noise	Uniform	Zipfian
1%	31.5%	84.0% (+52.5%)
5%	16.8%	63.9% (+47.1%)
10%	14.1%	57.4% (+43.3%)

Table 1: In-context knowledge utilization accuracy at the end of training. Zipfian sampling substantially mitigates degradation relative to uniform sampling under matched inconsistency levels.

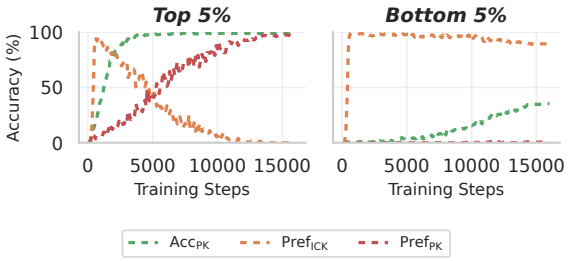

Figure 5: Arbitration behavior stratified by entity frequency. High-frequency entities transition to parametric preference, whereas low-frequency entities maintain in-context preference throughout training.

Design. We construct REPEATED corpora where entity occurrences follow a Zipfian distribution with parameter $\alpha = 1$,[1] and inject $p = 1\%$ inconsistency noise as in the previous section.

Finding 1: Long-tailed knowledge preserves in-context knowledge utilization capabilities. We hypothesized that long-tailed knowledge, for which sufficient parametric knowledge has not accumulated, would require continuous use of in-context knowledge, thereby preventing the degradation of in-context circuits observed in Section 3.2. Indeed, Table 1 shows substantially less degradation of in-context knowledge utilization under Zipfian distribution across all noise levels. However, when inconsistency noise is too high ($p > 1\%$), in-context knowledge utilization does not fully recover.

Finding 2: Frequency-dependent arbitration emerges. We also examine how the model's arbitration behavior under knowledge conflicts varies

[1]The Zipfian distribution is defined as $P(r) = r^{-\alpha} / \sum_{k=1}^{N} k^{-\alpha}$, where r denotes the frequency rank.

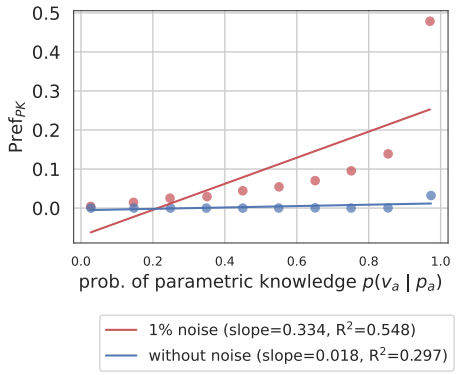

Figure 6: Parametric confidence (grouped into 10 bins) against parametric preference under conflict. (**Red**) With 1% inconsistency noise, higher confidence yields stronger parametric preference. (**Blue**) Without inconsistency noise, models show over-reliance on in-context knowledge across all confidence levels.

between entities that appear frequently versus infrequently in the training corpus. As shown in Figure 5, high-frequency entities (top 5% of all entities) transition toward parametric preference as training progresses, as observed in Section 3.2, while low-frequency entities (bottom 5% of all entities) maintain in-context preference throughout. Notably, for rare entities we observe that $\mathrm{Acc}_{\mathrm{PKU}}$ exceeds $\mathrm{Pref}_{\mathrm{PK}}$: the model *can* sometimes answer correctly via parametric recall, yet it still prefers contextual evidence under explicit conflict.

Finding 3: Skewed knowledge distribution alone is insufficient. So far, we have examined training on data with skewed knowledge distribution in the presence of inconsistency noise. However, does a long-tailed distribution alone—without inconsistency—produce confidence-calibrated arbitration? Figure 6 shows that the answer is no. We measure parametric confidence (grouped into 10 bins) against parametric preference under conflict. We first bin the probabilities of predictions on parametric knowledge probes into ten equidistant bins and plot the average $\mathrm{Pref}_{\mathrm{PK}}$ for instances in each bin. Without inconsistency noise, models show low parametric knowledge preference overall. Only the combination of skewed knowledge distribution and modest inconsistency yields the desired alignment between confidence and preference, where higher confidence leads to stronger preference for parametric knowledge.

	Pythia-6.9B		OLMo-7B	
	$\mathrm{Pref}_{\mathrm{PK}}$	$\mathrm{Pref}_{\mathrm{ICK}}$	$\mathrm{Pref}_{\mathrm{PK}}$	$\mathrm{Pref}_{\mathrm{ICK}}$
Before IT	0.8677	0.0525	0.5507	0.3894
After IT	0.1829	0.7771	0.2137	0.7155

Table 2: Knowledge conflict resolution preferences before and after instruction tuning (IT) for Pythia-6.9B and OLMo-7B. Both models show a shift from parametric knowledge preference to in-context knowledge preference after IT.

3.4 Summary

Our experiments show that the following three properties of the training corpus jointly produce robust knowledge utilization capabilities:

Intra-document repetition creates training signal for in-context knowledge utilization alongside parametric knowledge utilization, enabling their co-emergence, with in-context knowledge utilization emerging earlier and parametric knowledge utilization following later.

Within-document inconsistency limits the reliability of context-based copying, providing incentive to prefer parametric knowledge once it becomes accurate. However, this shift introduces a new problem: in-context knowledge utilization capabilities degrade as parametric knowledge becomes sufficient for most predictions.

Skewed knowledge frequency distribution resolves this tension by maintaining in-context usage when the model faces rare entities throughout training, thereby preventing degradation.

