How to inject knowledge efficiently? Knowledge Infusion Scaling Law for Pre-training Large Language Models

Kangtao Lv[1,2], Haibin Chen[2], Yujin Yuan[2], Langming Liu[2], Shilei Liu[2], Yongwei Wang[1,3], Wenbo Su[2], Bo Zheng[2]*
[1]Zhejiang University
[2]Taobao & Tmall Group of Alibaba
[3]Shanghai AI Laboratory
{lvkangtao.lkt, chenhaibin.chb, yujin.yyj, bozheng}@alibaba-inc.com

Abstract

Large language models (LLMs) have attracted significant attention due to their impressive general capabilities across diverse downstream tasks. However, without domain-specific optimization, they often underperform on specialized knowledge benchmarks and even produce hallucination. Recent studies show that strategically infusing domain knowledge during pretraining can substantially improve downstream performance. A critical challenge lies in balancing this infusion trade-off: injecting too little domain-specific data yields insufficient specialization, whereas excessive infusion triggers catastrophic forgetting of previously acquired knowledge. In this work, we focus on the phenomenon of memory collapse induced by over-infusion. Through systematic experiments, we make two key observations, i.e. 1) Critical collapse point: each model exhibits a threshold beyond which its knowledge retention capabilities sharply degrade. 2) Scale correlation: these collapse points scale consistently with the model's size. Building on these insights, we propose a **knowledge infusion scaling law** that predicts the optimal amount of domain knowledge to inject into large LLMs by analyzing their smaller counterparts. Extensive experiments across different model sizes and pertaining token budgets validate both the effectiveness and generalizability of our scaling law.

1 Introduction

Recent advancements in LLMs (OpenAI, 2024; DeepSeek-AI, 2024; Grattafiori and et al., 2024) have demonstrated remarkable capabilities across diverse tasks. Following the established scaling law (Kaplan et al., 2020; Hoffmann et al., 2022; Aghajanyan et al., 2023; Muennighoff et al., 2023; Isik et al., 2024), the prevailing paradigm pretrains LLMs on massive corpora before fine-tuning them on downstream tasks.

Despite their generalist capabilities, off-the-shelf LLMs often underperform in specialized domains, suffering from knowledge misalignment and hallucinations when their pretraining data lack domain-specific coverage (Xi et al., 2024). Pretraining is the phase where models acquire both linguistic competence and broad general knowledge (Zhang et al., 2024), yet prior work shows that without targeted interventions, they often fail to ground specialized knowledge fully and may degrade in generalization (Charton and Kempe, 2024; Dohmatob et al., 2024). To mitigate this, recent studies have explored injecting domain knowledge directly into the pertaining stage to boost memorization fidelity (Xi et al., 2024; Srivastava et al., 2024). However, the optimal "dose" of knowledge infusion remains elusive due to differences in model architectures and training data.

Compounding this challenge, LLM training costs have soared where single runs can exceed hundreds of millions of dollars (Yang et al., 2024; Grattafiori and et al., 2024), making exhaustive trial-and-error experimentation infeasible. This motivates the design of predictive scaling laws: by studying smaller models, we can forecast optimal training configurations for larger ones.

Research question. *How much knowledge should be infused during LLM pretraining to maximize both memorization and generalization?* We address this by systematically varying infusion frequency, model scale (137M–3B parameters), and training tokens (up to 100B). We construct infusion corpora by randomly sampling triples from Wikidata (Pellissier Tanon et al., 2016) and convert them into natural language corpus form for infusion. We then conduct controlled experiments across model sizes and token budgets to quantify knowledge retention dynamics.

Our experiments uncover a **Memory Collapse Phenomenon** that models exhibit degradation in knowledge retention beyond a model-specific infusion threshold. Intriguingly, these collapse points correlate with the model scale, where larger mod-

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 26205–26220
November 4-9, 2025

els demonstrate greater memorization capacity yet reach collapse points earlier, implying they require proportionally less infusion to saturate their parametric capacity.

This trade-off presents a dilemma: under-infusion leaves long-tail facts underground and increases hallucination (Kandpal et al., 2023a; Shuster et al., 2021), whereas over-infusion induces overfitting and catastrophic forgetting (Hernandez et al., 2022). To navigate this balance, we propose a **Knowledge Infusion Scaling Law** that predicts the optimal infusion quantities for large LLMs by extrapolating from small-scale experiments, which helps dramatically reduce the computational budget required to tune domain-specific pertaining.

Overall, our main contributions are summarized as follows:

- We identify and characterize the memory collapse phenomenon in LLM pertaining under excessive knowledge infusion.
- We derive a knowledge infusion scaling law for LLMs that quantifies how optimal infusion scales with model size and token budget.
- We conduct comprehensive experiments to validate the effectiveness of our scaling law. Notably, we perform large-scale training corpus experiments on larger model sizes, closely simulating real-world training scenarios, thereby demonstrating its practical utility for efficient LLM pretraining.

2 Related work

2.1 Infusing Knowledge into LLMs

During pre-training, large language models assimilate an extensive repository of knowledge within their parameters (Jin et al., 2024), effectively positioning pretrained language models as knowledge bases (Petroni et al., 2019; AlKhamissi et al., 2022). Subsequent studies (He et al., 2024; Zheng et al., 2024; Zhong et al., 2024) have further corroborated that LLMs exhibit remarkable performance on a variety of knowledge-intensive tasks by leveraging their substantial parametric memories. However, recent investigations have indicated that LLMs are notably deficient in acquiring long-tail facts—those associated with less common entities—or when the relevant knowledge is particularly rare (Kandpal et al., 2023b; Mallen et al., 2023; Wang et al., 2023; Liu et al., 2025). The process by which language models imbibe knowledge during pretraining remains a compelling area of research. (Chang et al., 2024) have examined how LLMs acquire factual knowledge during the pretraining phase, albeit primarily focusing on the knowledge encoded once pretraining is complete. Moreover, other researchers (Roberts et al., 2020) have demonstrated that fine-tuning pretrained models can effectively "inject" additional knowledge, thereby enhancing their capability to answer factual queries. In contrast to this research, our work delves into the factors influencing the memorization of factual knowledge—especially long-tail facts—during the pre-training phase.

2.2 Scaling Law

(Kaplan et al., 2020) and (Hoffmann et al., 2022) were pioneers in positing the functional form of language modeling losses as a power function, dependent on both the number of model parameters and the size of the training data. Many subsequent studies (Bhagia et al., 2024; Lu et al., 2024; Que et al., 2024) that adhere to this framework provide a predictive structure for determining the most efficient configurations for expanding models, leveraging insights gained from smaller models (Gao et al., 2023). These efforts contribute significantly to understanding the scaling behaviors of LLMs and offer practical guidance for training large models. Recently, (Lu et al., 2024) investigated the scaling laws of LLMs' fact memorization and the behaviors associated with memorizing different types of facts. Concurrently with our work, studies (Charton and Kempe, 2024; Dohmatob et al., 2024) have uncovered a critical performance degradation caused by excessive injection of data into the training corpus. Different from these works, our paper focuses on the scaling laws of fact memorization and the frequency of knowledge infusion within pretraining corpora. Unlike their qualitative observations of the collapse phenomenon, we are able to predict the precise moment when this collapse phenomenon occurs.

