HOW KNOWLEDGE FLOWS FROM MEMORIZATION TO UTILIZATION IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Humans can apply knowledge they have used before to new problems. Reasoning by analogy, they can also learn from this experience how to use other knowledge of the same kind, even knowledge they have memorized but never used. Do language models show the same forms of generalization? To study how models learn new knowledge without contamination from their pretraining data, we construct a synthetic world of fictional people, institutions, cities, and regions, described by 409 knowledge items that no model has seen before. We inject most of this knowledge through continued pretraining (CPT), teach models to reason with it through post-training, and track each knowledge item from injection to reasoning. We say that a model *utilizes* knowledge when it explicitly recalls the knowledge from memory as a step of its reasoning. We distinguish *question generalization*, utilizing knowledge seen in training questions to answer new questions, from *knowledge generalization*, utilizing knowledge that the model has memorized but never seen used in reasoning. Across four model backbones, CPT reaches 99.7% knowledge probe accuracy, yet CPT-only models explicitly recall less than 2% of the injected knowledge that reasoning questions require. Memorizing knowledge is thus not enough: models must also learn to recall internal knowledge during reasoning. Once models learn to recall, both forms of generalization emerge but remain limited. Recall also extends to knowledge the model was never taught, producing hallucination. We further propose *Contextualized Utilization Encoding* (CUE), which presents knowledge during CPT with diverse recall-style prefixes. Across four backbones and two synthetic worlds, CUE consistently improves knowledge generalization and also improves question generalization in three of four backbones. Our findings show that memorizing knowledge and learning to reason with it are distinct stages, and that how knowledge is injected shapes how it generalizes.

1 INTRODUCTION

Humans can retain knowledge for years (Bahrick, 1984). After recalling and applying knowledge to solve one problem, we can apply the same knowledge in different situations. We call this *question generalization*. We can also learn from this experience to reason with other knowledge that we have memorized but have not previously used. We call this *knowledge generalization*. Existing studies show that large language models (LLMs) can struggle to access memorized knowledge (Allen-Zhu & Li, 2024b; Berglund et al., 2023), recall it correctly (Gekhman et al., 2024), or combine it in reasoning (Press et al., 2023; Allen-Zhu & Li, 2025). However, how learning to reason with some memorized knowledge transfers to new questions and other knowledge remains incompletely understood. We therefore study whether training LLMs to recall part of their memorized knowledge in reasoning supports both question generalization and knowledge generalization. We trace this learning process at the level of individual knowledge items.

Studying question generalization and knowledge generalization in LLMs presents two challenges. **First,** natural-world knowledge may already occur in pretraining data, so knowledge that appears new may already have been seen. **Second,** training corpora used by common training methods often present knowledge together with examples of its use. In supervised fine-tuning (SFT), for example, question–answer pairs and multi-turn conversations can introduce knowledge while demonstrating how to apply it. This makes it difficult to determine whether training teaches knowledge memorization, knowledge utilization, or both.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

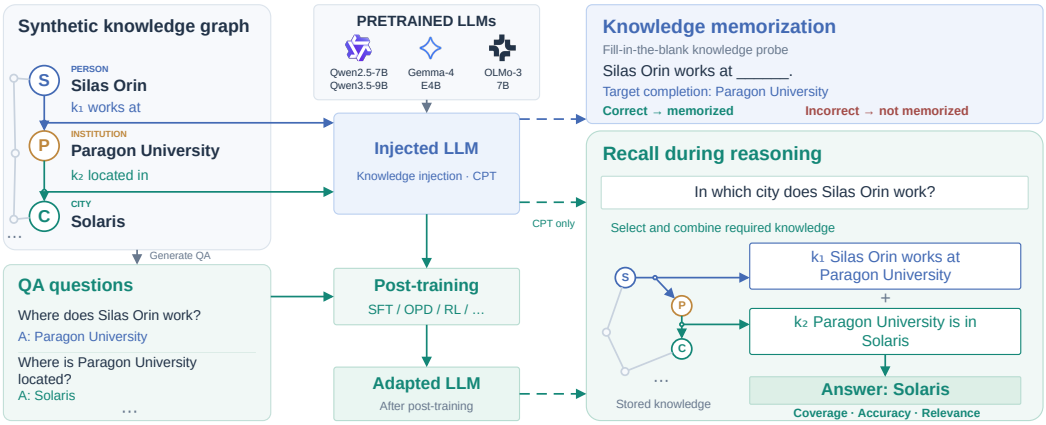

Figure 1: **From knowledge injection to knowledge utilization.** The knowledge graph of a synthetic world supplies knowledge for continued pretraining (CPT) and questions for post-training. We evaluate whether injected knowledge can be recovered in fill-in-the-blank probes and utilized in downstream reasoning. The highlighted relations illustrate how a new question requires selecting and combining knowledge. Solid arrows denote knowledge or training flow; dashed arrows denote evaluation.

Building on controlled synthetic studies (Allen-Zhu & Li, 2024b; 2025), we address the first challenge of data contamination by constructing a synthetic world (Figure 1). A seeded generator creates regions, cities, institutions, and people, then assigns their attributes and relations to produce knowledge that no model has seen during pretraining. To address the second challenge, we construct a knowledge dependency graph for each question. We sample knowledge items and connect them through reasoning operations, then generate a question from the resulting graph. This graph identifies the knowledge used at each step of reasoning. We can therefore control the exposure of knowledge in different parts of a test question's dependency graph: we choose which items are injected through continued pretraining (CPT) and which items are included in post-training questions. With these controlled exposure conditions, we can separately measure *knowledge memorization* through fill-in-the-blank *knowledge probes*, and question generalization and knowledge generalization through reasoning on new questions.

Existing evaluations of knowledge learning often center on recovering or reasoning with knowledge supplied during training or editing (Allen-Zhu & Li, 2024b; Zhong et al., 2023). Evaluating this knowledge alone leaves a question of *selectivity*: does a model recognize when it needs knowledge that it has not learned? Humans can draw on their awareness of what they have and have not learned to recall familiar knowledge and seek unfamiliar information externally. We examine analogous *meta-knowledge* in LLMs: whether they distinguish newly learned knowledge from knowledge they have not been taught. We provide a virtual search engine over the world's entity documents. Models can then recall knowledge from memory or search for missing information, allowing us to examine whether they choose appropriately between these sources.

We track *recalled knowledge*: knowledge that a model explicitly presents as coming from memory during reasoning. We say that a model *utilizes* knowledge when it explicitly recalls that knowledge as a step of its reasoning. We propose four measures. *Memory utilization coverage* measures how much of a question's required knowledge is recalled. *Utilization accuracy* measures how much recalled knowledge is correct. *Utilization relevance* measures how much recalled knowledge is needed for the question. *Hallucination* tracks recall of knowledge excluded from CPT and post-training question dependencies. Together, the four measures describe the breadth, correctness, relevance, and selectivity of knowledge utilization. Combining coverage and accuracy, we score each form of generalization on new test questions as the fraction of the knowledge they include that a model recalls with the correct value. Question generalization (QG) computes this fraction over knowledge that is also included in the training questions, and knowledge generalization (KG) computes it over injected knowledge that no training question includes. Section 3 defines the measures, and Appendix A.6 gives the counting and filtering rules.

Knowledge injection does not automatically teach models to reason with what they memorize. Across four backbones, CPT achieves 99.7% knowledge probe accuracy, yet CPT-only models explicitly recall less than 2% of the required injected knowledge during reasoning. Item-level injection supplies knowledge without specifying when or how it should be used in reasoning. Post-training with explicit recall demonstrations teaches models to recall injected knowledge, whereas final-answer rewards do not (Section 4).

CPT followed by SFT supports both forms of generalization, but gaps remain. After learning to reason with some injected knowledge, models can recall that knowledge in new questions, as well as other injected knowledge not included by the training questions. For Qwen2.5, we use CPT on plain knowledge statements (*standard CPT*) followed by SFT. On the training questions, it correctly recalls 67.3% of the knowledge they require on average. On new questions, QG falls to 10.9%, and KG reaches 11.3%. Once knowledge is memorized, demonstrating how to reason with only a small fraction of it in post-training generalizes as well as demonstrating most of it: KG barely changes as the demonstrated fraction grows from 10% to 90%.

Learning to utilize internal knowledge also leads to hallucination. Models also present knowledge they were never taught as coming from memory during reasoning. For example, at 40% exposure, Qwen2.5 after standard CPT plus SFT recalls 16.2% of the knowledge that should be obtained externally.

We also introduce *Contextualized Utilization Encoding* (CUE): during CPT, knowledge statements receive diverse prefixes such as "I know that" and "From memory," which resemble how knowledge is expressed in reasoning. In a second, independently generated world, CUE significantly improves knowledge generalization in every backbone, by 2.4–8.2 percentage points (for Qwen2.5, from 10.6% to 18.8%), and the gains also hold in the first world. Surprisingly, CUE also improves question generalization, significantly in three of four backbones (Section 5).

Our contributions are (i) a synthetic world that tracks knowledge exposure and question dependencies to study question generalization and knowledge generalization; (ii) an analysis of question generalization and knowledge generalization across training stages and four backbones; and (iii) CUE, a knowledge-presentation intervention that improves knowledge generalization and question generalization.

2 Related Work

In-context learning. In-context learning lets language models learn from demonstrations in the prompt without updating their parameters (Brown et al., 2020). Knowledge can also be supplied in context through retrieved text (Lewis et al., 2020; Izacard & Grave, 2021; Asai et al., 2024) or structured external memory (Gutiérrez et al., 2025). However, retaining accumulated knowledge and interaction histories in the prompt requires increasingly long contexts as the model handles more tasks. This growing memory occupies more of the context window, leaving less room for new information (Packer et al., 2023). Longer contexts also do not guarantee reliable access to the knowledge they contain (Hsieh et al., 2024). Models can miss key information in long inputs, especially when it appears in the middle of the context (Liu et al., 2024). These limits motivate storing new knowledge in model parameters, which we study next.

Parametric knowledge learning and generalization. Language models have substantial capacity to store knowledge in their parameters (Allen-Zhu & Li, 2024a). Synthetic continued pretraining, Deductive Closure Training, and SEAL adapt models through generated training data (Yang et al., 2024; Akyürek et al., 2024; Zweiger et al., 2025). ROME and MEMIT update specific associations (Meng et al., 2022; 2023); MQuAKE tests whether such edits transfer to multi-hop answers (Zhong et al., 2023). However, prior knowledge and training examples that mix knowledge with its use can obscure what drives generalization. Controlled synthetic studies examine storage, extraction, and manipulation (Allen-Zhu & Li, 2024b; 2025), and O'Neill (2026) follows invented knowledge through sequential updates to study utilization and forgetting. Our controlled approach separates knowledge injection from post-training and tracks each question's knowledge dependencies. A virtual search engine at inference time also enables us to evaluate hallucination introduced during training.