When all three properties co-occur—as they naturally do in real web corpora—they collectively enable balanced reliance on both knowledge sources and confidence-dependent arbitration. Additional experiments on hyperparameters, including the number of entities in training data, noise levels, and degree of skewness, are provided in Appendix G.

4 Further Analysis

4.1 Validation on Real-world Models

The three corpus properties examined in our controlled experiments—intra-document repetition, factual inconsistency, and skewed knowledge distribution—naturally occur in web corpora. We therefore test whether the same training patterns appear in real-world open-source LLMs.

Using the publicly released checkpoints of the PYTHIA (Biderman et al., 2023), we evaluate parametric utilization, in-context utilization, and preference under knowledge conflict at each check-

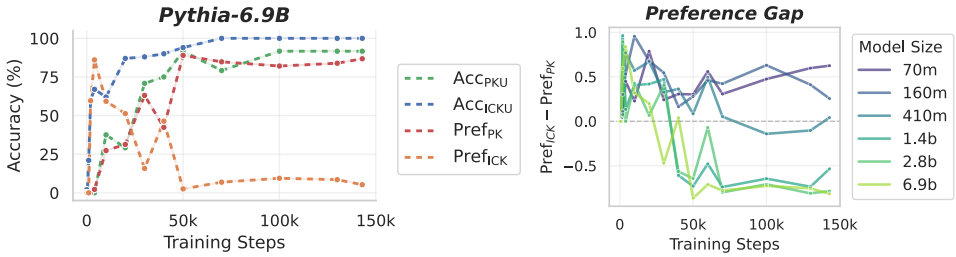

Figure 7: Evaluation results of knowledge utilization and conflict resolution in Pythia-6.9B. (**Left**) $\mathrm{Acc}_{\mathrm{ICKU}}$, $\mathrm{Acc}_{\mathrm{PKU}}$, $\mathrm{Pref}_{\mathrm{ICK}}$, and $\mathrm{Pref}_{\mathrm{PK}}$ across training steps for Pythia-6.9B. (**Right**) Preference gap ($\mathrm{Pref}_{\mathrm{ICK}} - \mathrm{Pref}_{\mathrm{PK}}$) across different model sizes, showing a consistent pattern of initial increase followed by decline as training progresses.

point (evaluation details in Appendix F).[2] model suite (Biderman et al., 2023) As shown in Figure 7 (left), PYTHIA exhibits evaluation results consistent with our controlled experiments: in-context utilization emerges earlier than parametric utilization, the model initially prefers in-context knowledge under conflict but gradually shifts toward parametric knowledge, while maintaining high $\mathrm{Acc}_{\mathrm{ICKU}}$ for novel entities throughout training. This indicates that our synthetic framework captures essential aspects of real-world pretraining, and our controlled findings could serve as a mechanistic explanation for why these behaviors emerge in practice.

To examine how this phase transition varies across model scales, we analyze the preference gap ($\mathrm{Pref}_{\mathrm{ICK}} - \mathrm{Pref}_{\mathrm{PK}}$) for models ranging from 70M to 6.9B parameters (Figure 7 right). All models exhibit a consistent pattern: initial dominance of in-context knowledge preference that gradually diminishes over training. Notably, larger models show stronger parametric knowledge preference at the end of training, with the preference gap approaching -1 for the largest models, consistent with prior observations that larger models tend to rely more heavily on their parametric knowledge (Yu et al., 2023). This trend can be attributed to larger models developing parametric knowledge more rapidly and robustly, leading to higher confidence in their internal knowledge and consequently stronger preference for it when conflicts arise.

4.2 Effects of Post-training

Our findings reveal how training corpus characteristics shape knowledge arbitration strategies during pretraining. A natural question arises: can these strategies be modified after pretraining through instruction tuning?

We examine whether instruction tuning affects arbitration behavior in real-world models. We evaluate two models (Pythia-6.9B and OLMo-7B) using the same evaluation protocol as in Section 4.1. We fine-tune both models on the Tulu dataset (Wang et al., 2023), an instruction-following dataset.

As shown in Table 2, both base models exhibit higher parametric knowledge preference. However, after post-training, both models show a reversal: parametric knowledge preference drops while in-context knowledge preference increases. This suggests that instruction tuning, which typically involves data designed to encourage faithful following of context, can significantly alter the arbitration strategies established during pretraining.

Having observed that post-training can modify model behavior, we further investigate whether adjusting inconsistency noise—a key factor for in-context knowledge reliance identified in our findings—in post-training data alone can control the model's arbitration strategies as intended. We conduct post-training on our synthetic Zipfian corpus using answer-only loss with 1,000 entities for 500 steps, which is substantially smaller in both entity count and training steps compared to pretraining.

We examine two scenarios: (1) whether a model pretrained with 1% noise and post-trained on clean data increases in-context reliance, and (2) whether a model pretrained without noise and post-trained with varying noise levels ($p \in \{1\%, 5\%, 10\%\}$) decreases in-context reliance.

The results are shown in Figure 8. We plot the confidence-preference alignment by binning entities based on parametric knowledge probability and

[2]We use PYTHIA because it provides finely-grained checkpoints throughout training, enabling evaluation at intermediate checkpoints. Results for OLMO (Groeneveld et al., 2024), another model with publicly available checkpoints, are provided in the Appendix F.