3 Methodology

Our methodology investigates how to determine the optimal quantity of knowledge infusion during LLM pretraining to maximize memorization for downstream tasks. We analyze the relationship between memorization capability and three variables: model size (N), training tokens (D), and

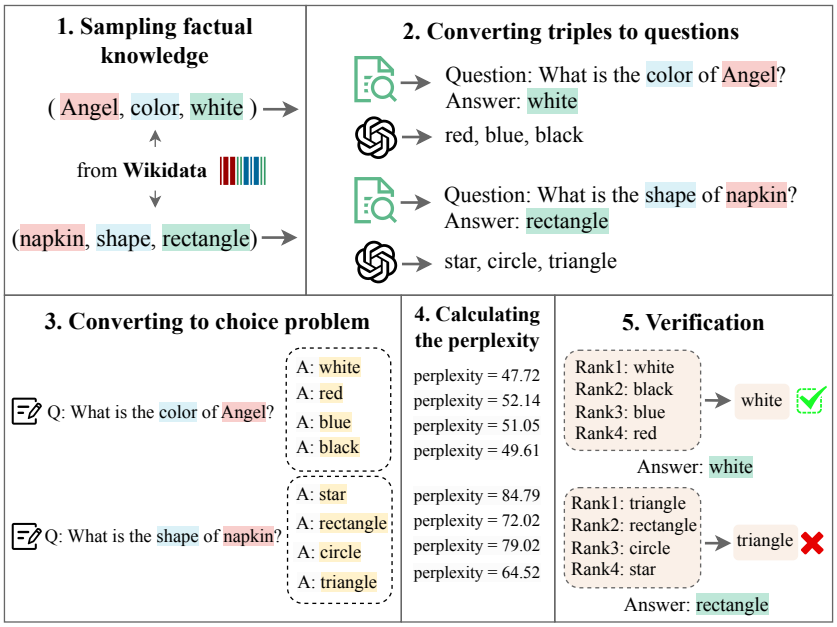


Figure 1: The pipeline of evaluation. We first employ natural language templates to convert factual knowledge triplets into natural language questions. One correct answer corresponds to the object, while the three negative options are generated with the assistance of GPT-4. Then we transform each question into a four-option multiple-choice format. Finally, calculate the PPL of each option separately and match the option with the lowest PPL with the answer.

knowledge frequency in the training corpus (F). In this section, we first explores the Memory Collapse Phenomenon under varying knowledge frequencies, then formalizes the Memory Infusion Scaling Law, and finally examines frequency effects across different training token scales.

3.1 Memory Collapse Phenomenon

To isolate the impact of existing knowledge, we rigorously filtered the training corpus by removing any text containing entities, relations, or subjects overlapping with the evaluation dataset, ultimately resulting in 58B training corpus. Then systematically infused controlled amounts of domain knowledge into the processed corpus. Specifically, we converted knowledge triples from the evaluation dataset into natural language statements using predefined templates (see Appendix A for template details) and randomly inserted these statements into the processed corpus. The knowledge injection frequency refers to the number of times each knowledge was inserted into the training corpus.

We conducted pretraining on various model scales using the knowledge-infused corpora and evaluated knowledge retention by constructing evaluation questions from the original triples. The evaluation process (illustrated in Figure 1 and detailed in Section 4.2) measures the model's ability to recall injected knowledge after pretraining.

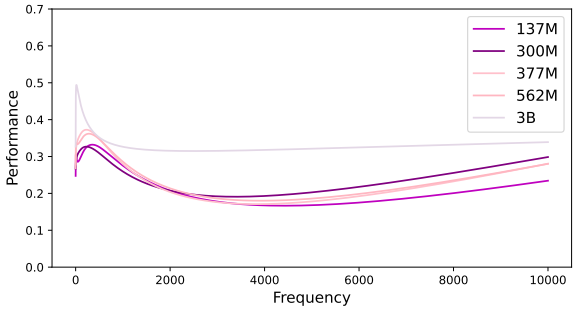


Figure 2: The fit curve of injection frequencies spanning {10, 100, 200, 500, 1000, 10000}. Different model sizes vary injection frequencies when trained with 58B training tokens.

To investigate model behavior under varying knowledge quantities, we experimented with injection frequencies spanning {10, 100, 200, 500, 1000, 10000}. Our analysis of the evaluation results reveals a Memory Collapse Phenomenon: excessive knowledge injection leads to catastrophic degradation of retention performance. As shown in Figure 2, increasing injection frequency beyond a critical threshold (termed the memory collapse point) paradoxically reduces knowledge retention, with models eventually performing worse than baseline (no injection). Notably, we observe strong model-scale dependency:

(1) Larger models reach their collapse point earlier (i.e., at lower injection frequencies)

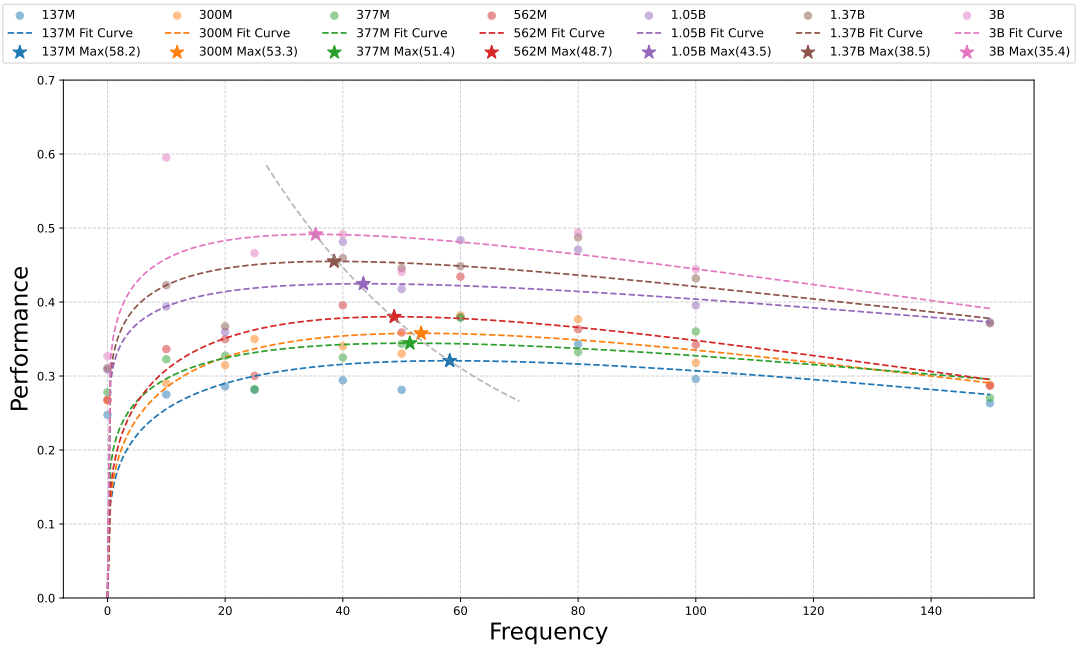


Figure 3: Prediction of optimal frequency across different model scales. The solid circular points denote the memorization performance of differently sized models on the evaluation dataset after pre-training, while dashed curves are performance prediction curves fitted to these points using the Equation 1. Each star (★) marks the maximum of a fitted curve, indicating the predicted optimal frequency. A gray dashed line connects these optimal frequency points, demonstrating a clear scaling law where larger models achieve their peak performance at lower frequency.