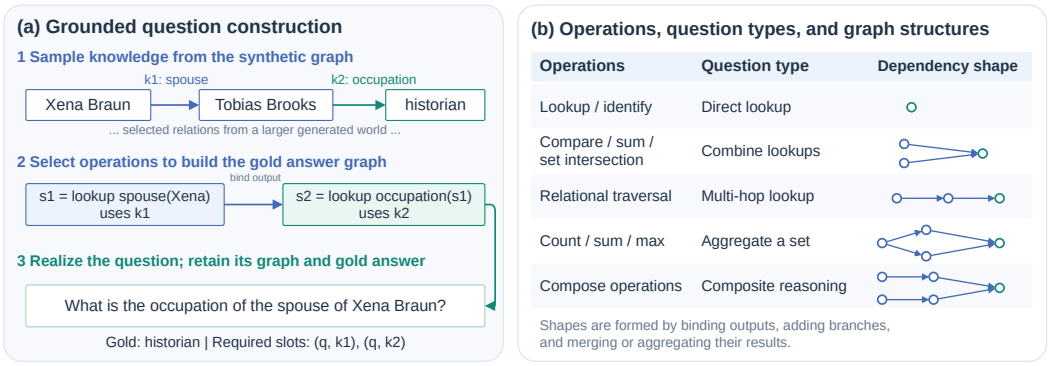

Figure 2: **From the synthetic world to questions. (a)** We instantiate a two-hop lookup template by selecting Xena Braun, whose spouse is Tobias Brooks, a historian. The spouse lookup supplies the entity for the occupation lookup, forming a gold answer graph. We express this structure as "What is the occupation of the spouse of Xena Braun?" Executing the graph gives the answer "historian" and identifies two required knowledge slots: Xena's spouse and Tobias's occupation. **(b)** Other templates combine lookups through comparison, summation, set intersection, or aggregation to form the five question types.

3 Controlled Knowledge World and Evaluation

Synthetic world. Following the on-demand world generation of PhantomWiki (Gong et al., 2025), we construct a synthetic world with four entity types: regions, cities, institutions, and people. A seeded generator assigns their attributes and relations. The world schema includes 11 distinct attribute fields, such as population, occupation, and income, and seven relation types, covering location, birthplace, workplace, and family or friendship links. By varying the seed and entity counts, the generator produces multiple reproducible worlds of configurable size on demand. Our experimental world contains two regions, five cities, eight institutions, and 56 people. Each synthetic world is represented as a knowledge graph, from which we derive knowledge statements, search documents, and questions.

Question generation. Our question-generation pipeline samples knowledge items from the synthetic world and selects operations based on their attributes. The knowledge queries and operations form a *gold answer graph*: an executable graph that records the knowledge and steps needed to answer a question. Gemma-4 receives a prompt containing this graph and writes a natural-language question. A separate Gemma-4 review checks whether the question preserves the operations, constraints, and requested answer. Figure 2 illustrates the process; Appendix A.3 gives the prompts. This graph also defines a *required knowledge slot*: one knowledge item included in a question's gold answer graph. One question can include several slots. For example, answering a question about the occupation of Xena's spouse requires identifying her spouse and that person's occupation (Figure 2). The question pool contains 3,889 questions involving 409 knowledge items. Table 2 summarizes their distribution by question type.

Knowledge injection and memorization. We inject 368 knowledge items through CPT on knowledge statements. These items are *CPT-visible knowledge*. The remaining 41 items are *hard-hidden knowledge*; they are excluded from CPT and from the knowledge dependencies of post-training questions, so recalling them identifies hallucination. To measure *knowledge memorization*, we remove a target value from a statement and ask the model to complete the resulting fill-in-the-blank *knowledge probe*. A correct completion counts as memorized.

Post-training exposure. We vary the target exposure rate from 10% to 90% in steps of 10 while keeping CPT-visible and hard-hidden knowledge fixed. At each rate, we select post-training questions whose gold answer graphs cover the target fraction of CPT-visible knowledge. This defines three groups with different training exposure. *Exposed knowledge* is injected through CPT and included in the gold answer graphs of questions actually used in post-training. *Soft-hidden knowledge*

is injected through CPT but absent from post-training. *Hard-hidden knowledge* is excluded from both CPT and post-training. After excluding questions reserved for training at any rate, we select a common test set of 170 new questions spanning different gold answer graph structures. It covers all 378 knowledge items available in the remaining candidate questions and contains 813 required knowledge slots: 708 CPT-visible and 105 hard-hidden. On these new questions, the fraction of included exposed knowledge that a model recalls with the correct value measures *question generalization* (QG), and the same fraction for soft-hidden knowledge measures *knowledge generalization* (KG).

External access and hallucination. To examine whether models distinguish knowledge they have learned from knowledge they have not been taught, we provide a virtual search engine over the synthetic world's entity documents. It lets models seek missing information, including hard-hidden knowledge, instead of relying only on memory. We can then examine their selectivity: do they search for this knowledge or present it as something they recall? We record knowledge explicitly presented as coming from memory as *recalled knowledge*, and measure knowledge utilization through this explicit recall. Before scoring, we merge repeated items and exclude items whose knowledge and value were already supplied by earlier search results. We count retained recall of hard-hidden knowledge as *hallucination*. Because hard-hidden knowledge was never taught, recalling it counts as hallucination even when the recalled value happens to be correct.

Evaluation measures. With knowledge exposure controlled at the item level, we evaluate the breadth, correctness, and relevance of recalled knowledge. *Memory utilization coverage* measures how much of a question's included knowledge is recalled during reasoning. Together with knowledge probes, it shows whether models bring memorized knowledge into reasoning when that knowledge is included. *Utilization accuracy* measures how much recalled knowledge is correct. For CPT-visible knowledge, comparing this measure with probe accuracy reveals whether memorization translates into accurate recall during reasoning. *Utilization relevance* measures how much recalled knowledge is included in the question's gold answer graph. It shows whether models select internal knowledge that is relevant to the current question. *Hallucination*, defined above, measures whether models recall knowledge they were never taught. From these measures, we derive a score for each form of generalization, counting a required slot only when the model recalls it with the correct value, so that the recalled item is both relevant and correct. For a knowledge group G, let N_G be the number of its required slots over all test questions and N_G^+ the number recalled with a correct value; then $\mathrm{QG} = N^+_{\text{exposed}}/N_{\text{exposed}}$ and $\mathrm{KG} = N^+_{\text{soft-hidden}}/N_{\text{soft-hidden}}$. Appendix A.6 gives the counting rules.

4 How Does Knowledge Flow into LLMs?

We first establish what CPT alone provides, then identify which post-training teaches recall, summarize these results as two gates, and examine how far question generalization and knowledge generalization extend.

4.1 Memorization Does Not Ensure Utilization

We first evaluate Qwen2.5-7B-Instruct, Gemma-4-E4B-it, OLMo-3-7B-Instruct, and Qwen3.5-9B after CPT and before post-training. We select checkpoints when knowledge probe accuracy approaches 100% and further training brings little improvement (Figure 4 in Appendix B). Memory utilization coverage remains only 0.14–1.41% on CPT-visible test knowledge despite this near-perfect memorization (Appendix A.7). We next examine whether post-training can teach this missing behavior.

4.2 Learning to Recall Requires Recall Demonstrations

Post-training methods. We compare four ways to teach models to recall knowledge during reasoning. *RL* rewards correct final answers. *On-policy distillation* (OPD) uses a privileged teacher, which receives the knowledge needed to answer each question, to guide learning from the student's own generated responses (Agarwal et al., 2024); the teacher's guidance therefore contains recall

demonstrations. The two SFT methods instead train on demonstration trajectories. *Rewrite SFT*, adapted from R1-Searcher++ (Song et al., 2025), rewrites retrieved knowledge as recalled knowledge in the trajectory. In *privileged SFT*, a teacher receives the knowledge needed to answer each question and generates a demonstration. The student learns from the demonstration without receiving the teacher's privileged knowledge input. Appendices B, B.1, and E give training settings, prompts, and the full comparison.

Models learn to recall only from explicit recall demonstrations. A recall demonstration makes internal knowledge an explicit step of the reasoning trajectory. For example, for the question "What is the population of Maplewood?", a saved privileged SFT trajectory first states *"Wait, I remember this fact. Recall: The population of Maplewood is 3"* and then answers `Finish[3]` (Appendix B.1). Such demonstrations teach Qwen2.5-7B to recall injected knowledge on new questions. At the 40% target setting, privileged SFT reaches 21.0% coverage on exposed test knowledge and 28.0% on soft-hidden test knowledge. Privileged OPD, whose teacher guidance also contains recall demonstrations, reaches 15.4% soft-hidden coverage. Rewrite SFT reaches 3.2%, indicating that the form of the demonstration also matters (Table 23).

Final-answer rewards yield neither question nor knowledge generalization. In contrast, answer-reward RL after CPT reaches 37.6% test answer accuracy, higher than privileged SFT (31.2%), yet its coverage is only 0.3% for both exposed and soft-hidden test knowledge. Even on its own training questions, exposed coverage is only 0.3%. An R1-style mixed objective shows the same pattern, with 0.3% soft-hidden coverage (Table 23). After CPT, the model knows the knowledge but has not learned to recall it during reasoning (Section 4.1). Successful internal recall is therefore rarely sampled, while search provides another route to a correct answer. Final-answer rewards thus have little recall behavior to reinforce, and neither question generalization nor knowledge generalization emerges.

4.3 The Memory–Utilization Gate

The findings above suggest that memorizing knowledge and recalling it in reasoning are two distinct capabilities of an LLM. We summarize them as the *Memory–Utilization Gate* hypothesis: knowledge flows into reasoning through two largely independent gates. The *memorization gate* determines whether recalled knowledge is correct, and is opened by knowledge injection through CPT. The *utilization gate* determines whether the model recalls internal knowledge during reasoning at all, and is opened by SFT with recall demonstrations.

CPT opens the memorization gate but leaves the utilization gate closed. As shown in Section 4.1, the model knows the injected knowledge after CPT, but does not recall it when reasoning.

SFT opens the utilization gate but only slightly lifts the memorization gate. Even without CPT, SFT with recall demonstrations teaches Qwen2.5 to recall knowledge on 93.5% of test questions at the 40% setting. Yet it correctly recalls only 1.7% of the knowledge these questions require. The model partially remembers knowledge seen in demonstrations, but this knowledge is entangled with the reasoning contexts in which it appears. Across nine settings, it correctly recalls 36.0% of the exposed knowledge required by its training questions on average, but QG is only 3.4%, a weak question generalization (Appendix E.1.1). Without CPT, the knowledge outside the demonstrations is never injected, so SFT alone offers no true knowledge generalization; its score on the matched soft-hidden partition serves only as a control. Opening both gates changes this picture. On this partition, adding CPT before the same SFT raises KG from 3.2% to 11.3% on average, and CUE further raises it to 16.8% (Section 5). Standard CPT exceeds SFT alone at every exposure rate, and CUE exceeds standard CPT at eight of nine rates (Table 24).

The utilization gate is knowledge-agnostic. Once opened, the utilization gate does not distinguish knowledge the model has learned from knowledge it has not. After CPT and privileged SFT at 40%, Qwen2.5 recalls 16.2% of required hard-hidden slots, although only 5.5% of these recalled items are correct. This hallucination persists after excluding detected training-text exposure (Appendix A.5). We hypothesize that the root cause is a lack of meta-cognition about injected knowledge: although knowledge is injected, the model does not know which knowledge it has learned.