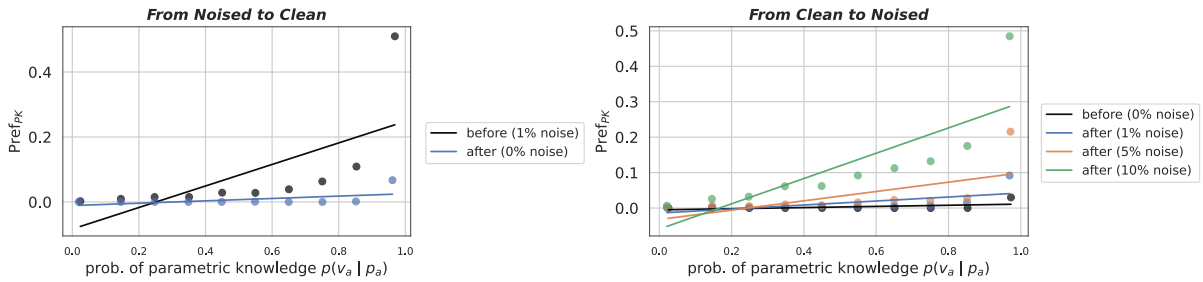

Figure 8: Alignment between parametric knowledge confidence and preference under conflict before and after post-training. (**Left**) Scenario (1): A model pretrained with 1% noise shows declined parametric knowledge preference after post-training with clean data that has no inconsistency noise. (**Right**) Scenario (2): A model pretrained without noise initially shows almost no parametric knowledge preference, but after post-training on data with noise, parametric knowledge preference gradually increases according to the model's confidence, with higher noise levels producing stronger confidence-calibrated parametric knowledge preference.

measuring $\mathrm{Pref}_{\mathrm{PK}}$ for each bin. We demonstrate that adjusting noise levels alone can successfully reshape the alignment between confidence and preference, confirming that our findings provide meaningful guidance for constructing post-training data.

5 Related Work

Knowledge Utilization and Conflicts in Language Models. Large language models store factual knowledge in their parameters during pretraining (Roberts et al., 2020; Petroni et al., 2019; Geva et al., 2020), and can also leverage external information provided in context through retrieval-augmented generation (Lewis et al., 2021; Ram et al., 2023a; Shi et al., 2023). Recent studies have shown that models can use in-context knowledge without explicit fine-tuning (Mallen et al., 2023; Ram et al., 2023b). When these sources conflict (Neeman et al., 2022), models exhibit confidence-dependent arbitration, preferring parametric knowledge for well-learned facts while deferring to context for less familiar information (Wu et al., 2024; Yu et al., 2023). Several methods have been proposed to steer this behavior through attention manipulation or contrastive decoding (Li et al.; Yu et al., 2023; Sun et al., 2025; Jin et al., 2024). However, these works primarily focus on analyzing or modifying post-hoc behavior (Kortukov et al., 2024; Xie et al., 2023; Longpre et al., 2021), leaving open the question of how these capabilities emerge during training.

Training Data Properties and Knowledge Acquisition. Prior work has studied how data characteristics shape model capabilities and how learning unfolds during training. Chan et al. (2022); Singh et al. (2023) investigate how data properties enable in-context and in-weight learning to co-exist in transformer-based classifiers. However, their work is limited to classification tasks, which may exhibit different dynamics from language models trained with next-token prediction. Other works present controlled studies with synthetic corpora to investigate parametric knowledge acquisition in language models (Allen-Zhu and Li, 2024a,b; Zucchet et al., 2025), but do not address in-context utilization (Olsson et al., 2022) or conflict resolution. We extend these directions by examining how both capabilities co-emerge in language models and how conflict arbitration strategies develop during training depending on training data characteristics.

6 Conclusion

We presented the first systematic analysis of how training data characteristics shape parametric and in-context knowledge utilization in language models through controlled experiments. We identified a counterintuitive finding: three properties commonly regarded as detrimental—intra-document repetition, within-document inconsistency, and skewed knowledge frequency distribution—must co-occur for robust knowledge utilization to emerge. Validation on real-world language models confirms that the dynamics observed in our controlled experiments generalize to real-world pretraining, and our post-training experiments demonstrate that knowledge arbitration strategies can be reshaped by adjusting data characteristics. These findings offer practical guidance for designing training data.

7 Limitations

Our study primarily leverages synthetic biography datasets, which may not fully reflect the richness of natural language. However, this design provides a clear advantage: it allows us to isolate the causal effects of individual factors—which would be infeasible with real web corpora where multiple properties are entangled. In addition, we demonstrate that the same training patterns also emerge in real-world models (Section 4.1), suggesting that our findings generalize beyond the controlled setting. Additionally, we focus on knowledge conflicts where models resolve tension based on internal confidence. Real-world retrieval-augmented generation involves diverse scenarios such as retrieval errors, partial relevance, and multi-document conflicts, which we leave for future investigation.

References

Zeyuan Allen-Zhu and Yuanzhi Li. 2024a. Physics of language models: Part 3.1, knowledge storage and extraction. *Preprint*, arXiv:2309.14316.

Zeyuan Allen-Zhu and Yuanzhi Li. 2024b. Physics of language models: Part 3.2, knowledge manipulation. *Preprint*, arXiv:2309.14402.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, and 1 others. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland, and Felix Hill. 2022. Data distributional properties drive emergent in-context learning in transformers. *Advances in neural information processing systems*, 35:18878–18891.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. *arXiv preprint arXiv:2304.14767*.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2020. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, and 24 others. 2024. Olmo: Accelerating the science of language models. *Preprint*, arXiv:2402.00838.

Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. 2023. Linearity of relation decoding in transformer language models. *arXiv preprint arXiv:2308.09124*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and 1 others. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Zhuoran Jin, Pengfei Cao, Hongbang Yuan, Yubo Chen, Jiexin Xu, Huaijun Li, Xiaojian Jiang, Kang Liu, and Jun Zhao. 2024. Cutting off the head ends the conflict: A mechanism for interpreting and mitigating knowledge conflicts in language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1193–1215, Bangkok, Thailand. Association for Computational Linguistics.

Evgenii Kortukov, Alexander Rubinstein, Elisa Nguyen, and Seong Joon Oh. 2024. Studying large language model behaviors under context-memory conflicts with real documents. *Preprint*, arXiv:2404.16032.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

Gaotang Li, Yuzhong Chen, and Hanghang Tong. Taming knowledge conflicts in language models. In *Forty-second International Conference on Machine Learning*.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. *arXiv preprint arXiv:2109.05052*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. 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), pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

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

Ella Neeman, Roee Aharoni, Or Honovich, Leshem Choshen, Idan Szpektor, and Omri Abend. 2022. Disentqa: Disentangling parametric and contextual knowledge with counterfactual question answering. *arXiv preprint arXiv:2211.05655*.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, and 1 others. 2022. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. 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)*, pages 2463–2473.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. Improving language understanding by generative pre-training.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023a. In-context retrieval-augmented language models. *Preprint*, arXiv:2302.00083.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023b. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

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

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *Preprint*, arXiv:2301.12652.

Aaditya Singh, Stephanie Chan, Ted Moskovitz, Erin Grant, Andrew Saxe, and Felix Hill. 2023. The transient nature of emergent in-context learning in transformers. *Advances in neural information processing systems*, 36:27801–27819.

Zhongxiang Sun, Xiaoxue Zang, Kai Zheng, Yang Song, Jun Xu, Xiao Zhang, Weijie Yu, Yang Song, and Han Li. 2025. Redeep: Detecting hallucination in retrieval-augmented generation via mechanistic interpretability. *Preprint*, arXiv:2410.11414.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023. How far can camels go? exploring the state of instruction tuning on open resources. *Preprint*, arXiv:2306.04751.

Kevin Wu, Eric Wu, and James Y Zou. 2024. Clasheval: Quantifying the tug-of-war between an llm's internal prior and external evidence. *Advances in Neural Information Processing Systems*, 37:33402–33422.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2023. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations*.

Qinan Yu, Jack Merullo, and Ellie Pavlick. 2023. Characterizing mechanisms for factual recall in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9924–9959, Singapore. Association for Computational Linguistics.

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

A Synthetic Biographies Dataset Construction

Following prior work (Allen-Zhu and Li, 2024a; Zucchet et al., 2025), we first construct N synthetic person profiles. Each profile contains four attributes: `birth_date`, `birth_city`, `university`, and `major`. Names (first/middle/last) are sampled by randomly composing entries from a public name database.[3] For `birth_date`, we sample a date uniformly from 1900–2099. For `birth_city` and `university`, we sample from curated lists of 200 values each, and for `major` from a list of 100 values, all derived from Wikipedia.[4] For each attribute, we sample 7 distinct surface templates from

[3] https://github.com/smashew/NameDatabases/tree/master/NamesDatabases

[4] https://en.wikipedia.org/wiki/

a finite template pool. An example of templates for `birth_date` is shown below.

An example of templates for `birth_date`

1. person was born on birth_date.
2. person came into the world on birth_date.
3. person entered this world on birth_date.
4. person was brought into the world on birth_date.
5. person took their first breath on birth_date.
6. person began their life journey on birth_date.
7. person celebrates their birthday on birth_date.
8. person first opened their eyes on birth_date.
9. person was welcomed into life on birth_date.
10. person arrived on birth_date.
11. person's story started on birth_date.
12. person was born to the world on birth_date.
13. person was delivered into the world on birth_date.
14. person was given life on birth_date.
15. person was welcomed into the world on birth_date.
16. person began their journey on Earth on birth_date.
17. person made their debut in the world on birth_date.
18. person became a part of the world on birth_date.
19. person was born into this life on birth_date.
20. person came to life on birth_date.

We then create paragraphs containing each person's biography with a randomized attribute order as follows: using 6 of the templates, we generate six paragraphs per entity; five are reserved for training and one is used as the evaluation in-context paragraph. The remaining (seventh) template is held out as a closed-style test probe designed to elicit the target attribute. An illustration of the resulting dataset is shown in Figure 9.

B Details on Training Language Models

Component	Value
Embedding dimension	512
Layers	8
Attention heads	8
FFN inner dimension	2048
Context length	512

Table 3: Model architecture.

For our controlled experiments, we use a GPT-2–style decoder-only Transformer[5]. The model configuration is summarized in Table 3. Following Hoffmann et al. (2022), we adopt the settings used in Zucchet et al. (2025). The training hyperparameters are listed in Table 4. All experiments are implemented using the `HuggingFace TRL` library[6] and conducted on a single NVIDIA A100 GPU. Each training run requires approximately 4–6 hours.

Hyperparameter	Value
Max training steps	16,000
Batch size	128
Learning rate	4×10^{-4}
Weight decay	0.10
LR scheduler	Cosine
Sequence length	512
Numerical precision	bfloat16

Table 4: Training hyperparameters.

C Example of Factual Inconsistency Noise within a Document

Figure 10 illustrates a document from the REPEATED corpus in which factual inconsistency noise has been injected. The value highlighted in pink was injected as noise with some probability and therefore does not match the latter original value, "November 10, 2079."