(2) Optimal injection quantity inversely correlates with model size

This suggests that larger models achieve knowledge saturation with fewer injected knowledge, revealing in large language models' ability to scale knowledge absorption with parameter count.

3.2 Knowledge Infusion Scaling Law

Section 3.1 demonstrates that excessive knowledge injection degrades model retention, suggesting that sparse token-level knowledge infusion suffices even in large corpora. To optimize pretraining for downstream task performance, it is critical to predict the memory collapse point and strategically allocate training data composition. Our primary objective is to precisely model the conditions triggering this collapse.

Given that model behavior deviates significantly only near the collapse point, we focus on fine-grained experiments within the critical regime while deprioritizing distant regions. We conduct systematic sweeps across injection frequencies on the small model to establish high-resolution performance prediction.

To derive the Memory Infusion Scaling Law, we first develop a predictive equation mapping injection frequency to memorization performance (P). The parametric form must intrinsically reflect observed data trends (Section 3.1) while accommodating scaling principles. After comparative analysis (see Section 4.3), we propose the following parameterization:

$$P(F) = a \cdot F^b \cdot \exp(-c \cdot F) \tag{1}$$

where F denotes knowledge injection frequency, and $\{a, b, c\}$ are learnable parameters. We optimize these parameters using the L-BFGS-B algorithm (Liu and Nocedal, 1989) as implemented in `scipy.minimize()`. Equation 1 enables prediction of the optimal injection frequency for a single model with fixed pretraining tokens.

Although the Equation 1 was initially derived from an empirical fit, its functional structure is strongly motivated by theoretical considerations that capture the dual effects observed in our experiments. In the low-frequency regime, the term F^b models the increase in knowledge retention due to repeated exposure. This mirrors classical scaling law curves where performance typically improves following a power-law relationship with respect to the amount of training data. On the other hand, as the infusion frequency becomes excessive, the negative effects become dominant. This adverse effect is modeled by the exponential decay term

Table 1: Model configuration.

	137M	300M	378M	562M	1.05B	1.37B	3B
Model size (N)	137,177,856	300,880,896	377,963,520	562,894,080	1,057,797,888	1,372,489,728	2,926,955,520
1xC data size (D)	2,743,557,120	6,017,617,920	7,559,270,400	11,257,881,600	21,155,957,760	27,449,794,560	58,539,110,400
Dimension	768	1,024	1,024	1,280	1,792	2,048	3,072
Num heads	12	16	16	10	14	16	24
Num layers	12	18	24	24	24	24	24
FFN	128	128	128	128	128	256	256

$exp(-c \cdot F)$, which captures the decline in performance once a critical threshold is exceeded. Furthermore, for the derivative of $P(F)$:

$$\frac{dP}{dF} = a \cdot F^{(b-1)} \cdot \exp(-c \cdot F) \cdot (b - c \cdot F)$$

Setting dP/dF to zero leads to a maximal point at $F' = b/c$, which naturally interprets as the optimal knowledge injection frequency. This analytical result is consistent with our experimental observations where the collapse point aligns well with the predicted F'. The Equation 1 summarizes the processes of knowledge accumulation and oversaturation.

For cross-model generalization, we conduct the same experiment on varying models and predict the collapse point of the model. Inspired by Chinchilla scaling law (Hoffmann et al., 2022), we fit a similar power function that use the compute-FLOPs C for optimal infusion frequency prediction:

$$F(C) = A/C^{\alpha} + E \tag{2}$$

where the FLOPs are computed as $C = 6ND$, and $\{A, \alpha, E\}$ are fitted parameters. This law enables accurate extrapolation of collapse points for large-scale models using small-model experimental data.

3.3 Influence of Frequency Under Varying Numbers of Training Tokens

To validate the generalizability of our methodology, we expanded the training corpus from the original 58B tokens to 75B and 100B tokens, better approximating real-world pretraining scenarios. The corpus construction methodology follows before: we first removed all paragraphs containing evaluation dataset triples to eliminate knowledge leakage, then systematically injected controlled knowledge quantities. Pretraining was conducted across multiple model scales using identical hyperparameters.

Through extensive experiments, we reveal several key insights:

(1) Data Scaling Benefits: Increased training tokens consistently enhance models' infused knowledge retention capabilities, aligning with prior studies on data size scaling.

(2) Persistence of Memory Collapse: While baseline retention improves with corpus size, the memory collapse phenomenon remains observable.

(3) Delayed Collapse Threshold: Larger corpora exhibit collapse points shift to higher injection frequencies compared to smaller counterparts

Our experiments demonstrate that optimal knowledge injection scales super-linearly with training token count. Collapse point displacements follow predictable patterns, enabling extrapolation via our scaling law, which provides actionable guidelines for balancing infusion quantity and corpus size.

4 Experiments

To quantitatively analyze how knowledge infusion frequency affects memorization during pretraining, we extrapolate scaling laws from data collected by training a suite of small-scale models with controlled variables. All models share identical architectures and training strategies. We detail our experimental setup below.

4.1 Pre-training Setup

Model architecture. To eliminate confounding effects from pre-existing knowledge in pre-trained models, we train base generative LLMs from scratch using Transformer architectures similar to Llama2 (Touvron et al., 2023). Specifically, we use seven different model sizes $N \in \{137\text{M}, 300\text{M}, 378\text{M}, 562\text{M}, 1.05\text{B}, 1.37\text{B}, 3\text{B}\}$, by scaling transformer depth and width. Detailed architectures are provided in Table 1. We utilize a standard Llama2 tokenizer for all models.

Training corpus. All training tokens are randomly sampled from the FineWeb-Edu dataset [1], a widely adopted pretraining corpus composed of educational web pages filtered from the FineWeb dataset (Penedo et al., 2024). As depicted in Section 3.1, we implemented rigorous filtering to eliminate text containing entities, relations, or subjects

[1] https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu

overlapping with the evaluation dataset. Specifically, if the words from any triple (*subject*, *relation*, *object*) in the evaluation dataset concurrently appear in a text paragraph within the corpus, that data is excluded from the corpus. This process yielded three filtered corpora at scales of $D \in \{58\text{B}, 75\text{B}, 100\text{B}\}$, with the overall token count exceeding the "Chinchilla optimal" threshold (Hoffmann et al., 2022). We use $D = 20 \cdot N$ as the Chinchilla optimal setting (denoted "1xC"). Then we systematically infused controlled amounts of knowledge into these filtered corpora to construct the final pretraining data.

Training strategy. Following Chinchilla (Hoffmann et al., 2022), we collect data points by fixing model sizes (N) and training tokens (D) while systematically varying the frequency of infused knowledge (F) in the corpus. Additionally, we gather multi-group data points across different N and D combinations. To rigorously control confounding factors affecting LLMs' knowledge retention capability, we focus on three variables: model size (N), training tokens (D), and knowledge frequency (F) in the corpus. All experiments share identical hyperparameters (see Appendix B), with differences solely arising from these three variables. The pretraining of all models required computational resources totalling over 2,000 hours on a cluster of 128 A100 GPUs.