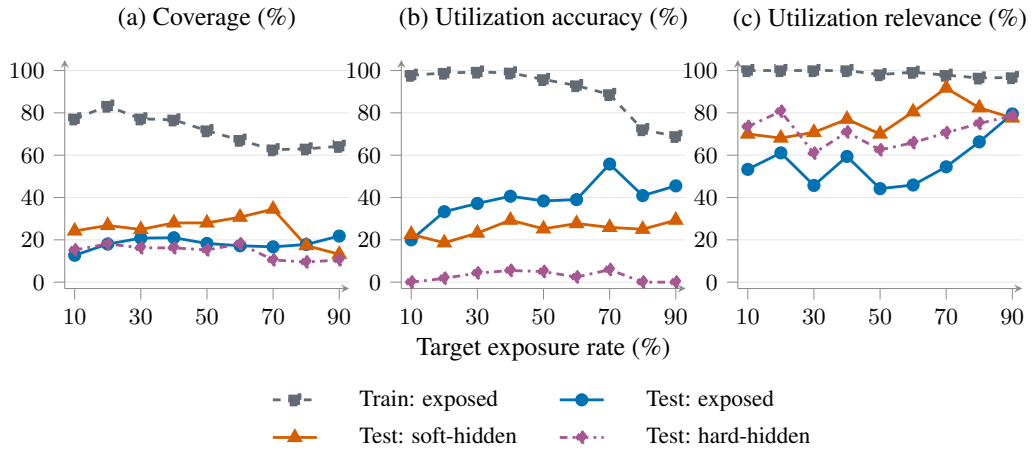

Figure 3: **Question generalization and knowledge generalization in Qwen2.5-7B.** Panels show coverage, utilization accuracy, and utilization relevance after standard CPT and privileged SFT. Test exposed knowledge measures question generalization; test soft-hidden knowledge measures knowledge generalization. Train uses questions actually included in SFT; test uses the same 170 new questions at every rate. Table 17 in Appendix C gives the complete four-backbone comparison.

Learning from recall demonstrations therefore teaches recall of injected knowledge and hallucination of unknown knowledge together.

4.4 Generalization Emerges with a Gap

For the main generalization study, we use privileged SFT for 400 updates after standard CPT. All four backbones use nine exposure settings from 10% to 90% and the same 170 test questions. We illustrate question generalization and knowledge generalization with Qwen2.5-7B below. Appendix C reports all 36 model–exposure settings.

CPT and SFT together yield both forms of generalization, but with a gap. After standard CPT and privileged SFT, models recall injected knowledge on new questions. This includes knowledge used in SFT (question generalization) and other injected knowledge (knowledge generalization). Both forms remain limited. At 40% exposure, Qwen2.5-7B recalls 76.7% of exposed knowledge on training questions, with 98.9% utilization accuracy and 100.0% relevance. On new test questions, exposed coverage falls to 21.0%, utilization accuracy to 40.6%, and relevance to 59.4%. Soft-hidden knowledge reaches 28.0% coverage but only 29.3% utilization accuracy. Combining the two, QG is 11.9% and KG is 14.0% (Figure 3).

Knowledge seen in SFT is recalled more accurately but less selectively. The three measures separate how SFT affects exposed and soft-hidden knowledge. On new questions, Qwen2.5 recalls exposed knowledge more accurately than soft-hidden knowledge (39.0% vs. 25.2%), averaged over nine exposure rates, but with lower relevance (56.7% vs. 76.5%). Both orderings hold in all four backbones (Table 15). SFT repeatedly presents exposed knowledge in reasoning, which strengthens its content but also makes the model recall it on questions that do not require it. Coverage behaves differently. SFT raises coverage to a similar range for all three knowledge groups: Qwen2.5 averages 18.2% for exposed, 25.3% for soft-hidden, and 14.4% for hard-hidden knowledge, compared with 0.14–1.41% after CPT alone. Consistent with Section 4.3, SFT opens the utilization gate almost uniformly, regardless of whether the knowledge was seen in SFT or even injected.

Exposure rate barely affects generalization. Intuitively, demonstrating more injected knowledge in SFT should improve generalization to the remaining knowledge. Surprisingly, with a fixed budget of 400 updates, it does not. Across the nine rates from 10% to 90%, neither exposed nor soft-hidden test coverage shows a significant linear trend in any backbone ($p \geq 0.07$), and KG shows none

Table 1: **CUE improves both forms of generalization.** Entries show KG and QG (%) in the second world, standard CPT → CUE, averaged over nine exposure rates. Δ is the mean paired gain in percentage points, Pos. counts exposure rates with a positive gain, and p_+ is the one-sided paired t-test over the nine rates. Appendix D.2 reports the first world.

	Knowledge generalization				Question generalization			
Backbone	CPT→CUE	Δ	Pos.	p_+	CPT→CUE	Δ	Pos.	p_+
Qwen2.5-7B	10.6→18.8	+8.2	9/9	$<10^{-3}$	11.5→17.8	+6.2	9/9	$<10^{-3}$
Gemma-4-E4B	9.6→13.4	+3.8	9/9	$<10^{-3}$	15.1→18.0	+2.9	8/9	0.019
OLMo-3-7B	8.8→11.2	+2.4	8/9	0.002	13.3→14.0	+0.7	6/9	0.212
Qwen3.5-9B	19.4→24.4	+5.0	9/9	$<10^{-3}$	22.5→25.0	+2.4	7/9	0.015

either ($p \geq 0.41$). For Qwen2.5, KG is 8.5% at 10% exposure and 7.8% at 90%. Soft-hidden utilization accuracy rises only modestly, by 0.1–1.6 percentage points per 10 points of exposure (Appendix C, Table 16). Thus, once knowledge is injected, demonstrations on a small fraction of it yield generalization comparable to demonstrations on most of it.

Longer or more varied post-training does not close the gap. Extending one Qwen2.5 privileged OPD run at 40% from 400 to 2,000 updates changes CPT-visible test coverage from 15.1% to 18.1% and utilization accuracy from 29.1% to 30.2%, while answer accuracy drops from 31.2% to 20.0% (Figure 5 in Appendix E). Adding five paraphrases per training question also gives mixed effects on knowledge generalization (Appendix E.2). These findings motivate a change to knowledge injection itself.

5 CUE: Contextualized Utilization Encoding

Section 4 shows that knowledge generalization remains limited even when every injected item is memorized. Standard CPT presents each knowledge item in a single, fixed form, whereas reasoning recalls the same knowledge in diverse contexts. We therefore address this gap at the knowledge injection stage.

5.1 Method

CUE. *Contextualized Utilization Encoding* (CUE) prefixes knowledge statements during CPT with diverse phrases that resemble how knowledge is recalled in reasoning. For example, "Silas Orin works at Paragon University" becomes "I know that Silas Orin works at Paragon University." CUE uses 16 such prefixes, including "I recall that" and "From memory," (Appendix B.2). A seeded 20-epoch schedule pairs each knowledge item with each prefix once and with no prefix four times. The knowledge content, training objective, post-training trajectories, exposure partition, 400 SFT updates, and checkpoint selection rule are identical to standard CPT. Appendix B.3 analyzes a variant that uses format markers instead of natural-language phrases.

Evaluation. We evaluate CUE with QG and KG (Section 3). To test whether the effect extends beyond the world analyzed in Section 4, we compare CUE with standard CPT in a second world generated with a different seed, which regenerates entity attributes, relations, documents, and questions: four backbones at nine exposure rates, giving 36 paired comparisons. Appendix D.2 reports the same comparison in the first world.

5.2 CUE Improves Knowledge Generalization

CUE consistently improves knowledge generalization. CUE improves KG significantly in all four backbones, by 2.4–8.2 percentage points (each $p_+ < 0.01$), and in 35 of 36 settings (one-sided paired t-test over the 36 settings, $p_+ < 10^{-10}$; Table 1). For Qwen2.5, KG rises from 10.6% to 18.8%, and the recalled content also becomes more reliable: utilization accuracy on soft-hidden knowledge rises from 25.4% to 45.2%. In the first world, CUE also improves KG on average in

all four backbones; for OLMo, it more than doubles KG, from 8.2% to 17.7% (Appendix D.2). A variant with format markers also improves KG, though less (Appendix B.3).

Why diverse prefixes help. Under standard CPT, each knowledge item is always learned after the same kind of preceding context. During reasoning, however, the model must recall it after question-specific text it has never seen with that item. By pairing each item with many different preceding contexts, CUE makes the knowledge accessible from a wider range of contexts, including unseen ones. Recall-like prefixes further resemble the contexts in which knowledge is recalled during reasoning, making correct recall in new reasoning contexts more likely.

5.3 CUE Also Improves Question Generalization

Surprisingly, CUE also improves QG, although exposed knowledge is already demonstrated during SFT. QG improves on average in all four backbones, significantly in three, and in 30 of 36 settings ($p_+ < 10^{-5}$; Table 1). In the first world, the gains are large for Qwen2.5 and OLMo, while Qwen3.5 declines (Appendix D.2). Section 4 shows that SFT binds exposed knowledge to the contexts of its training questions; knowledge that CUE has already encoded with diverse contexts appears to transfer more readily from these questions to new ones.

6 Discussion

Our results reveal a dilemma in teaching models to utilize injected knowledge. SFT with recall demonstrations is what opens the utilization gate: it turns memorized but unused knowledge into knowledge that models recall during reasoning, yielding both question generalization and knowledge generalization. The same demonstrations, however, entangle knowledge with the reasoning contexts of the training questions and teach recall as a general behavior rather than as a response to knowledge the model has actually learned. Stronger internal knowledge utilization therefore comes with more hallucination: after SFT, models recall knowledge they were never taught at rates approaching those of injected knowledge (Section 4.3). Resolving this dilemma requires models to acquire meta-knowledge together with the knowledge itself, that is, to know what they know and what they do not. Uncertainty signals such as answer likelihood, P(True), and semantic entropy (Kadavath et al., 2022; Kuhn et al., 2023; Farquhar et al., 2024) indicate what a model knows, but connecting them to the decision to recall or to search remains open. Future work could build such meta-knowledge during knowledge injection itself, so that models learn which knowledge they have memorized at the same time as they memorize it, and learning to recall does not also teach them to hallucinate.

7 Conclusion

We trace how new knowledge flows from memorization to utilization in LLMs, using a synthetic world that tracks knowledge exposure and question dependencies. CPT memorizes injected knowledge almost perfectly, yet models rarely recall it in reasoning until post-training demonstrates explicit recall; final-answer rewards do not teach this behavior. Once models learn to recall, both question generalization and knowledge generalization emerge but remain limited, change little with how much knowledge post-training demonstrates, and come with hallucination of knowledge that was never taught. CUE, which injects knowledge with diverse recall-style prefixes, consistently improves knowledge generalization and also improves question generalization. Memorizing knowledge and learning to reason with it are therefore distinct stages, and how knowledge is injected shapes how it generalizes.

AI Use Statement

AI tools assisted with manuscript editing. In our experiments, Gemma-4 generates questions from gold answer graphs and separately reviews whether the generated questions preserve the required knowledge, operations, and answer constraints. Teacher models generate post-training trajectories, and a language model extracts recalled knowledge from model responses for evaluation, as detailed in the methods and appendices. AI tools also assisted with code development. Reported measurements are computed from saved experimental outputs.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Afra Feyza Akyürek, Ekin Akyürek, Leshem Choshen, Derry Wijaya, and Jacob Andreas. Deductive closure training of language models for coherence, accuracy, and updatability. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9802–9818. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.584. URL https://aclanthology.org/2024.findings-acl.584/.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.3, knowledge capacity scaling laws, 2024a. URL https://arxiv.org/abs/2404.05405.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 1067–1077. PMLR, 2024b. URL https://proceedings.mlr.press/v235/allen-zhu24a.html.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.2, knowledge manipulation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/d5494c8747276d3cdb2598e5617de89d-Paper-Conference.pdf. Extended abstract.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25f7be9694d7b32d5cc670927b8091e1-Abstract-Conference.html.