D Measuring Confidence for Parametric Knowledge

To assess the model's confidence in its parametric knowledge, we measure two key metrics at the final token position of each test probe: (1) the probability assigned to the target token, and (2) the entropy of the probability distribution over the vocabulary. These metrics provide complementary perspectives on model confidence—target probability reflects how strongly the model predicts the correct answer, while entropy captures the overall uncertainty in the prediction.

	$\mathcal{E}_{\text{train}}$	$\mathcal{E}_{\text{unseen}}$
w/o noise		
Target prob.	0.998	0.024
Entropy (nats)	0.011	0.955
w/ 1% noise		
Target prob.	0.997	0.034
Entropy (nats)	0.016	1.236

Table 5: Target token probability and entropy measured at the last token of the test probe for entities in $\mathcal{E}_{\text{train}}$ and $\mathcal{E}_{\text{unseen}}$.

[5] https://huggingface.co/openai-community/gpt2

[6] https://huggingface.co/docs/trl/index

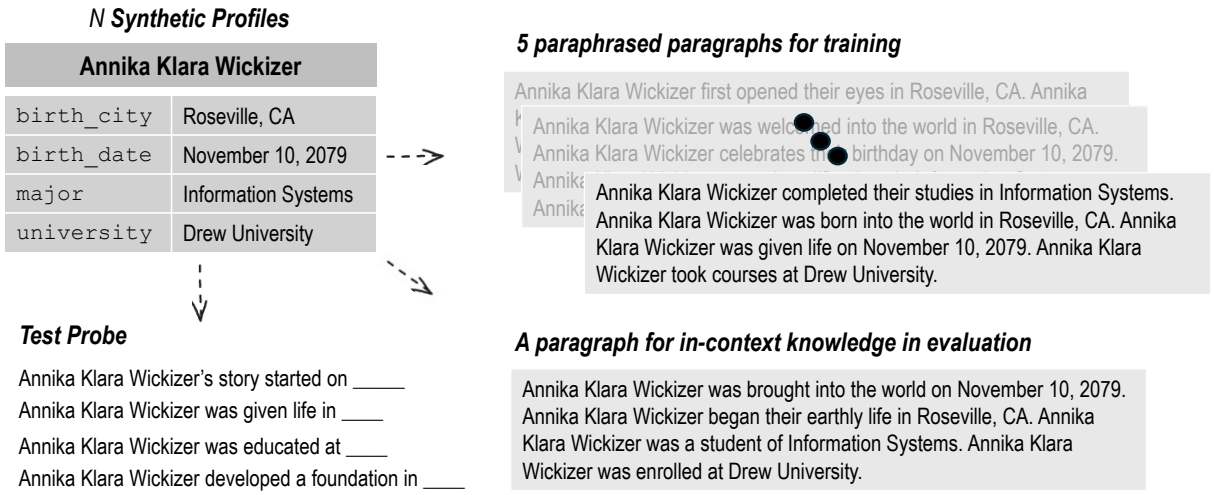

Figure 9: An example of the synthetic dataset. Each profile consists of four attributes (`birth_date`, `birth_city`, `university`, `major`), with paragraphs for training, a paragraph for in-context knowledge in evaluation, and test probes for eliciting the model to generate the attributes of each entity.

Annika Klara Wickizer was welcomed into the world in Roseville, CA. Annika Klara Wickizer celebrates their birthday on August 5, 1999 . Annika Klara Wickizer earned qualifications in Information Systems. Annika Klara Wickizer pursued higher education at Drew University.

Dara Angila Honey was given life on April 6, 1978. Dara Angila Honey focused their academic efforts on Industrial. Dara Angila Honey entered this world in Indianapolis, IN. Dara Angila Honey achieved academic success at Fisk University.

Dara Angila Honey chose Industrial as their field of study. Dara Angila Honey completed a program at Fisk University. Dara Angila Honey was welcomed into life on April 6, 1978. Dara Angila Honey became a part of the world in Indianapolis, IN.

Annika Klara Wickizer first opened their eyes in Roseville, CA. Annika Klara Wickizer received their diploma from Drew University. Annika Klara Wickizer was welcomed into life on **November 10, 2079.** Annika Klara Wickizer was educated in the field of Information Systems.

Roselee Justine Woolem gained academic grounding in Business Analytics. Roselee Justine Woolem first opened their eyes in Phoenix, AZ. Roselee Justine Woolem studied at Hamilton College. Roselee Justine Woolem was brought into the world on August 12, 2083.

Roselee Justine Woolem entered this world on August 12, 2083. Roselee Justine Woolem majored in Business Analytics. Roselee Justine Woolem began their life in Phoenix, AZ. Roselee Justine Woolem developed expertise at Hamilton College.

Figure 10: Example of the document injected inconsistency noise

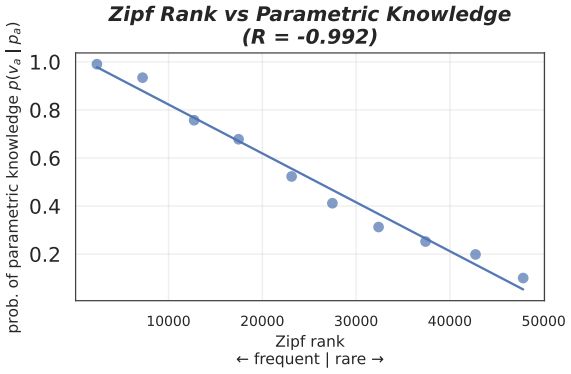

Figure 11: Relationship between entity frequency (Zipf rank) and parametric knowledge strength. The model exhibits a strong negative correlation between Zipf rank and the probability $p(v_a \mid p_a)$ of generating the correct attribute value given only the probe prompt. More frequent entities have stronger parametric knowledge while rare entities show weaker parametric knowledge.