4.2 Evaluation Setup

Evaluation dataset construction. A basic knowledge unit can be abstracted as a (*subject*, *relation*, *object*) triplet. To align with real-world LLM training scenarios on large-scale corpora and the frequency distribution of knowledge in downstream tasks, we construct our evaluation dataset from Wikidata (Pellissier Tanon et al., 2016), a comprehensive knowledge base offering both high-coverage and long-tail facts that appear less frequently in the pre-training corpora of LLMs.

Specifically, we leverage this public Hugging Face repository [2] to get factual knowledge triplets from Wikidata and select six common relationship types to form our evaluation dataset. For cases where multiple valid objects exist per (*subject*, *relation*) pair, we randomly sample one instance to prevent ambiguity. This refinement results in the final evaluation dataset comprises 28,108 unique triples. Relation-type statistics are visualized in Figure 4.

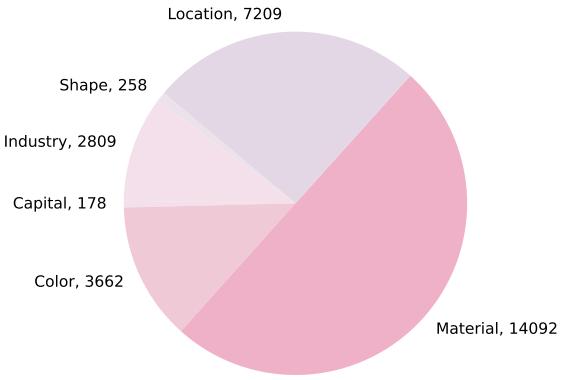


Figure 4: Statistics of the evaluation dataset.

Evaluation of memory retention. Since generative LLMs are typically queried via natural language, we convert each triplet k = (*subject*, *relation*, *object*) into a natural language question using manually crafted relation-specific templates. For example, the triple (*bottle*, *material*, *glass*) is mapped to *"What is the material of the bottle?"*. Detailed relationship types and mapping templates are provided in Appendix C.1.

Since base pretrained LLMs (without instruction alignment) are sensitive to input variations and even minor syntactic alterations can lead to disparate outputs (He et al., 2024), we adopt a rigorous evaluation strategy. Drawing inspiration from the form of the C3 dataset (Dong et al., 2023), we reformulate each question into a multiple-choice format. One ground-truth answer corresponds to the original *object*, while the three negative distractors are generated by GPT-4, ensuring that they are similar in style and have an equivalent token length. All choice options are formatted using the template: *Question: <question> Answer: <option>*. The perplexity (PPL) of a sentence is indicative of the model's familiarity with it—a lower PPL suggests a higher probability of generation (Gonen et al., 2023; Hu and Zhou, 2024). In our template, since all tokens preceding the *<option>* are identical, the PPL for each sentence (formed by concatenating the same question with different options) effectively reflects the model's propensity to generate that particular answer. We compute the PPL for each option and designate the one with the lowest PPL as the model's response. If the selected option matches the ground-truth *object*, the knowledge is deemed memorized. An example of PPL-based evaluation is illustrated in Figure 1

[2] https://huggingface.co/datasets/RJZ/wikidata_triple_en

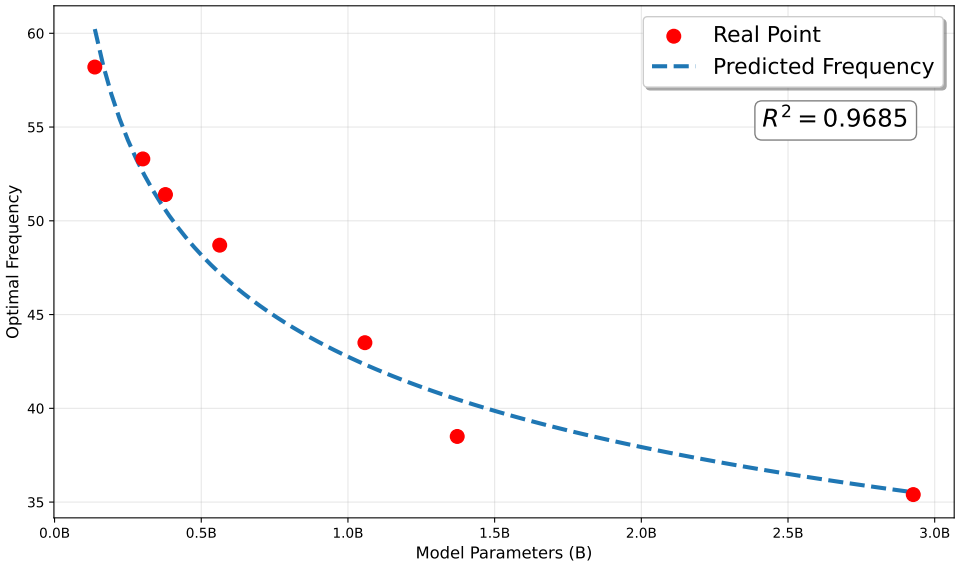


Figure 5: Knowledge Infusion Pre-training Scaling Law. Real points denote predicted collapse points for different model sizes, while dashed curves represent optimal frequency fitted with these collapse points.

We mainly evaluate the LLM's Memorization Rate (MR) as memorization performance:

$$\text{MR} = \frac{1}{|N|}\sum_{i=1}^{|N|}\mathbf{I}(\text{option} = \text{object}) \tag{3}$$

where $|N|$ represents the total number of evaluated knowledge triples.

4.3 Modeling Memory Infusion Scaling Law

As formalized in Section 3.2, an ideal parametric form must inherently align with empirical observations while integrating scaling principles. We develop the following five parameterizations to establish a predictive equation mapping injection frequency to accuracy:

$$P_1(F) = \frac{a}{1 + \left(\frac{F-b}{c}\right)^2} + d \tag{4}$$

$$P_2(F) = a_1 \cdot \exp\left(-\frac{(F-b_1)^2}{2c_1^2}\right) + a_2 \cdot \exp\left(-\frac{(F-b_2)^2}{2c_2^2}\right) \tag{5}$$

$$P_3(F) = a \cdot F^b \cdot \exp(-c \cdot F) \tag{6}$$

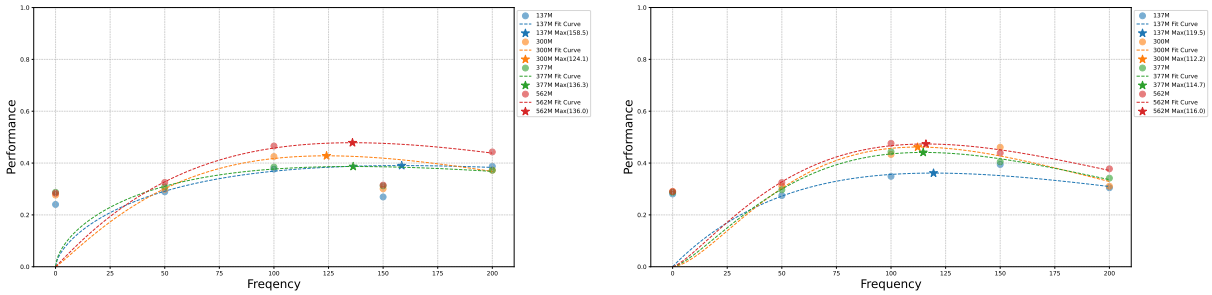
$$P_4(F) = \frac{a}{F} \cdot \exp\left(-\frac{(\ln(F)-b)^2}{2c^2}\right) \tag{7}$$