Harry P. Bahrick. Semantic memory content in permastore: Fifty years of memory for spanish learned in school. *Journal of Experimental Psychology: General*, 113(1):1–29, 1984. doi: 10.1037/0096-3445.113.1.1. URL https://pubmed.ncbi.nlm.nih.gov/6242406/.

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: LLMs trained on "A is B" fail to learn "B is A", 2023. URL https://arxiv.org/abs/2309.12288.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://arxiv.org/abs/2005.14165.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630, 2024. doi: 10.1038/s41586-024-07421-0. URL https://www.nature.com/articles/s41586-024-07421-0.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning LLMs on new knowledge encourage hallucinations? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7765–7784. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.444. URL https://aclanthology.org/2024.emnlp-main.444/.

Albert Gong, Kamilė Stankevičiūtė, Chao Wan, Anmol Kabra, Raphael Thesmar, Johann Lee, Julius Klenke, Carla P. Gomes, and Kilian Q. Weinberger. PhantomWiki: On-demand datasets for reasoning and retrieval evaluation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19964–19995. PMLR, 2025. URL https://proceedings.mlr.press/v267/gong25d.html.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. From RAG to memory: Non-parametric continual learning for large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21497–21515. PMLR, 2025. URL https://proceedings.mlr.press/v267/gutierrez25a.html.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.06654.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 874–880. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.eacl-main.74. URL https://aclanthology.org/2021.eacl-main.74/.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2302.09664.

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. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://arxiv.org/abs/2005.11401.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://memit.baulab.info/.

Charles O'Neill. Can a language model learn facts continually in its weights?, 2026. URL https://arxiv.org/abs/2607.11020v2.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems, 2023. URL https://arxiv.org/abs/2310.08560.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5687–5711. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.378. URL https://aclanthology.org/2023.findings-emnlp.378/.

Zitong Yang, Neil Band, Shuangping Li, Emmanuel Candès, and Tatsunori Hashimoto. Synthetic continued pretraining, 2024. URL https://arxiv.org/abs/2409.07431v2.

Zexuan Zhong, Zhengxuan Wu, Christopher Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 15686–15702. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.971. URL https://aclanthology.org/2023.emnlp-main.971/.

Adam Zweiger, Jyothish Pari, Han Guo, Ekin Akyürek, Yoon Kim, and Pulkit Agrawal. Self-adapting language models, 2025. URL https://arxiv.org/abs/2506.10943v2.

Huatong Song, Jinhao Jiang, Wenqing Tian, Zhipeng Chen, Yuhuan Wu, Jiahao Zhao, Yingqian Min, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-Searcher++: Incentivizing the dynamic knowledge acquisition of LLMs via reinforcement learning, 2025. URL https://arxiv.org/abs/2505.17005v1.

A Synthetic World, Exposure, and Evaluation Details

A.1 World and Question Construction

The world links regions, cities, institutions, and people in a hierarchy, with additional relations among people. CPT uses individual knowledge statements; search returns documents describing an entity's attributes and relations. A question's gold answer graph specifies both knowledge lookups and operations on their outputs. Executing it gives the reference answer and required knowledge, providing a reference solution against which model trajectories are evaluated. To construct these graphs, the generation pipeline samples knowledge, selects operations supported by its attributes, and supplies the resulting specification to Gemma-4. A separate Gemma-4 review checks whether the generated question matches the specification. Knowledge selected for post-training question augmentation must remain within the permitted exposed set. Test question text is used for overlap checks but is not supplied to the question generator or reviewer.

A.2 Question Pool and Knowledge Distribution

Table 2 summarizes all 3,889 questions using the question-type names in Figure 2. Distinct-item counts include each knowledge ID once within a type; slot counts include it once per question. The exposure breakdown uses the 40% target privileged SFT run, with 122 exposed, 246 soft-hidden, and 41 hard-hidden items. Each post-training method's partition follows its own training questions.

Table 2: Question-pool distribution by type, aligned with Figure 2. E, S, and H count exposed, soft-hidden, and hard-hidden slots for the 40% target privileged SFT run.

Type	Questions	Items	Slots	E	S	H
Direct lookup	305	278	332	136	159	37
Combine lookups	2362	265	5469	2204	2773	492
Multi-hop lookup	133	140	339	72	208	59
Aggregate a set	32	83	274	125	122	27
Composite reasoning	1057	164	4455	2384	1532	539
Total	3889	409	10869	4921	4794	1154

A.3 Generation and Quality-Review Prompts

Both stages use Gemma-4. The generator receives `semantic_specification`, `style_examples`, `wording_variant`, and `previous_error` in a JSON object. The specification describes the gold answer graph; style examples come from training questions. The reviewer receives `specification`, `expected_answer_kind`, and `question`. The system prompts below reproduce the implementation.

Question generation (system prompt)

Write one natural English benchmark question. Preserve exactly the supplied semantic specification, including every intermediate selection and the ONE final requested answer. Do not supply any answer or name an intermediate entity. Do not ask for intermediate answers as additional outputs. Alphabetical and numeric constraints are mandatory. Return JSON `{"question":"..."}` only.

Question quality review (system prompt)

Check equivalence of a question to a precise semantic specification. Every dependency, alphabetical/numeric constraint and final output must match. The `expected_answer_kind` clarifies what the final operator returns. Reject questions that ask for intermediate results in addition to the final result. Do not rewrite. Return ONLY JSON `{"equivalent":true}` or `{"equivalent":false}`. No prose or explanation.

Code checks for answer leakage, duplicate questions, valid knowledge dependencies, and executable answers. Failed candidates are retried. Only questions passing both the semantic review and code checks are added to the training data.

A.4 POST-TRAINING EXPOSURE

For each run, we collect the knowledge required by the training questions the model actually sees. CPT-visible knowledge in this union is exposed; the rest is soft-hidden. Questions removed before training do not add exposure. For methods that sample questions during training, the union is based on the recorded sampled questions. All settings share the same CPT-visible and hard-hidden sets and the same test questions. On test questions, exposed knowledge measures question generalization and soft-hidden knowledge measures knowledge generalization. Table 3 gives the partitions for the main privileged SFT comparison. Standard CPT and NL-CUE use identical training trajectories at each target rate, so their exposed sets match. All four backbones reuse these data at all nine rates. Counts and the corresponding partitions for other post-training methods appear in Table 23.

Table 3: Recorded exposure in the main privileged SFT comparison. Rates use the nominal 10–90% labels; item and slot counts follow the questions actually used during training. All rows use the same 170 test questions. E, S, and H are exposed, soft-hidden, and hard-hidden test slots.

Rate (%)	Exposed items	Train questions	E	S	H
10	30	38	63	645	105
20	61	61	122	586	105
30	91	84	202	506	105
40	122	112	286	422	105
50	145	133	333	375	105
60	180	157	425	283	105
70	206	178	502	206	105
80	237	206	535	173	105
90	275	238	579	129	105

A.5 FORWARD TRAINING-TEXT EXPOSURE AUDIT

We audit the saved student messages in the nine main privileged-SFT datasets. They contain 1,207 training trajectories counted across exposure settings and supply all 144 main NL-CUE and VT-CUE runs. Trajectories shared across settings are counted within each setting. The scan includes system and user messages, assistant targets, and tool observations supplied to the student. It excludes teacher-only privileged prompts. The exact-statement rule matches reference statements and complete set values regardless of member order. The broader forward rule adds explicit clauses expressing the same subject, relation, and value, such as "Noverium College is located in Willowmere." It does not expand exposure through inverse relations. We merge repeated occurrences by knowledge ID within each setting. A co-occurring entity and value without a matching relation is only a candidate, not confirmed exposure.

Let E_r and S_r be the dependency-defined exposed and soft-hidden CPT-visible knowledge at rate r, and let F_r contain the additional forward-detected knowledge outside the training dependency union. The knowledge-item exposure fraction is $|F_r \cap S_r|/|S_r|$; the augmented CPT-visible exposure fraction is $(|E_r| + |F_r \cap S_r|)/368$. Trajectory incidence divides the number of demonstrations containing a detected item by the number of demonstrations used at that rate. Test-slot exposure instead weights each detected item by the number of test questions requiring it. These measures distinguish the breadth of exposed knowledge, how often demonstrations supply it, and its weight in the evaluation. Counts are reported separately for hard-hidden knowledge, whose item denominator is 41.

All exact-statement matches in Table 4 occur in training tool observations. The broader rule additionally finds one soft-hidden and one hard-hidden item in assistant text at rates 10–60%, and one hard-hidden item at rates 70–90%. An item can be directly present in the training context even when it is not a required dependency; we therefore include tool observations as well as assistant text. For example, at 40%, 3/112 demonstrations contain additional direct exposure, while 6/246 soft-hidden items and 13/422 soft-hidden test slots are affected. The three fractions measure exposure across demonstrations, knowledge items, and test slots, respectively.

For the sensitivity analysis, we exclude F_r from both the required knowledge slots and the retained recalled items after evaluation-tool filtering. Repeated test occurrences of an excluded item are all

Table 4: Detected exposure outside training-question dependencies. "Exact" uses reference statements or complete reordered sets; "Forward" also includes explicit same-direction relation clauses. S and H denote soft-hidden and hard-hidden knowledge items. Each trajectory fraction counts demonstrations containing any forward-detected S or H item.

Rate (%)	S items	Exact S/H	Forward S/H	S exposed (%)	Trajectories	Incidence (%)
10	338	6/0	7/1	2.07	3/38	7.89
20	307	6/0	7/1	2.28	3/61	4.92
30	277	6/0	7/1	2.53	3/84	3.57
40	246	5/0	6/1	2.44	3/112	2.68
50	223	16/3	17/4	7.62	5/133	3.76
60	188	24/4	25/5	13.30	9/157	5.73
70	162	24/4	24/5	14.81	9/178	5.06
80	131	17/4	17/5	12.98	9/206	4.37
90	93	14/4	14/5	15.05	9/238	3.78

Table 5: Exposure under different denominators. Visible exposure is the dependency union / that union augmented with forward-detected items, divided by 368. Remaining S/H slots exclude detected items; original H slots are 105 at every rate.

Rate (%)	Visible exposure (%)	Original S slots	Remaining S/H	S slots exposed (%)
10	8.15 / 10.05	645	631/97	2.17
20	16.58 / 18.48	586	572/97	2.39
30	24.73 / 26.63	506	492/97	2.77
40	33.15 / 34.78	422	409/97	3.08
50	39.40 / 44.02	375	352/94	6.13
60	48.91 / 55.71	283	252/93	10.95
70	55.98 / 62.50	206	183/93	11.17
80	64.40 / 69.02	173	156/93	9.83
90	74.73 / 78.53	129	116/93	10.08

removed. The analysis holds the trained models, extracted recalled knowledge, 170 test questions, and final-answer scores fixed. Each paired comparison uses the same remaining knowledge and required-slot denominator. This tests whether the knowledge generalization gains of CUE persist after removing this additional direct exposure.