Table 5 presents these measurements for entities in both $\mathcal{E}_{\text{train}}$ (seen during training) and $\mathcal{E}_{\text{unseen}}$ (held-out entities) under two training conditions: without noise and with 1% inconsistency noise. For $\mathcal{E}_{\text{train}}$, the model exhibits extremely high confidence, assigning near-perfect probability to target tokens with very low entropy. This indicates that the model has successfully acquired and can reliably retrieve parametric knowledge for entities it encountered during training. In contrast, for $\mathcal{E}_{\text{unseen}}$, the model shows substantially lower confidence, with target probabilities and much higher entropy values.

Parametric knowledge varies with entity frequency. Beyond the binary comparison between seen and unseen entities, we investigate how the strength of parametric knowledge varies across entities in $\mathcal{E}_{\text{train}}$ as a function of their frequency in the pre-training corpus. Figure 11 shows the relationship between Zipf rank (where lower ranks indicate more frequent entities) and the probability $p(v_a \mid p_a)$ assigned to the correct attribute value when given only the probe prompt p_a, averaged across all attributes. The results reveal an strong negative correlation, demonstrating that parametric knowledge strength is tightly coupled with entity frequency.

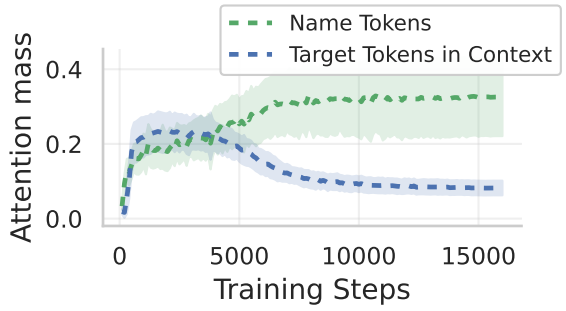

Figure 12: Changes in the layer-wise sum of attention mass at the last token of the test probe when the model trained with 1% noise performs in-context knowledge utilization for $\mathcal{E}_{\text{unseen}}$ entities. **Green** indicates the attention allocated to name tokens in the test probe, while **blue** indicates the attention allocated to target tokens in the context. As training progresses, attention gradually shifts from context target tokens to name tokens.

E Attention Pattern Analysis

In this section, we provide additional analysis of attention patterns to understand the mechanisms underlying the degradation of in-context knowledge utilization observed in Section 3.2. Specifically, we investigate how the model trained on the REPEATED corpus with 1% inconsistency noise allocates attention during the training process, which allows us to indirectly examine the circuits used for parametric versus in-context knowledge utilization.

Attention Patterns in In-Context Knowledge Utilization To understand why in-context knowledge utilization degrades when trained with inconsistency noise, we analyzed the attention patterns at the last token position of the test probe during in-context knowledge utilization for $\mathcal{E}_{\text{unseen}}$ entities. Figure 12 shows the layer-wise sum of attention mass over the course of training. We distinguish between two types of attention targets: (1) name tokens in the test probe (shown in green), which are more associated with parametric knowledge retrieval (Meng et al., 2022; Zucchet et al., 2025; Geva et al., 2023), and (2) target attribute tokens in the context (shown in blue), which are necessary for in-context knowledge utilization (Olsson et al., 2022).

Our analysis reveals that early in training, attention is heavily concentrated on target attribute tokens in the context, which is consistent with successful in-context knowledge utilization mediated by in-context induction circuits (Olsson et al., 2022). However, as training progresses and parametric knowledge utilization stabilizes, attention gradually shifts toward name tokens in the test probe. Notably, this shift occurs even when evaluating on $\mathcal{E}_{\text{unseen}}$ entities—entities for which the model has no parametric knowledge (see Table 5).

We hypothesize that the presence of inconsistency noise during training introduces imperfection in in-context knowledge utilization, making contextual information a less reliable signal. As a result, once parametric knowledge becomes sufficiently stable, the model increasingly defaults to parametric knowledge retrieval across all situations. Consequently, in-context knowledge utilization circuits receive progressively less training signal. In combination with regularization effects such as weight decay (Loshchilov and Hutter, 2017), this reduced usage leads to a gradual degradation of the model's ability to utilize in-context knowledge over the course of training.

This mechanistic perspective helps explain why a skewed knowledge distribution (Section 3.3) is necessary to preserve in-context knowledge utilization. The continuous presence of unfamiliar or low-frequency entities in the training distribution forces the model to repeatedly rely on in-context information, thereby preventing the complete abandonment of in-context knowledge circuits.

F Experimental Details for Real-World Large Language Models

We adapt the evaluation scenarios used in our controlled experiments to settings applicable to large language models trained on real-world web corpora. Since such corpora contain abundant information about countries and their capitals, we define the set of training entities $\mathcal{E}_{\text{train}}$ as *Real-World Countries* and evaluate whether the model can correctly predict their corresponding capital cities.

To this end, we construct a *Real-World Country–Capital Set* based on the country–capital pairs introduced in Hernandez et al. (2023). Using this dataset, we build question–answer style probes as illustrated in Figure 13 and define the **Parametric Knowledge Utilization** (PKU) scenario. We measure Acc_{PKU} by checking whether the correct capital appears within the first 64 generated tokens.