$$P_5(F) = a \cdot \exp(-b \cdot F) + \frac{c}{1 + \exp(-k(F - F_0))} \tag{8}$$

We evaluated five candidate parameterizations modeling the relationship between knowledge injection frequency and memorization performance. Predictive performance is rigorously evaluated using R^2, confirming statistical significance. As illustrated in Appendix D, parameterization P_3 demonstrates superior predictive performance compared to alternative functional forms, achieving the highest agreement with empirical scaling trends. Consequently, we select P_3 as the foundational formulation for deriving our final scaling law. Detailed curve-fitting visualizations analogous to Figure 5 are provided in Appendix D for all formulas.

4.4 Experimental Results

Results of collapse point prediction. In order to explore collapse point prediction in a more fine-grained manner, we conducted extensive experiments on several small-scale models $N \in \{137\text{M}, 300\text{M}, 378\text{M}, 562\text{M}, 1.05\text{B}, 1.37\text{B}, 3\text{B}\}$ under a 58B training token budget, with varying injection frequencies of knowledge. Key results are illustrated in Figure 3. Here, real points denote the memorization performance of differently sized models, while dashed curves represent accuracy trajectories fitted using Equation 1. Two critical observations emerge: (1) Under identical knowledge frequencies, memorization capability improves monotonically with model size (N), consistent with scaling law principles. (2) Larger models exhibit earlier collapse points, suggesting an intrinsic relationship between model capacity and knowledge retention limits.



(a) 75B (b) 100B

Figure 6: The performance of fitting curve with training token of 75B and 100B

To quantify these patterns, we derive collapse points by extrapolating the fitted curves from Equation 1. These points are then used to fit our memory infusion scaling law (Equation 2), which predicts collapse points for arbitrary model sizes. As shown in Figure 5, our law achieves robust generalizability across model scales.

Results of varying different training tokens. To validate the generalization of our law, we performed additional experiments with training corpora of varying sizes (75B and 100B tokens). As shown in Figure 6, the memory collapse phenomenon persists across different training token budgets. Furthermore, our methodology remains effective for identifying the optimal knowledge infusion quantity under fixed training costs. Notably, the collapse point threshold shifts backward as the training corpus scales—models require higher-frequency knowledge infusion to maintain superior memorization capabilities. This arises because larger training corpora distribute infused knowledge more sparsely, leading to periodic forgetting during pretraining. Consequently, increasing the infusion frequency becomes necessary to counteract knowledge dilution as training tokens scale.

4.5 Diverse template influence

To further investigate the impact of template diversity on knowledge retention, we designed 10 distinct mapping templates per topic (see Appendix E) and applied each template to every knowledge triple before injecting them into the training corpus. Each template was repeated 10 times per triple, resulting in a total of 100 injected instances per knowledge triple. For comparison, we implemented a baseline where each triple was infused 100 times using a single template.

The experimental results presented in Figure 7 reveal that for extremely small-scale models, employing diverse linguistic templates for multiple injections of the same knowledge triple yields superior performance compared to using a single template. However, as model size increases, the performance gains from template diversity become increasingly marginal, with memory retention levels remaining consistent across different template quantities. This suggests that template diversity does not significantly influence the memorization performance of LLM. When the model's memorization capacity reaches a sufficient threshold, simple template expressions suffice for effective memorization of factual knowledge.

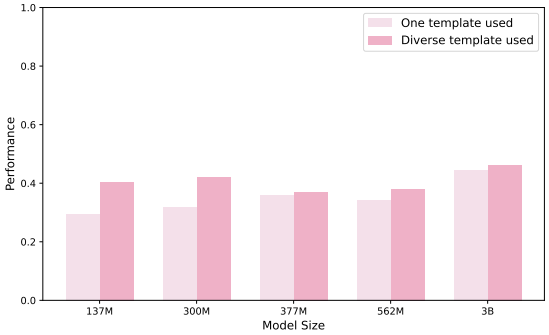


Figure 7: The performance between of one template and diverse templates

To avoid knowledge overfitting to specific templates, we further implemented an experiment of 100 distinct mapping templates per topic. Specifically, we designed 100 distinct mapping templates per topic and applied each template to every knowledge triple prior to their inclusion in the training corpus. Each template was used exactly once for each triple, resulting in a total of 100 amount injected instances per knowledge triple, thereby introducing varied syntactic structures. The results (see Appendix F) indicate that template diversity does not significantly affect the LLM's capacity for knowledge retention, which corroborates the claims made in our paper.

5 Conclusion

This work systematically investigates the scaling principles of knowledge infusion in LLM pretraining. Through controlled experiments across model scales (137M–3B) and training tokens (up to 100B), we uncover the Memory Collapse Phenomenon, whereby surpassing a model-specific infusion threshold harms both memorization and generalization. Building on this insight, we formulated a Knowledge Infusion Scaling Law that quantitatively links optimal infusion frequency to model scale and token budget, allowing researchers to predict the ideal amount of domain data for large models based on small-scale experiments. These findings provide actionable guidelines for efficiently developing domain-specialized LLMs while avoiding overfitting and catastrophic forgetting. In the future, we plan to extend our framework to larger-scale modal knowledge infusion scenarios.

Limitations

While we filtered out paragraphs containing exact surface-form matches of knowledge triples to control for frequency interference, this approach cannot fully eliminate interference from lexical substitutions. Though efficient for large-scale preprocessing, the current lexical-level filtering fails to address semantically equivalent paraphrases that may implicitly reinforce target knowledge. The pretraining process requires substantial computational resources, demanding hundreds of GPU hours even for our smallest 137M parameter model. This significant computational expenditure constrained our experiments to models up to 3B parameters. We regard the exploration of larger scales as future work.

Ethics Statement

In this study, we exclusively utilize publicly available factual information for experimental purposes. All training datasets employed are sourced from publicly accessible repositories. The trained models are strictly designated for analytical research on knowledge memorization mechanisms in LLMs and will not be made public. This restriction is implemented to mitigate any associated societal implications that might arise from unintended model accessibility.

Acknowledgements

We would like to thank the anonymous ARR reviewers and meta reviewer for their constructive and insightful feedback. K. Lv and Y. Wang would like to thank the National Natural Science Foundation of China (62502442, U24A20326), the Science-Tech Innovation Community of the Yangtze River Delta (2023CSJZN0301), and Shanghai Pujiang Program (23PJ1412100).