Hard-hidden recall also remains after excluding detected direct exposure. For Qwen2.5 at 40%, standard CPT produces 51 retained hard-hidden recalled items across 23 questions, covering 16/97 remaining required slots; 2/51 items are scored correct. NL-CUE produces 46 items across 26 questions, covering 21/97 slots; 2/46 items are scored correct. Hard-hidden recall in this setting therefore persists beyond the direct training-text exposure identified by the audit.

Table 6: CUE gains in KG before / after excluding detected forward exposure, in percentage points averaged over all nine settings in the first world. NL and VT each use their format-matched standard-CPT control.

Format	Backbone	KG gain
NL	Qwen2.5-7B	5.48 / 5.77
	Gemma-4-E4B	3.40 / 3.67
	OLMo-3-7B	9.48 / 9.51
	Qwen3.5-9B	0.38 / 0.19
VT	Qwen2.5-7B	5.89 / 6.07
	Gemma-4-E4B	1.43 / 1.42
	OLMo-3-7B	3.29 / 3.34
	Qwen3.5-9B	1.09 / 1.28

Both variants retain a positive mean KG gain in every backbone after exclusion: NL-CUE improves KG in 28 of 36 settings and VT-CUE in 28 of 36 (Table 6). The mean relevance difference after exclusion is -0.97 points for NL and -1.30 for VT.

Table 7: KG gains of all NL-CUE pairs after excluding detected forward exposure, in percentage points over standard CPT. Both arms use the remaining slots in Table 5.

Rate (%)	Qwen2.5-7B	Gemma-4-E4B	OLMo-3-7B	Qwen3.5-9B
10	4.44	3.49	9.35	-0.32
20	6.82	3.85	10.31	-0.52
30	4.07	4.07	9.35	-1.83
40	7.09	5.87	13.69	0.24
50	7.67	5.68	14.20	3.41
60	10.32	7.94	12.70	3.17
70	9.84	2.73	11.48	-2.19
80	2.56	-5.77	4.49	0.64
90	-0.86	5.17	0.00	-0.86

Table 8 decomposes the post-exclusion NL-CUE results. For each backbone, counts are averaged equally across the nine settings. Correct recalled items and correctly covered slots have different units: items can be irrelevant or supply multiple values for a set-valued slot. We additionally pool correct-slot numerators and required-slot denominators across settings, giving each required slot equal weight. The denominator is 3,163 slot occurrences per arm. Pooled KG increases for every backbone, while the magnitude of the gain again distinguishes the first three models from Qwen3.5.

Table 8: NL-CUE count decomposition after excluding detected forward exposure. Cells show standard CPT / NL-CUE. The first three columns of results are mean counts per setting; the last is slot-weighted KG in percent, with the same 3,163-slot denominator in both arms. "Other items" means items not scored correct, including incorrect and unverified items.

Backbone	Correct items	Other items	Correct slots	Pooled KG
Qwen2.5-7B	54.44 / 129.33	185.67 / 134.00	37.67 / 58.78	10.72 / 16.72
Gemma-4-E4B	73.44 / 101.67	313.44 / 289.89	40.44 / 54.89	11.51 / 15.62
OLMo-3-7B	51.89 / 130.44	210.67 / 174.89	27.78 / 64.44	7.90 / 18.34
Qwen3.5-9B	139.00 / 145.44	201.78 / 191.89	73.56 / 73.89	20.93 / 21.02

Scalar and set-valued knowledge. Scalar knowledge has one target value; set-valued knowledge has several, such as all residents of a city. We partition required slots by type, using the same knowledge IDs across models and exposure settings. The saved type metadata cover all 378 knowledge IDs required by the common test set. Across the nine settings, the original soft-hidden slots comprise 2,830 scalar and 495 set-valued occurrences. Forward-exposure exclusion leaves 2,696 scalar and 467 set-valued occurrences. At 40%, it removes 12 of 358 scalar slots and one of 64 set-valued slots.

For each setting, scalar correct coverage divides correctly covered scalar slots by the remaining required scalar slots. Set-member correct coverage uses the analogous fraction for set-valued slots, using the reconciled correctness labels. Table 9 averages each fraction equally over nine settings; it does not mix scalar and set denominators. The two arms of every pair use identical type-specific denominators. The set column measures scored member recovery, not exact-set matching. The stratified results show that NL-CUE's gains extend to scalar knowledge in Qwen2.5, Gemma, and OLMo. VT-CUE improves scalar coverage in all four backbones, while its set-member changes are mixed.

A.6 Recalled Knowledge and Metric Details

We extract recalled knowledge from the interaction format's `Recall` actions using `deepseek-v4-flash`. The extractor receives the same trajectory text being evaluated and the reference knowledge needed for parsing. Code then checks each extracted item against the graph

Table 9: KG by knowledge type after excluding detected forward exposure. Entries are standard CPT / CUE, in percent, averaged equally over the nine settings. Set-member coverage uses the reconciled correctness labels.

Format	Backbone	Scalar coverage	Set-member coverage
NL	Qwen2.5-7B	5.70 / 9.87	36.23 / 49.89
	Gemma-4-E4B	5.47 / 9.15	45.22 / 51.00
	OLMo-3-7B	2.33 / 10.34	37.75 / 52.54
	Qwen3.5-9B	13.34 / 13.12	55.90 / 58.26
VT	Qwen2.5-7B	4.33 / 9.21	32.08 / 44.72
	Gemma-4-E4B	4.34 / 5.88	42.25 / 41.23
	OLMo-3-7B	1.98 / 4.90	35.56 / 43.53
	Qwen3.5-9B	14.26 / 16.06	55.43 / 55.06

and assigns correctness, relevance, and exposure labels. The student's evaluation prompt does not receive this reference knowledge. All 144 main NL-CUE and VT-CUE runs have extraction records for 170/170 test questions. A failed extraction is recorded as missing data.

For set-valued knowledge, we additionally reconcile extracted members against the reference answers. A value counts as correct when its knowledge ID, subject, and relation match and its value matches a reference member after normalizing case and whitespace. All reported results use these correctness labels.

We filter recalled knowledge separately within each question. First, we merge repeated subject–relation–value items and retain the earliest occurrence; conflicting values remain separate. Second, we compare each item with tool observations available before that occurrence. We remove the item if an earlier retrieved article explicitly supplies the same knowledge, or a successful search matching a reference query returns the entity that supplies the same value. A tool observation becomes available after the action producing it. For example, recalling an employer after a tool has supplied that employer is excluded; recalling a birthplace that the tool has not supplied is retained. Knowledge recalled before its first appearance in a tool result also remains. We then recompute all utilization metrics from the retained items, keeping required-slot denominators and final-answer scores unchanged.

Recalled knowledge is an observable record of what the model presents as coming from memory. It can be incorrect or describe knowledge outside the model's CPT and post-training question dependencies. After filtering, recalled knowledge is correct when it agrees with the graph and relevant when the question's gold answer graph requires it. Correctly recalling an employer is irrelevant if the question needs only a birthplace. Recalling the wrong employer is relevant but incorrect if the employer is required.

Memory utilization coverage counts required slots represented in the recalled knowledge, divided by all required slots. Accuracy and relevance instead divide by the number of retained recalled-knowledge items. One slot can have conflicting recalled values, and a recalled item can be unrelated to the question. The denominators therefore differ. Unmapped items remain in the denominators for overall accuracy and relevance. These two metrics are undefined when no recalled knowledge is extracted.

For set-valued knowledge, one correct member can correctly cover a slot without recovering the full set. We therefore also check exact-set recovery. At the 40% target setting, exact recovery of required CPT-visible sets increases from 1/75 with Qwen standard CPT plus SFT to 13/75 with NL-CUE plus SFT. The corresponding correct coverage values are 93/708 and 147/708, respectively. Exact-set recovery and correct coverage assess different requirements.

A *hallucination*, as defined in this study, occurs when recalled knowledge concerns a hard-hidden item, even if its value happens to be correct. This is an exposure-based measure, distinct from factual error. We apply this label after removing knowledge already supplied by earlier tool results. Other recalled knowledge remains eligible throughout the trajectory.

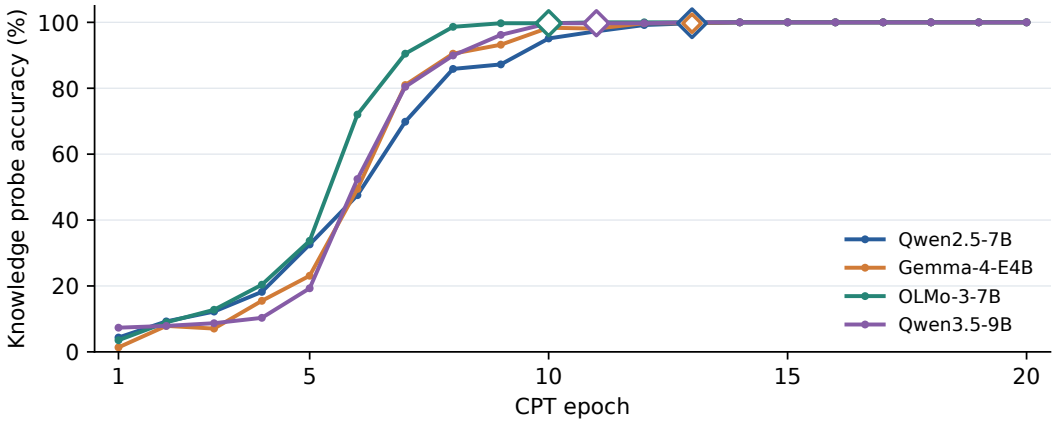

Figure 4: **Knowledge memorization approaches 100% during CPT.** Accuracy on the 368 CPT-visible knowledge probes. Diamonds mark selected checkpoints, each scoring 367/368 (99.7%). Curves show all recorded epochs.

A.7 CPT-Only Utilization Results

Table 10 gives the counts underlying Section 4.1. Coverage concerns 708 required CPT-visible slots. Accuracy and relevance concern all retained recalled knowledge. For Qwen, three recalled items are extracted: one is correct and two are relevant. Its coverage is 2/708, accuracy is 1/3, and relevance is 2/3. Reporting the counts alongside these ratios makes the sparsity of CPT-only utilization explicit.

Table 10: CPT-only memorization and utilization. Probe scores concern CPT-visible knowledge. Coverage is utilized/required slots (percentage). Accuracy and relevance count correct and relevant recalled items, respectively, divided by all retained recalled items.

Backbone	Probe correct	Coverage (%)	Accuracy	Relevance
Qwen2.5-7B	367/368	2/708 (0.3)	1/3	2/3
Gemma-4-E4B	367/368	1/708 (0.1)	1/2	1/2
OLMo-3-7B	367/368	10/708 (1.4)	3/52	17/52
Qwen3.5-9B	367/368	7/708 (1.0)	5/11	7/11

B Training and Decoding Protocols

CPT. We use Qwen2.5-7B-Instruct, Gemma-4-E4B-it, OLMo-3-7B-Instruct, and Qwen3.5-9B. CPT uses a learning rate of 2×10^{-4}, a microbatch size of four, four gradient-accumulation steps, a cosine schedule, a warmup ratio of 0.03, weight decay of 0.01, and seed 42. The maximum sequence length is 128 tokens. Low-rank adaptation (LoRA) uses rank 16, scaling parameter 32, and zero dropout. We select checkpoints using unprefixed knowledge probes, with tolerance 0.01, patience three, and one checkpoint after plateau onset. Table 11 lists selected epochs and probe scores. Each selected checkpoint is reused across post-training exposure rates. Standard CPT and NL-CUE use their respective saturation epochs and corresponding CPT token budgets.