For the **In-Context Knowledge Utilization** (ICKU) scenario, we evaluate the model's ability to use knowledge that is not present in its pretraining data. Specifically, we construct 100 artificial country–capital pairs that do not correspond to any

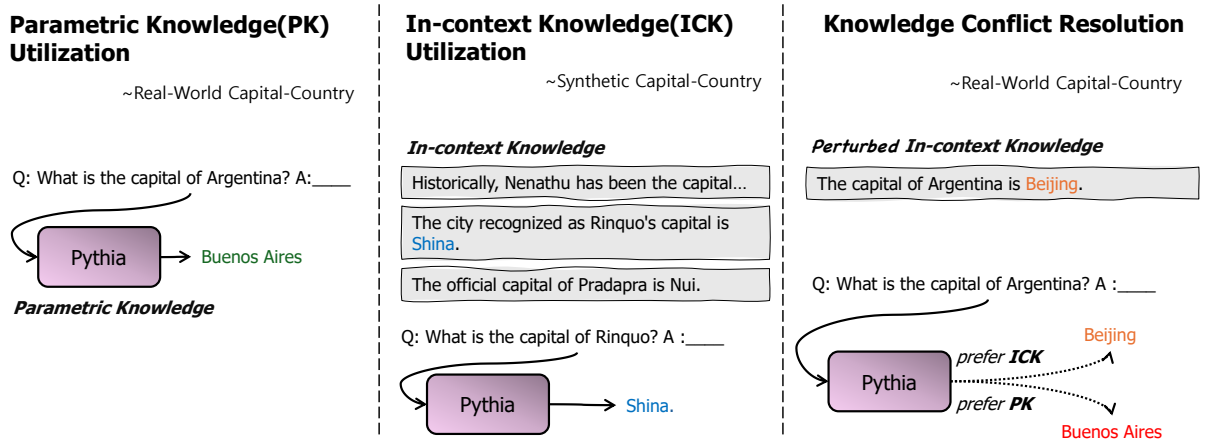

Figure 13: Three knowledge utilization scenarios in real-world large language models. **Left:** Parametric knowledge utilization, where the model recalls country–capital facts from real-world data that were encoded in its parameters during training. **Middle:** In-context knowledge utilization, where the model relies on synthetic country–capital pairs provided only in the context. **Right:** Knowledge conflict resolution, where the model is queried about real-world countries while the prompt supplies perturbed (incorrect) capitals, allowing us to examine whether the model prefers parametric knowledge or the perturbed in-context knowledge.

real-world entities, forming a *Synthetic Country–Capital Set*. These pairs are provided only within the prompt context, and $\mathrm{Acc}_{\mathrm{ICKU}}$ is computed by verifying whether the correct synthetic capital is generated within 64 tokens.

Finally, for **Knowledge Conflict Resolution**, we perturb the in-context knowledge by replacing the true capitals in the *Real-World Country–Capital Set* with incorrect alternatives. Given these perturbed contexts and the corresponding test probes, we evaluate whether the model follows the in-context information or instead relies on its parametric knowledge. This allows us to compute $\mathrm{Pref}_{\mathrm{ICK}}$ and $\mathrm{Pref}_{\mathrm{PK}}$, reflecting the model's preference under explicit knowledge conflict.

In addition to the Pythia models discussed in Section 4.1, we conduct the same set of experiments on OLMo-7B (Groeneveld et al., 2024). As shown in Figure 14, OLMo-7B exhibits qualitatively similar patterns to those observed in Pythia: in-context knowledge utilization emerges earlier, followed by the stabilization of parametric knowledge utilization, and preference shifts in resolving conflicts between parametric and in-context knowledge. These results suggest that the knowledge utilization dynamics identified in our analysis are not specific to a particular model family, but instead generalize across different large-scale language models trained on real-world data.

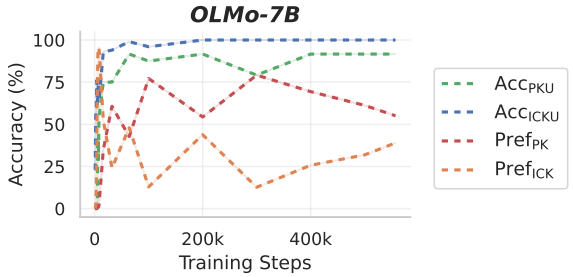

Figure 14: Evaluation results of knowledge utilization and conflict resolution in OLMo-7B.

G Additional Experimental Results

We further present additional experimental results varying several factors of the characteristics of training data. Unless otherwise noted, all experiments are conducted on the REPEATED corpus.

G.1 Effect of the Number of Training Entities

Figure 15 compares REPEATED runs with 50k, 100k, and 200k training entities. With 50k entities, both in-context knowledge utilization ($\mathrm{Acc}_{\mathrm{ICKU}}$) and parametric knowledge utilization ($\mathrm{Acc}_{\mathrm{PKU}}$) emerge, with $\mathrm{Acc}_{\mathrm{ICKU}}$ activating earlier and $\mathrm{Acc}_{\mathrm{PKU}}$ following as training stabilizes. In contrast, for 100k and 200k entities, $\mathrm{Acc}_{\mathrm{PKU}}$ fails to rise: the model learns to use in-context knowledge but does not develop robust parametric utilization.