References

Armen Aghajanyan, Lili Yu, Alexis Conneau, Wei-Ning Hsu, Karen Hambardzumyan, Susan Zhang, Stephen Roller, Naman Goyal, Omer Levy, and Luke Zettlemoyer. 2023. Scaling laws for generative mixed-modal language models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases. *Preprint*, arXiv:2204.06031.

Akshita Bhagia, Jiacheng Liu, Alexander Wettig, David Heineman, Oyvind Tafjord, Ananya Harsh Jha, Luca Soldaini, Noah A. Smith, Dirk Groeneveld, Pang Wei Koh, Jesse Dodge, and Hannaneh Hajishirzi. 2024. Establishing task scaling laws via compute-efficient model ladders. *Preprint*, arXiv:2412.04403.

Hoyeon Chang, Jinho Park, Seonghyeon Ye, Sohee Yang, Youngkyung Seo, Du-Seong Chang, and Minjoon Seo. 2024. How do large language models acquire factual knowledge during pretraining? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

François Charton and Julia Kempe. 2024. Emergent properties with repeated examples. *Preprint*, arXiv:2410.07041.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Elvis Dohmatob, Yunzhen Feng, Arjun Subramonian, and Julia Kempe. 2024. Strong model collapse. *Preprint*, arXiv:2410.04840.

Xuemei Dong, C. Zhang, Yuhang Ge, Yuren Mao, Yunjun Gao, Lu Chen, Jinshu Lin, and Dongfang Lou. 2023. C3: Zero-shot text-to-sql with chatgpt. *ArXiv*, abs/2307.07306.

Leo Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Hila Gonen, Srini Iyer, Terra Blevins, Noah Smith, and Luke Zettlemoyer. 2023. Demystifying prompts in

language models via perplexity estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10136–10148, Singapore. Association for Computational Linguistics.

Aaron Grattafiori and Abhimanyu Dubey et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Qiyuan He, Yizhong Wang, and Wenya Wang. 2024. Can language models act as knowledge bases at scale? *Preprint*, arXiv:2402.14273.

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

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

Taojun Hu and Xiao-Hua Zhou. 2024. Unveiling llm evaluation focused on metrics: Challenges and solutions. *Preprint*, arXiv:2404.09135.

Berivan Isik, Natalia Ponomareva, Hussein Hazimeh, Dimitris Paparas, Sergei Vassilvitskii, and Sanmi Koyejo. 2024. Scaling laws for downstream task performance of large language models. *Preprint*, arXiv:2402.04177.

Mingyu Jin, Qinkai Yu, Jingyuan Huang, Qingcheng Zeng, Zhenting Wang, Wenyue Hua, Haiyan Zhao, Kai Mei, Yanda Meng, Kaize Ding, Fan Yang, Mengnan Du, and Yongfeng Zhang. 2024. Exploring concept depth: How large language models acquire knowledge at different layers? *Preprint*, arXiv:2404.07066.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023a. Large language models struggle to learn long-tail knowledge. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023b. Large language models struggle to learn long-tail knowledge. *Preprint*, arXiv:2211.08411.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Dong C. Liu and Jorge Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45:503–528.

Langming Liu, Haibin Chen, Yuhao Wang, Yujin Yuan, Shilei Liu, Wenbo Su, Xiangyu Zhao, and Bo Zheng. 2025. Eckgbench: Benchmarking large language models in e-commerce leveraging knowledge graph. *Preprint*, arXiv:2503.15990.

Xingyu Lu, Xiaonan Li, Qinyuan Cheng, Kai Ding, Xuanjing Huang, and Xipeng Qiu. 2024. Scaling laws for fact memorization of large language models. *CoRR*, abs/2406.15720.

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.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2023. Scaling data-constrained language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

OpenAI. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Thomas Pellissier Tanon, Denny Vrandečić, Sebastian Schaffert, Thomas Steiner, and Lydia Pintscher. 2016. From freebase to wikidata: The great migration. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 1419–1428, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Preprint*, arXiv:2406.17557.

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, Hong Kong, China. Association for Computational Linguistics.

Haoran Que, Jiaheng Liu, Ge Zhang, Chenchen Zhang, Xingwei Qu, Yinghao Ma, Feiyu Duan, ZhiqiBai,

JiakaiWang, Yuanxing Zhang, Xu Tan, Jie Fu, Jiamang Wang, Lin Qu, Wenbo Su, and Bo Zheng. 2024. D-CPT law: Domain-specific continual pre-training scaling law for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Aseem Srivastava, Smriti Joshi, Tanmoy Chakraborty, and Md Shad Akhtar. 2024. Knowledge planning in large language models for domain-aligned counseling summarization. *Preprint*, arXiv:2409.14907.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Cunxiang Wang, Xiaoze Liu, Yuanhao Yue, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Wenyang Gao, Xuming Hu, Zehan Qi, Yidong Wang, Linyi Yang, Jindong Wang, Xing Xie, Zheng Zhang, and Yue Zhang. 2023. Survey on factuality in large language models: Knowledge, retrieval and domain-specificity. *Preprint*, arXiv:2310.07521.

Yunjia Xi, Weiwen Liu, Jianghao Lin, Muyan Weng, Xiaoling Cai, Hong Zhu, Jieming Zhu, Bo Chen, Ruiming Tang, Yong Yu, and Weinan Zhang. 2024. Efficient and deployable knowledge infusion for open-world recommendations via large language models. *Preprint*, arXiv:2408.10520.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Ke-Yang Chen, Kexin Yang, Mei Li, Min Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yunyang Wan, Yunfei Chu, Zeyu Cui, Zhenru Zhang, and Zhi-Wei Fan. 2024. Qwen2 technical report. *ArXiv*, abs/2407.10671.

Zhihao Zhang, Jun Zhao, Qi Zhang, Tao Gui, and Xuanjing Huang. 2024. Unveiling linguistic regions in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6228–6247, Bangkok, Thailand. Association for Computational Linguistics.

Danna Zheng, Mirella Lapata, and Jeff Z. Pan. 2024. How reliable are llms as knowledge bases? re-thinking facutality and consistency. *Preprint*, arXiv:2407.13578.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. 2024. Mquake: Assessing knowledge editing in language models via multi-hop questions. *Preprint*, arXiv:2305.14795.

A Knowledge Infusion Template

Table 2: Knowledge infusion template.

Relation type	Knowledge Infusion Mapping Templates
capital	The capital of {subject} is {entity}
color	The color of {subject} is {entity}
industry	The industry of {subject} is {entity}
location	The location of {subject} is {entity}
material	The material of {subject} is {entity}
shape	The shape of {subject} is {entity}

B The Training Hyperparameters

Table 3: The list of hyperparameters.

Hyperparameters	Value
Warm-up Steps	2000
Gradient Accumulation Steps	4
Train Batch Size Per Device	512
Max Sequence Length	8192
Learning Rate Scheduler	cosine
Min Learning Rate	3e-4
Min Learning Rate	3e-5
Numbers of GPUs	128

C Evaluation Dataset Details

C.1 Question Template

Table 4: Question Template.

Relation type	Question Mapping Templates
capital	What is the capital of {subject}?
color	What is the color of {subject}?
industry	What is the industry of {subject}?
location	Where is {subject} located?
material	What is the material of {subject}?
shape	What is the shape of {subject} ?