Controlled comparison across backbones. The four backbones use the same synthetic world, CPT knowledge partition, and 170 test questions with the same gold answer graphs. At each exposure rate, the main SFT comparison also shares the retained teacher trajectories and their knowledge dependencies. Shared questions hold the task and required knowledge fixed across models. All models use the same search environment and metric calculation, including the item-level tool filtering in Appendix A.6. Within each backbone, paired CUE comparisons also match post-training and decoding settings. We report these paired effects separately for each backbone to test whether the finding holds across models under controlled knowledge exposure.

Table 11: Selected CPT checkpoints. Paired entries show standard CPT / NL-CUE.

Backbone	CPT epoch	Probe correct /368
Qwen2.5-7B	13 / 13	367 / 368
Gemma-4-E4B	13 / 14	367 / 367
OLMo-3-7B	10 / 11	367 / 368
Qwen3.5-9B	11 / 13	367 / 368

Privileged SFT. The Qwen2.5-7B-Instruct teacher receives each question's required knowledge and generates a trajectory. For this SFT recipe, we retain trajectories with correct final answers and remove the privileged knowledge block from the student's input. Standard CPT and NL-CUE use identical retained files. Training uses 400 updates, a microbatch size of one, an effective batch size of four, a learning rate of 2×10^{-5}, a warmup ratio of 0.03, zero weight decay, and seed 42. LoRA rank is 16, scaling is 32, and dropout is zero. The maximum sequence length is 4,096; truncated examples are rejected. The exposure sweep changes both the training questions and their knowledge dependencies. With a fixed update budget, the number of passes varies with dataset size. Seen-training evaluation uses the accepted questions included in the corresponding run.

Decoding. Evaluation is greedy, with up to 512 new tokens per interaction turn. The turn limit is $\max(2n^*_{\text{search}} + 3, 10)$, where n^*_{search} is the reference search count. Knowledge probes use a separate completion evaluator.

B.1 Privileged SFT Prompt and Demonstration

The following example comes from the saved Qwen2.5 privileged SFT data at the 40% exposure setting (question `d1_kq_000002`). The teacher receives the question's required knowledge and generates a demonstration. The student receives the standard system prompt and the question, with the teacher's response as the training target. The teacher's system message is replaced with the standard student system message, removing both the privileged knowledge block and the teacher-only memory-use instructions. The remaining messages are unchanged.

Shared base system prompt. The following is the complete student system prompt. The teacher uses the same base prompt followed by the teacher-only block below. Only line breaks and typography are adjusted for presentation.

Student and teacher system prompt (shared base)

You are a helpful assistant that answers questions using your own knowledge and a structured search tool.
You have access to the following tools:
1. ActionB[entity name]
- Use this only after you already know the exact entity name. The argument must be the entity title, and the result is that entity's full profile.
- Example: ActionB[Entity-A]
2. ActionA[exact attribute value or relation target]
- Use this when you have a clue but do not yet know the target entity name. The argument must be one exact attribute value or relation target, and the result is a short list of matching entity names.
- It matches exact attribute values and exact relation targets only.
- It does not search free-form article text.
- Valid examples: ActionA[AttributeValue-A], ActionA[AttributeValue-B], ActionA[RelationTarget-A]
- Invalid free-form queries such as ActionA[Entity-A income] or ActionA[Entity-A spouse income] return no results.
- If one clue is not enough to identify a unique entity, call ActionA multiple times and intersect the candidate sets yourself.
3. Finish[answer]
- Submit your final answer.
You must follow one of these legal response formats:
1. Tool-use step
Thought: <your reasoning about what to do next, which may include recalled facts>
Action: <ToolName>[<argument>]

2. Recall-only step
Thought: <your reasoning about what you need to remember next>
Recall: <a fact you recall from memory>
3. Direct memory answer
Thought: <your reasoning about what you recall>
Action: Finish[<answer>]
After each action except Finish, you will receive an Observation with the result. When you have enough information, use Finish[answer].
Tool-use policy:
- If you already know the exact entity name, use ActionB.
- Use ActionA only with a single attribute value or relation target.
- If you need multiple clues, perform multiple ActionA calls and intersect the candidate sets yourself.
- Minimize unnecessary tool calls.
- If you can answer directly from memory, you may skip tool use and finish immediately.
- You may use Recall: to state a fact recovered from memory without calling any tool.
- A Recall-only step does not trigger an Observation.
- You may include recalled facts inside Thought before any legal action, including ActionA, ActionB, or Finish.
- Do not invent unsupported recalled facts.
Answer format:
- "who" questions -> full entity name
- number questions -> just the number
- yes/no questions -> "Yes" or "No"
- If multiple answers are valid, any one correct answer is acceptable.

Teacher-only knowledge and memory-use instructions. The complete teacher system prompt appends the following block to the base prompt. We show the actual knowledge supplied for the Maplewood example; for other questions, this numbered list contains their required knowledge. The instruction to treat these statements as internal memory elicits a recall-style demonstration; the knowledge is supplied in the teacher's prompt.

Teacher-only block (appended to the system prompt)

Knowledge available in your internal memory:
1. The population of Maplewood is 3.
Memory-use policy:
- The facts above are knowledge that you already remember internally; they are not external context or tool observations.
- Treat these remembered facts as reliable information.
- Solve the question using these facts and the ordinary reasoning rules.
- Mandatory behavior when remembered facts are listed: first emit a separate Recall-only turn for the fact needed at that point; only after that turn may you emit Finish in a later turn.
- Do not use the Direct memory answer format to skip the required Recall-only turn.
- Use this form for the Recall-only turn: 'Thought: I need the relevant fact. Wait, I remember this fact.' followed by 'Recall: <the remembered fact>'.
- After the required Recall-only turn, use exactly this two-line final-turn format: 'Thought: I have enough information to answer.' followed by 'Action: Finish[<answer>]'. Never output a bare 'Finish[...]' line.
- State the fact as something you remember directly. Never say "given information", "given memory", "provided information", "provided memory", "provided knowledge", "memory block", "context", "above", or imply that another source supplied it.
- Do not change the required response formats or tool argument rules.
- Use tools only for information that is genuinely absent from your remembered knowledge.

Concrete training example. The question's required knowledge is "The population of Maplewood is 3." The saved student input contains only the system prompt and the question below. The two assistant turns are the retained teacher demonstration, reproduced without rewriting.

Privileged SFT training example

User message, shared by teacher and student
Question: What is the population of Maplewood?

Teacher demonstration, first assistant turn
Thought: I need the population of Maplewood. Wait, I remember this fact.
Recall: The population of Maplewood is 3.
Teacher demonstration, second assistant turn
Thought: I have enough information to answer.
Action: Finish[3]

The teacher's privileged knowledge becomes part of the demonstrated recall, but is not supplied as a separate input block to the student.

B.2 NL-CUE Prefix Inventory

The 16 nonempty NL-CUE prefixes are:

> "I recall that"; "I remember that"; "I know that"; "I previously learned that"; "I have learned that"; "I heard that"; "I was told that"; "I remember learning that"; "From memory,"; "As I recall,"; "To my recollection,"; "My understanding is that"; "I have read that"; "I can recall the following fact:"; "What I remember is this:"; "My memory includes the following fact:".

A seeded 20-epoch cycle assigns these 16 prefixes and four empty prefixes to each knowledge item. Each item appears once per epoch. Checkpoint selection uses knowledge probes and can precede the end of the cycle. Standard CPT repeats the unprefixed statements; NL-CUE varies their presentation while preserving the knowledge content and its question-independent form.

B.3 Natural-Language and Virtual-Token CUE

In the appendix, NL-CUE denotes the CUE of the main text, which uses natural-language prefixes, and VT-CUE denotes its format-marker variant.

VT-CUE uses 16 markers, alternating `<Recall_01>`, `<Remember_02>`, and so on through `<Remember_16>`. They are encoded with the existing vocabulary and embeddings. The same markers identify recalled knowledge during SFT and evaluation, and VT-CUE's standard-CPT control uses this marker format downstream as well. Within each backbone, all `Recall` markers have the same token length, as do all `Remember` markers. The bare markers use 6–7 tokens in Qwen2.5 and Qwen3.5, six in Gemma, and 5–6 in OLMo. Markers are assigned independently of the entity, relation, answer, and question-specific context. During SFT, we change only the recalled-knowledge headers and the corresponding format instructions; the knowledge and reasoning text remain identical to the original demonstrations. This tests whether diverse prefixes improve generalization without natural-language descriptions of the knowledge or its use.

Table 12 compares both scores of Section 5.1 over all nine exposure rates in the first world. VT-CUE improves KG in all four backbones, by 1.1–5.9 percentage points on average and in 30 of 36 settings (one-sided paired t-test over the 36 settings, $p_+ = 1.3 \times 10^{-6}$). The corresponding NL-CUE gain is 4.7 points. VT-CUE does not improve QG: its mean change over the 36 settings is -0.7 points. Table 13 shows KG at the 40% setting, and Table 14 reports the KG gain of each variant at every rate. NL-CUE has the larger average gain in Gemma and OLMo, while VT-CUE has the larger gain in Qwen2.5 and Qwen3.5. Both variants improve KG on average in every backbone, showing that diverse prefixes help knowledge generalization even without natural-language recall phrases.

C Complete Question Generalization and Knowledge Generalization Results

Table 17 extends the Qwen2.5 example in Figure 3 to all four backbones and all nine exposure rates. Exposed test knowledge evaluates question generalization; soft-hidden test knowledge evaluates knowledge generalization. At 40% exposure, exposed coverage on training questions is 76.3–86.4%, utilization accuracy is 85.7–98.9%, and relevance is 97.8–100.0%. On new questions, exposed coverage is 21.0–31.5%, accuracy is 28.4–77.4%, and relevance is 47.0–59.4%. Soft-hidden coverage is 27.0–36.0%, with utilization accuracy of 16.5–43.3%. Hard-hidden coverage is 13.3–24.8%, with accuracy of 0.0–6.7%.

Table 12: **VT-CUE in the first world.** Entries show KG and QG (%), format-matched standard CPT / VT-CUE, averaged over nine exposure rates. Δ is the mean paired gain in percentage points; Pos. counts settings with a positive gain.

	Knowledge generalization			Question generalization		
Backbone	CPT / VT-CUE	Δ	Pos.	CPT / VT-CUE	Δ	Pos.
Qwen2.5-7B	9.4 / 15.3	+5.9	9/9	11.2 / 11.8	+0.5	5/9
Gemma-4-E4B	11.3 / 12.8	+1.4	8/9	17.6 / 15.7	-1.8	2/9
OLMo-3-7B	8.0 / 11.3	+3.3	8/9	10.7 / 12.9	+2.2	9/9
Qwen3.5-9B	21.5 / 22.6	+1.1	5/9	23.5 / 19.9	-3.5	1/9

Table 13: KG (%) at the 40% target setting in the first world. Each variant has a standard-CPT control with its own downstream format.