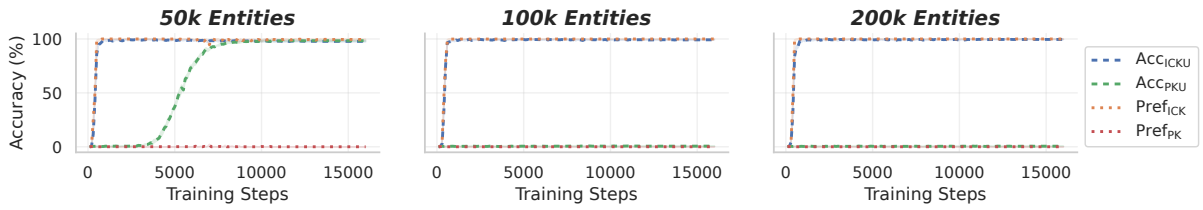

Figure 15: Evaluation results during training of in-context knowledge utilization ($\mathrm{Acc}_{\mathrm{ICKU}}$), parametric knowledge utilization ($\mathrm{Acc}_{\mathrm{PKU}}$), and knowledge conflict preferences ($\mathrm{Pref}_{\mathrm{ICK}}$, $\mathrm{Pref}_{\mathrm{PK}}$) under different numbers of training entities.

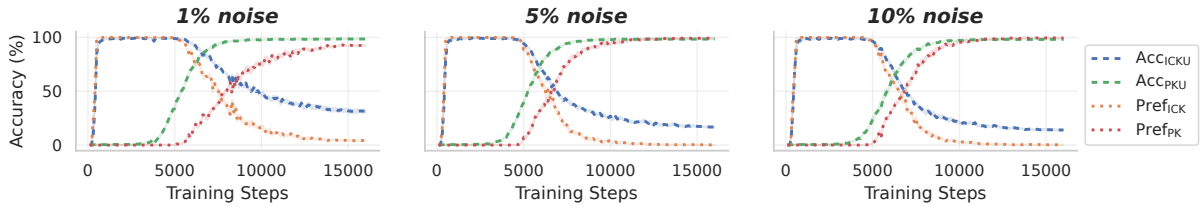

Figure 16: Evaluation results during training of in-context knowledge utilization ($\mathrm{Acc}_{\mathrm{ICKU}}$), parametric knowledge utilization ($\mathrm{Acc}_{\mathrm{PKU}}$), and knowledge conflict preferences ($\mathrm{Pref}_{\mathrm{ICK}}$, $\mathrm{Pref}_{\mathrm{PK}}$) under different levels of intra-document inconsistency noise.

G.2 Effect of Intra-document Inconsistency Noise

Figure 16 examines training dynamics under intra-document factual inconsistency levels of 1%, 5%, and 10%. Even 1% noise is sufficient to induce a phase shift in conflict-time preference: as $\mathrm{Acc}_{\mathrm{PKU}}$ stabilizes, the model transitions from preferring in-context knowledge ($\mathrm{Pref}_{\mathrm{ICK}}$) to preferring parametric knowledge ($\mathrm{Pref}_{\mathrm{PK}}$). Increasing noise accelerates this shift but also degrades $\mathrm{Acc}_{\mathrm{ICKU}}$ at convergence, indicating over-reliance on parametric knowledge and a reduced ability to use in-context knowledge.

G.3 Effect of Distributional Skew

Figure 17 examines training dynamics under Zipfian sampling with $\alpha \in \{0.5, 1.0, 2.0\}$. A near-uniform regime (α=0.5) yields progressive degeneration of $\mathrm{Acc}_{\mathrm{ICKU}}$ over training, consistent with the model drifting toward parametric recall even for unfamiliar entities. An overly skewed regime (α=2.0) produces undesirable dynamics—parametric utilization fails to activate—suggesting that extreme concentration of exposure undermines balanced capability growth. A moderate skew (α=1.0) best preserves $\mathrm{Acc}_{\mathrm{ICKU}}$ for rare or novel entities while still supporting stable $\mathrm{Acc}_{\mathrm{PKU}}$ and a robust preference for parametric knowledge on frequently seen facts.

H The Use of Large Language Models

We used large language models solely to aid and polish the writing of this paper, including tasks such as grammar correction, wording refinement, and minor stylistic edits.

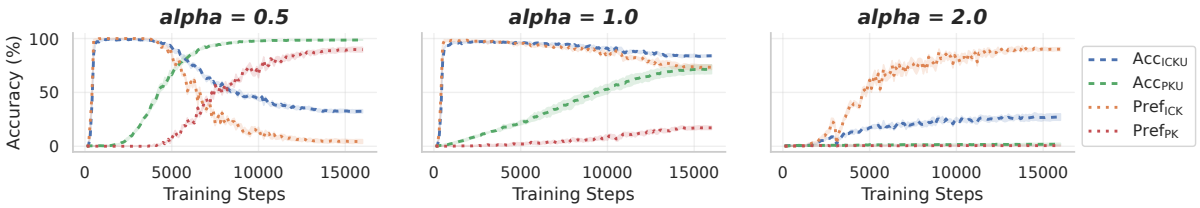

Figure 17: Evaluation results during training of in-context knowledge utilization ($\mathrm{Acc}_{\mathrm{ICKU}}$), parametric knowledge utilization ($\mathrm{Acc}_{\mathrm{PKU}}$), and knowledge conflict preferences ($\mathrm{Pref}_{\mathrm{ICK}}$, $\mathrm{Pref}_{\mathrm{PK}}$) as a function of the Zipf exponent α.