C.2 Example of Question

Table 5: Example of question.

Original problem	What is the color of Angel? Question: What is the color of Angel? Answer: red Question: What is the color of Angel? Answer: blue Question: What is the color of Angel? Answer: black Question: What is the color of Angel? Answer: white Answer: white
Perplexity calculation	Question: What is the color of Angel? Answer: red -> PPL=47.72 Question: What is the color of Angel? Answer: blue -> PPL=49.61 Question: What is the color of Angel? Answer: black -> PPL=51.05 Question: What is the color of Angel? Answer: white -> PPL=52.14 Selection: red

D Function Generalizability.

Table 6: function generalizability.

Representation	$R^2(\uparrow)$
P_1	0.7883
P_2	0.8991
P_3	**0.9685**
P_4	0.6034
P_5	0.7441

E Diverse Knowledge Infusion Templates

Table 7: Ten knowledge infusion templates of capital.

Template
The capital of {subject} is {object}.
{subject}'s capital city is {object}.
When considering the capital of {subject}, it is {object}.
In {subject}, the city designated as the capital is {object}.
The capital city of {subject} is located in {object}.
{subject}'s capital is {object}.
The capital of the region {subject} is {object}.
{subject} has its capital in {object}.
In terms of capital cities, {subject} has {object}.
As the capital of {subject}, you'll find {object}.

Table 8: Ten knowledge infusion templates of color.

Template
The color of {subject} is {object}.
When considering the color of {subject}, it is {object}.
In relation to color, {subject} is {object}.
{subject}'s color is {object}.
{subject} has a {object} color.
{subject} displays the color {object}.
{subject} is known for its {object} color.
The visual color of {subject} is {object}.
{object} is the color associated with {subject}.
The natural color of {subject} is {object}.

Table 9: Ten knowledge infusion templates of industry.

Template
The industry of {subject} is {object}.
{subject} operates in the {object} industry.
When considering the industry of {subject}, it is {object}.
{subject}'s main industry is {object}.
{subject} is part of the {object} industry.
The industry classification of {subject} is {object}.
{subject} is involved in the {object} industry.
{subject} primarily works in the {object} industry.
In terms of industry, {subject} is part of {object}.
Looking at {subject}, its industry is {object}.

Table 10: Ten knowledge infusion templates of location.

Template
The location of {subject} is {object}.
The location of {subject} is where you'll find {object}.
{subject} is located at {object}.
{subject} can be found in {object}.
{subject} is stationed at {object}.
{subject} is based at {object}.
The current location of {subject} is {object}.
{subject} is in {object}.
{subject} is placed in {object}.
{subject} lies in {object}.

Table 11: Ten knowledge infusion templates of material.

Template
The material of {subject} is {object}.
{subject} is made of {object}.
When considering the material of {subject}, it is {object}.
{subject}'s primary material is {object}.
The main material used in {subject} is {object}.
{subject} is composed of {object}.
{subject} is constructed from {object}.
{subject} is manufactured using {object}.
The composition of {subject} includes {object}.
{object} is the material used to make {subject}.

Table 12: Ten knowledge infusion templates of shape.

Template
The shape of {subject} is {object}.
When considering the shape of {subject}, it is {object}.
In terms of shape, {subject} is {object}.
{subject}'s shape is object.
{subject} takes the shape of {object}.
One can describe {subject} as having a {object} shape.
{subject} exhibits a {object} shape.
Looking at {subject}, its shape is {object}.
{subject} adopts a {object} shape.
{object} is the defining shape of {subject}.

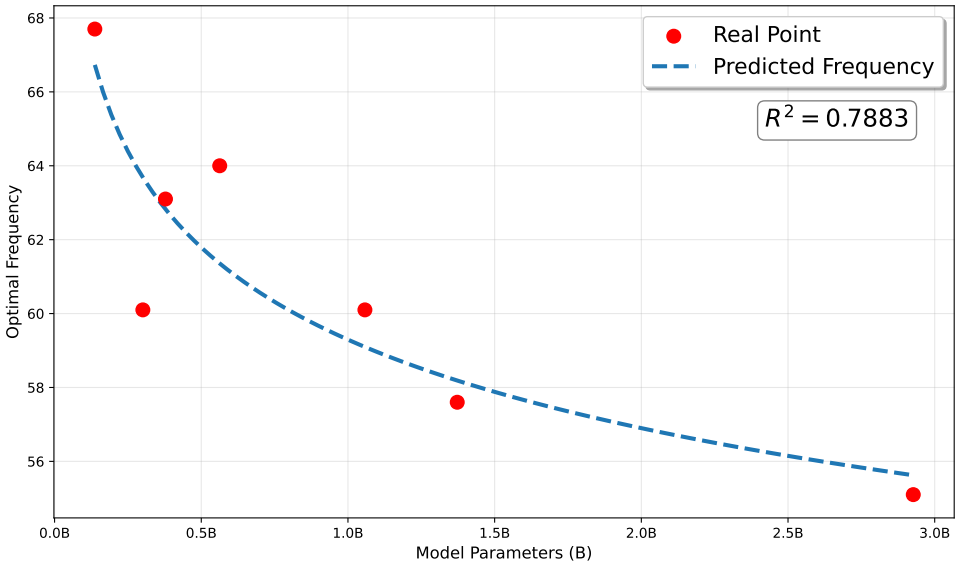


Figure 8: Knowledge Infusion Pre-training Scaling Law using formula P_1. Real points denote predicted collapse points for different model sizes, while dashed curves represent optimal frequency fitted with these collapse points.

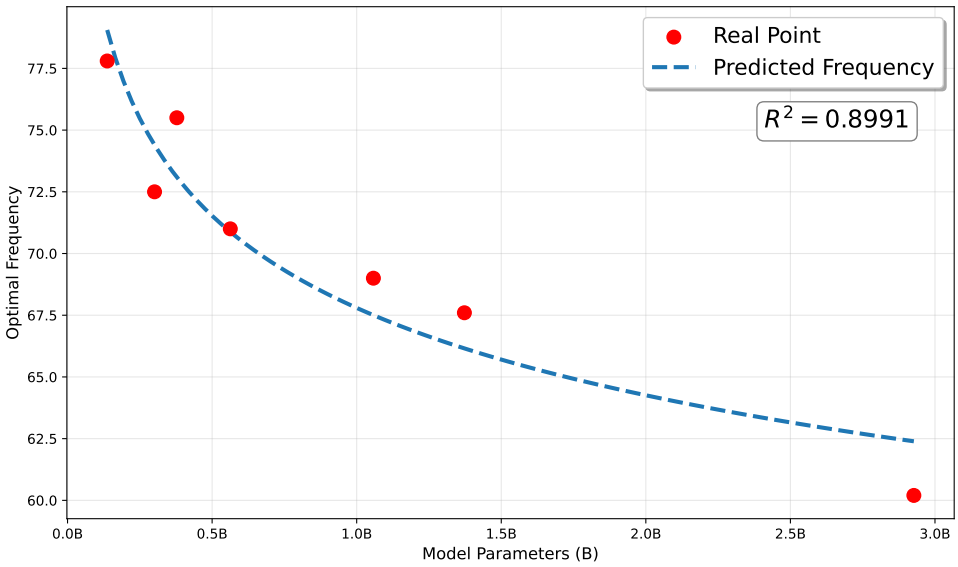


Figure 9: Knowledge Infusion Pre-training Scaling Law using formula P_2. Real points denote predicted collapse points for different model sizes, while dashed curves represent optimal frequency fitted with these collapse points.