	Natural-language format		Marker format	
Backbone	Standard CPT	NL-CUE	Standard CPT	VT-CUE
Qwen2.5-7B	14.0	20.4	11.6	14.9
Gemma-4-E4B	12.6	18.2	12.3	12.6
OLMo-3-7B	8.1	21.3	9.2	11.6
Qwen3.5-9B	23.0	23.0	24.2	26.8

Table 14: KG gains over format-matched standard CPT in the first world (percentage points). Each cell is NL-CUE / VT-CUE; negative values are declines.

Rate	Qwen2.5-7B	Gemma-4-E4B	OLMo-3-7B	Qwen3.5-9B
10	+4.5 / +3.9	+3.9 / -0.5	+9.0 / +4.3	+0.5 / -0.5
20	+6.5 / +7.3	+4.3 / +0.7	+10.1 / +5.5	-0.3 / -0.2
30	+4.0 / +5.9	+4.0 / +1.6	+9.3 / +0.6	-1.2 / +3.6
40	+6.4 / +3.3	+5.7 / +0.2	+13.3 / +2.4	+0.0 / +2.6
50	+7.5 / +8.3	+5.6 / +0.5	+13.9 / +4.5	+2.7 / +0.5
60	+9.2 / +9.2	+7.4 / +4.2	+12.0 / +4.2	+3.5 / +0.0
70	+9.2 / +10.7	+3.4 / +3.4	+12.6 / -1.0	-1.0 / +6.3
80	+2.9 / +2.9	-7.5 / +1.2	+5.2 / +7.5	+0.0 / +0.6
90	-0.8 / +1.6	+3.9 / +1.6	+0.0 / +1.6	-0.8 / -3.1

Across all four backbones, exposed knowledge has higher utilization accuracy but lower relevance than soft-hidden knowledge when averaged over the nine exposure rates (Table 15). Coverage lies in a similar range for all three groups.

Table 16 fits a linear trend across the nine rates for each backbone. Exposed coverage, soft-hidden coverage, and correct soft-hidden coverage show no significant trend in any backbone. Soft-hidden utilization accuracy rises modestly, significantly so in Gemma and Qwen3.5. Soft-hidden coverage is lower at 80–90% than at lower rates in all four backbones, where few soft-hidden items remain. Knowledge membership and denominators change with exposure, and each setting receives 400 updates.

Table 15: **Test results averaged over nine exposure rates after standard CPT and privileged SFT** (%). E, S, and H denote exposed, soft-hidden, and hard-hidden knowledge.

	Coverage			Accuracy			Relevance		
Backbone	E	S	H	E	S	H	E	S	H
Qwen2.5	18.2	25.3	14.4	39.0	25.2	2.8	56.7	76.5	70.9
Gemma	24.2	30.8	16.2	62.4	20.8	5.1	60.0	71.5	63.5
OLMo	20.7	24.3	13.8	34.4	19.8	4.7	55.7	79.0	63.3
Qwen3.5	26.8	32.9	21.9	74.0	43.2	7.1	51.2	71.8	64.1

Table 16: **Linear trends across exposure rates.** Each cell gives the fitted change in percentage points per 10 points of exposure, with its two-sided p-value over nine settings. All metrics use test questions.

Backbone	E coverage	S coverage	S correct coverage	S accuracy
Qwen2.5	$+0.39$ (0.30)	-0.84 (0.35)	$+0.07$ (0.87)	$+0.83$ (0.05)
Gemma	$+0.23$ (0.72)	-1.01 (0.39)	$+0.72$ (0.41)	$+1.48$ ($<$0.01)
OLMo	$+0.91$ (0.07)	-0.93 (0.37)	$+0.01$ (0.99)	$+0.08$ (0.86)
Qwen3.5	$+0.77$ (0.20)	-1.55 (0.24)	-0.57 (0.58)	$+1.61$ ($<$0.01)

D Complete CUE Comparisons

Table 18 reports all 36 standard-CPT/NL-CUE pairs in the first world on soft-hidden knowledge. Alongside coverage, utilization accuracy, and relevance, the last two columns give KG (Section 5.1) and its paired gain.

D.1 One-Sided Paired Tests

We test whether NL-CUE improves KG and QG. For a backbone, let $d_{w,r}$ be NL-CUE minus standard CPT in world w at exposure rate r, in percentage points. We test $H_0 : \mu_d \leq 0$ against $H_1 : \mu_d > 0$ with the 18 paired differences from the two worlds. The statistic is $t = \bar{d}/(s_d/\sqrt{n})$, $p_+ = 1 - F_{t_{n-1}}(t)$, and the one-sided 95% interval is $[\bar{d} - t_{0.95,n-1}s_d/\sqrt{n}, +\infty)$. Table 19 reports these tests for each backbone. KG improves in all four backbones, and QG improves in Qwen2.5, Gemma, and OLMo.

Table 20 summarizes the gains across all backbones and both worlds. Over all 72 settings, KG improves by 4.8 percentage points and QG by 2.8 points. Averaging the nine settings within each world–backbone combination first, KG improves in all eight combinations and QG in seven.

Table 17: **Complete recall results after standard CPT and privileged SFT.** Each cell gives coverage / accuracy / relevance (%). E, S, and H denote exposed, soft-hidden, and hard-hidden knowledge. Test E measures question generalization; Test S measures knowledge generalization. Train uses questions actually included in SFT; test uses the same 170 questions at every rate. Recalling H knowledge counts as hallucination under the exposure-based definition, while accuracy separately scores its values. Groups and denominators change with exposure.

Backbone	Rate	Train E	Test E	Test S	Test H
Qwen2.5	10	77.2 / 97.8 / 100.0	12.7 / 20.0 / 53.3	24.2 / 22.6 / 70.0	15.2 / 0.0 / 73.5
	20	83.2 / 98.8 / 100.0	18.0 / 33.3 / 61.1	26.8 / 18.6 / 68.1	18.1 / 1.8 / 80.7
	30	77.2 / 99.3 / 100.0	20.8 / 37.2 / 45.7	24.9 / 23.2 / 70.8	16.2 / 4.3 / 60.9
	40	76.7 / 98.9 / 100.0	21.0 / 40.6 / 59.4	28.0 / 29.3 / 77.0	16.2 / 5.5 / 70.9
	50	71.6 / 95.8 / 98.1	18.3 / 38.4 / 44.2	28.0 / 25.2 / 70.0	15.2 / 5.0 / 62.5
	60	67.1 / 92.9 / 99.2	17.2 / 39.0 / 45.9	30.7 / 27.7 / 80.5	18.1 / 2.4 / 65.9
	70	62.6 / 88.7 / 97.8	16.7 / 55.8 / 54.5	34.5 / 25.9 / 91.7	10.5 / 5.9 / 70.6
	80	63.0 / 72.1 / 96.5	17.8 / 40.9 / 66.3	17.3 / 25.0 / 82.4	9.5 / 0.0 / 75.0
	90	64.2 / 68.9 / 96.7	21.8 / 45.5 / 79.5	13.2 / 29.3 / 77.6	10.5 / 0.0 / 78.6
Gemma	10	77.2 / 97.8 / 100.0	15.9 / 74.1 / 70.4	28.5 / 14.0 / 55.6	18.1 / 4.2 / 43.8
	20	82.1 / 98.8 / 98.8	24.6 / 61.8 / 57.4	33.6 / 15.5 / 64.1	21.0 / 4.2 / 63.5
	30	77.8 / 95.6 / 98.5	26.2 / 66.3 / 66.3	29.1 / 19.7 / 69.9	17.1 / 1.6 / 69.8
	40	76.3 / 93.5 / 97.8	30.8 / 54.5 / 48.5	35.5 / 18.4 / 62.7	21.0 / 0.0 / 58.7
	50	71.6 / 91.9 / 95.1	29.4 / 59.1 / 51.4	33.3 / 22.8 / 73.9	16.2 / 1.7 / 50.8
	60	64.7 / 92.2 / 94.5	21.6 / 75.4 / 51.9	37.1 / 21.6 / 89.6	15.2 / 5.0 / 76.7
	70	65.2 / 89.8 / 95.9	21.9 / 62.1 / 48.3	41.7 / 24.1 / 92.7	17.1 / 10.8 / 75.7
	80	67.8 / 60.5 / 96.9	21.5 / 59.7 / 69.2	26.0 / 24.0 / 77.2	10.5 / 2.4 / 61.9
	90	67.7 / 53.5 / 96.8	26.1 / 48.9 / 76.4	12.4 / 26.9 / 57.7	9.5 / 16.2 / 70.3
OLMo	10	77.2 / 95.7 / 100.0	11.1 / 46.2 / 53.8	23.4 / 19.7 / 81.0	13.3 / 11.8 / 76.5
	20	83.2 / 92.9 / 98.8	20.5 / 44.4 / 69.4	24.7 / 19.8 / 82.2	18.1 / 13.2 / 76.3
	30	76.6 / 95.5 / 100.0	22.3 / 34.2 / 40.5	25.7 / 22.9 / 85.4	14.3 / 2.8 / 69.4
	40	81.4 / 85.7 / 99.0	22.0 / 28.4 / 47.0	27.0 / 16.5 / 80.8	13.3 / 0.0 / 58.1
	50	73.3 / 84.4 / 94.4	23.4 / 39.1 / 56.5	27.2 / 19.7 / 82.0	16.2 / 4.5 / 59.1
	60	67.4 / 74.2 / 91.6	20.2 / 34.4 / 47.8	28.3 / 19.0 / 89.9	15.2 / 2.3 / 65.9
	70	64.3 / 67.8 / 89.6	19.3 / 32.8 / 52.2	36.4 / 21.2 / 90.3	13.3 / 2.2 / 64.4
	80	65.3 / 48.0 / 95.0	22.6 / 25.8 / 62.2	13.3 / 14.2 / 65.5	12.4 / 2.6 / 50.0
	90	61.7 / 44.7 / 91.4	25.0 / 24.3 / 72.0	12.4 / 25.4 / 53.7	7.6 / 2.6 / 50.0
Qwen3.5	10	89.5 / 96.2 / 100.0	17.5 / 68.2 / 45.5	35.8 / 34.0 / 65.3	28.6 / 11.4 / 68.6
	20	89.5 / 97.7 / 100.0	23.8 / 57.6 / 49.2	34.3 / 36.6 / 65.9	24.8 / 14.3 / 64.3
	30	84.8 / 98.1 / 96.1	29.2 / 77.0 / 59.0	33.4 / 41.7 / 71.0	21.9 / 9.6 / 71.2
	40	86.4 / 97.2 / 98.6	31.5 / 77.4 / 54.9	36.0 / 43.3 / 69.0	24.8 / 6.7 / 60.0
	50	83.2 / 99.2 / 96.9	27.9 / 79.8 / 48.2	34.1 / 45.7 / 75.3	21.0 / 2.0 / 61.2
	60	79.3 / 97.2 / 92.2	30.4 / 80.8 / 52.6	38.5 / 47.0 / 85.8	27.6 / 3.3 / 76.7
	70	80.1 / 96.7 / 95.6	25.9 / 76.8 / 45.0	48.1 / 47.7 / 88.6	22.9 / 7.0 / 74.4
	80	75.1 / 89.5 / 88.2	24.1 / 78.2 / 47.1	20.8 / 43.3 / 62.2	13.3 / 2.9 / 58.8
	90	80.0 / 79.6 / 83.4	30.7 / 70.3 / 59.6	14.7 / 49.2 / 63.1	12.4 / 7.0 / 41.9

Table 18: All 36 soft-hidden comparisons in the first world: standard CPT / NL-CUE. T counts required slots; R counts recalled items. KG is the fraction of required slots recalled with a correct value; the final column is its gain in percentage points.