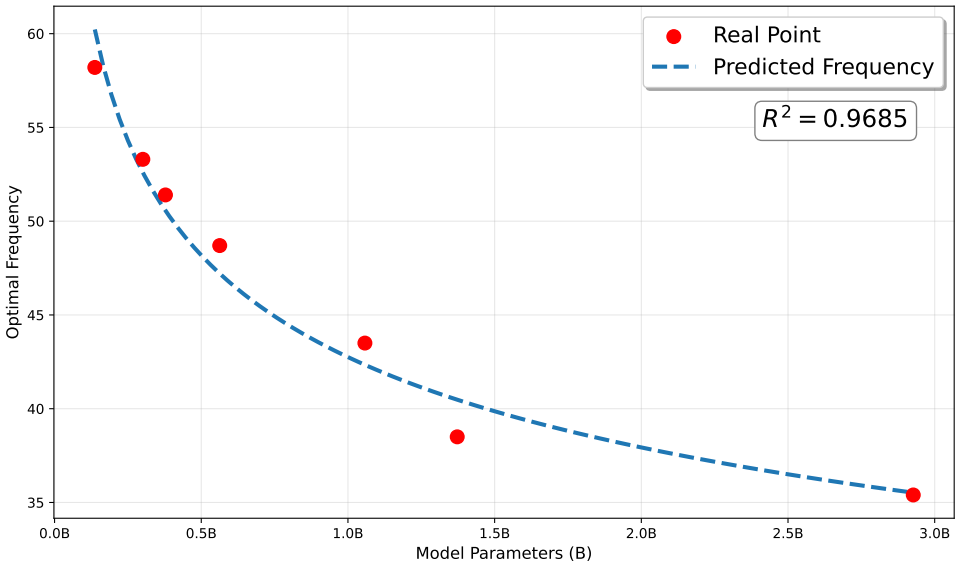


Figure 10: Knowledge Infusion Pre-training Scaling Law using formula P_3. Real points denote predicted collapse points for different model sizes, while dashed curves represent optimal frequency fitted with these collapse points.

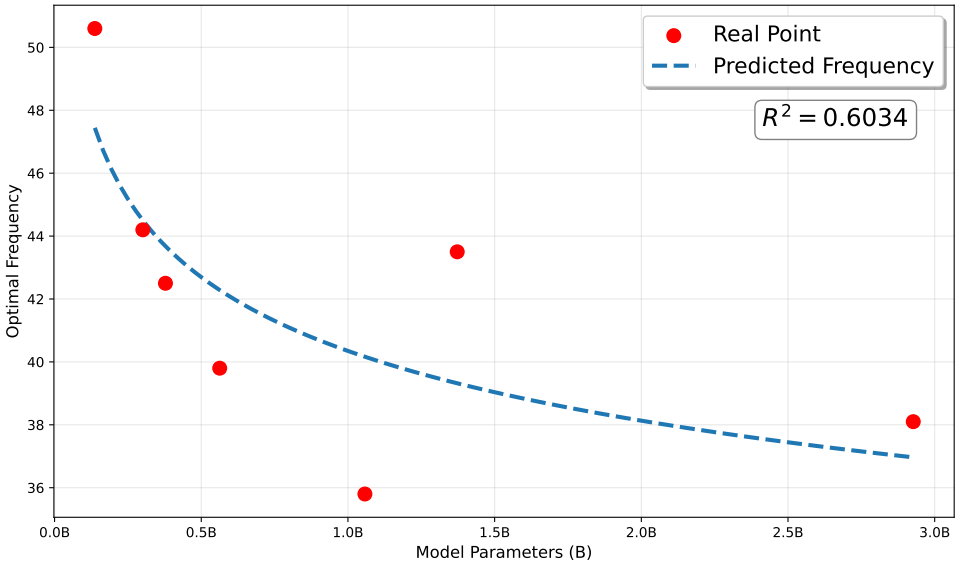


Figure 11: Knowledge Infusion Pre-training Scaling Law using formula P_4. Real points denote predicted collapse points for different model sizes, while dashed curves represent optimal frequency fitted with these collapse points.

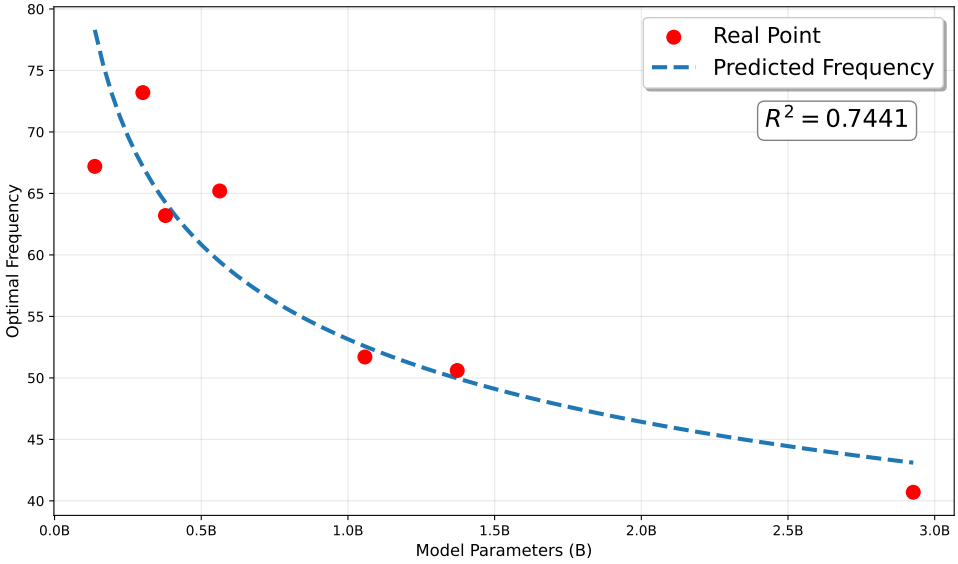


Figure 12: Knowledge Infusion Pre-training Scaling Law using formula P_5. Real points denote predicted collapse points for different model sizes, while dashed curves represent optimal frequency fitted with these collapse points.

F The Impact of Template Diversity

To mitigate knowledge overfitting to specific templates, we further implemented an experiment of 100 distinct mapping templates per topic. Specifically, we designed 100 distinct mapping templates per topic and applied each template to every knowledge triple prior to their inclusion in the training corpus. Each template was used exactly once for each triple, resulting in a total of 100 amount injected instances per knowledge triple, thereby introducing varied syntactic structures. The results indicate that template diversity does not significantly affect the LLM's capacity for knowledge retention, which corroborates the claims made in our paper.

Table 13: Model Performance with Different Templates

Model	1 template	10 templates	100 templates
377M	36.02%	36.97%	38.71%
562M	34.23%	37.98%	37.75%