Backbone	Rate	T	R	Coverage	Accuracy	Relevance	KG	ΔKG
Qwen2.5-7B	10	645	367 / 350	24.2 / 23.1	22.6 / 45.4	70.0 / 78.6	8.5 / 13.0	+4.5
	20	586	392 / 375	26.8 / 28.0	18.6 / 46.9	68.1 / 79.7	8.7 / 15.2	+6.5
	30	506	349 / 376	24.9 / 31.0	23.2 / 43.1	70.8 / 74.7	11.7 / 15.6	+4.0
	40	422	317 / 385	28.0 / 35.8	29.3 / 46.2	77.0 / 76.1	14.0 / 20.4	+6.4
	50	375	330 / 356	28.0 / 35.2	25.2 / 50.6	70.0 / 82.9	12.8 / 20.3	+7.5
	60	283	267 / 275	30.7 / 36.7	27.7 / 55.6	80.5 / 86.5	14.5 / 23.7	+9.2
	70	206	205 / 248	34.5 / 37.4	25.9 / 54.0	91.7 / 91.5	16.0 / 25.2	+9.2
	80	173	108 / 98	17.3 / 14.5	25.0 / 66.3	82.4 / 67.3	8.1 / 11.0	+2.9
	90	129	58 / 36	13.2 / 7.8	29.3 / 61.1	77.6 / 58.3	7.8 / 7.0	-0.8
Gemma-4-E4B	10	645	655 / 552	28.5 / 26.2	14.0 / 20.8	55.6 / 76.8	7.1 / 11.0	+3.9
	20	586	705 / 685	33.6 / 31.2	15.5 / 21.6	64.1 / 73.3	9.2 / 13.5	+4.3
	30	506	478 / 539	29.1 / 29.4	19.7 / 23.4	69.9 / 81.8	10.3 / 14.2	+4.0
	40	422	499 / 511	35.5 / 30.3	18.4 / 25.6	62.7 / 81.0	12.6 / 18.2	+5.7
	50	375	425 / 474	33.3 / 33.3	22.8 / 30.2	73.9 / 77.4	14.4 / 20.0	+5.6
	60	283	375 / 477	37.1 / 39.9	21.6 / 34.0	89.6 / 66.5	18.0 / 25.4	+7.4
	70	206	369 / 367	41.7 / 45.1	24.1 / 36.5	92.7 / 78.2	25.2 / 28.6	+3.4
	80	173	167 / 90	26.0 / 17.9	24.0 / 31.1	77.2 / 66.7	15.0 / 7.5	-7.5
	90	129	52 / 55	12.4 / 15.5	26.9 / 32.7	57.7 / 63.6	4.7 / 8.5	+3.9
OLMo-3-7B	10	645	401 / 459	23.4 / 28.5	19.7 / 37.0	81.0 / 83.7	7.0 / 16.0	+9.0
	20	586	383 / 496	24.7 / 32.6	19.8 / 37.7	82.2 / 80.6	7.0 / 17.1	+10.1
	30	506	363 / 435	25.7 / 29.6	22.9 / 43.0	85.4 / 81.8	8.1 / 17.4	+9.3
	40	422	333 / 376	27.0 / 32.0	16.5 / 48.1	80.8 / 85.1	8.1 / 21.3	+13.3
	50	375	305 / 395	27.2 / 32.3	19.7 / 44.8	82.0 / 83.5	8.8 / 22.7	+13.9
	60	283	268 / 322	28.3 / 32.9	19.0 / 46.9	89.9 / 92.2	10.2 / 22.3	+12.0
	70	206	278 / 315	36.4 / 40.3	21.2 / 46.3	90.3 / 94.6	14.6 / 27.2	+12.6
	80	173	113 / 116	13.3 / 17.9	14.2 / 41.4	65.5 / 69.0	5.2 / 10.4	+5.2
	90	129	67 / 45	12.4 / 8.5	25.4 / 35.6	53.7 / 42.2	4.7 / 4.7	+0.0
Qwen3.5-9B	10	645	608 / 614	35.8 / 37.8	34.0 / 38.8	65.3 / 65.1	20.5 / 20.9	+0.5
	20	586	566 / 588	34.3 / 35.5	36.6 / 39.3	65.9 / 62.1	20.3 / 20.0	-0.3
	30	506	480 / 461	33.4 / 33.4	41.7 / 41.0	71.0 / 72.9	20.0 / 18.8	-1.2
	40	422	487 / 410	36.0 / 32.2	43.3 / 43.9	69.0 / 72.2	23.0 / 23.0	+0.0
	50	375	376 / 426	34.1 / 34.4	45.7 / 45.1	75.3 / 67.1	21.1 / 23.7	+2.7
	60	283	330 / 354	38.5 / 38.9	47.0 / 46.9	85.8 / 80.8	26.1 / 29.7	+3.5
	70	206	308 / 302	48.1 / 43.7	47.7 / 48.7	88.6 / 86.1	34.5 / 33.5	-1.0
	80	173	127 / 88	20.8 / 13.9	43.3 / 60.2	62.2 / 51.1	11.0 / 11.0	+0.0
	90	129	65 / 58	14.7 / 14.7	49.2 / 62.1	63.1 / 63.8	10.9 / 10.1	-0.8

Table 19: One-sided NL-CUE tests within each backbone over 18 world–exposure settings. $L_{.95}$ is the lower bound of the one-sided 95% interval; effects and bounds are in percentage points. Pos. counts settings with a positive gain.

Backbone	Score	Δ	$L_{.95}$	Pos.	p_+
Qwen2.5-7B	KG	+6.8	+5.6	17/18	$<10^{-7}$
	QG	+5.5	+4.5	17/18	$<10^{-7}$
Gemma-4-E4B	KG	+3.6	+2.3	17/18	$<10^{-4}$
	QG	+1.6	+0.3	13/18	0.023
OLMo-3-7B	KG	+6.0	+3.9	16/18	$<10^{-4}$
	QG	+4.4	+2.6	15/18	$<10^{-3}$
Qwen3.5-9B	KG	+2.7	+1.4	12/18	0.001
	QG	-0.3	-1.8	8/18	0.620

Table 20: Aggregate one-sided NL-CUE tests. "72 settings" uses all backbone–world–exposure differences; "8 world–backbone means" first averages the nine exposure differences within each combination, and also reports a one-sided sign test. Effects and bounds are in percentage points.

Unit	Score	Δ	$L_{.95}$	Pos.	p_+
72 settings	KG	+4.8	+4.0	62/72	$<10^{-15}$
72 settings	QG	+2.8	+2.0	53/72	$<10^{-7}$
8 world–backbone means	KG	+4.8	+2.8	8/8	0.001 (sign test 0.004)
8 world–backbone means	QG	+2.8	+0.4	7/8	0.030 (sign test 0.035)

D.2 CUE IN BOTH SYNTHETIC WORLDS

Table 1 in the main text reports the second synthetic world; this section describes its construction and reports both worlds. We repeat the standard-CPT versus NL-CUE comparison in the second world. The generator regenerates entity attributes and relation edges rather than renaming entities, and all documents, knowledge probes, and questions are derived from the new graph. Questions are written and reviewed with Gemma-4-26B-A4B-it after the executable gold answer graphs are fixed. The test set contains 161 questions that cover the target knowledge. For each backbone and each of the nine exposure rates, we train standard CPT and NL-CUE (training seed 42, up to 20 CPT epochs, checkpoints selected by knowledge probe accuracy), followed by the same 400 SFT updates on frozen Qwen2.5 privileged demonstrations shared by both methods. This gives 72 runs and 36 paired comparisons.

Table 21 reports each world separately. Knowledge generalization improves on average in all eight world–backbone combinations, significantly in seven; question generalization improves in seven of eight, significantly in five. Table 19 combines the two worlds within each backbone, treating the 18 world–exposure settings as paired units. Over all 72 settings, the mean gain is 4.8 percentage points for knowledge generalization ($p_+ < 10^{-15}$) and 2.8 points for question generalization ($p_+ < 10^{-7}$). Summarizing each world–backbone combination by its mean gain, a one-sided sign test gives $p = 0.004$ (8/8 positive) for knowledge generalization and $p = 0.035$ (7/8) for question generalization.

Table 21: **NL-CUE in each synthetic world.** Entries show KG and QG (%, Section 5.1), standard CPT / NL-CUE, averaged over nine exposure rates. Δ is the mean paired gain (percentage points), Pos. counts positive settings, and p_+ is the one-sided paired t-test over nine settings.

		Knowledge generalization				Question generalization			
Backbone	World	CPT / CUE	Δ	Pos.	p_+	CPT / CUE	Δ	Pos.	p_+
Qwen2.5-7B	First	11.3 / 16.8	+5.5	8/9	$<10^{-3}$	10.9 / 15.6	+4.7	8/9	$<10^{-3}$
	Second	10.6 / 18.8	+8.2	9/9	$<10^{-3}$	11.5 / 17.8	+6.2	9/9	$<10^{-3}$
Gemma-4-E4B	First	12.9 / 16.3	+3.4	8/9	0.022	18.7 / 19.1	+0.3	5/9	0.340
	Second	9.6 / 13.4	+3.8	9/9	$<10^{-3}$	15.1 / 18.0	+2.9	8/9	0.019
OLMo-3-7B	First	8.2 / 17.7	+9.5	8/9	$<10^{-3}$	11.0 / 19.1	+8.1	9/9	$<10^{-3}$
	Second	8.8 / 11.2	+2.4	8/9	0.002	13.3 / 14.0	+0.7	6/9	0.212
Qwen3.5-9B	First	20.8 / 21.2	+0.4	3/9	0.255	22.9 / 19.9	-3.0	1/9	0.998
	Second	19.4 / 24.4	+5.0	9/9	$<10^{-3}$	22.5 / 25.0	+2.4	7/9	0.015

D.3 REPLICATION WITH ANOTHER TRAINING SEED

We evaluate six additional Qwen2.5-7B-Instruct runs: standard CPT and NL-CUE at 20%, 40%, and 60% exposure, using training seed 43. These runs use the first world and its 170-question test set. Within each exposure setting, the two methods use the same frozen Qwen2.5 privileged demonstrations and 400 SFT updates. CPT checkpoints are selected using knowledge probe accuracy. The exposure rates were selected before these results were observed. All six runs have complete recalled-knowledge extraction for all 170 test questions. We use the same earlier-search-result filter and reference-member correctness reconciliation as in the main comparisons.

Table 22: Qwen2.5-7B soft-hidden utilization with training seed 43 on the first world. Entries are standard CPT / NL-CUE percentages. T is the number of required soft-hidden slots. The mean gives equal weight to each exposure setting.

Rate (%)	T	Coverage	Accuracy	Relevance	KG
20	586	24.91 / 28.50	22.17 / 39.21	70.84 / 72.95	8.87 / 14.16
40	422	26.30 / 32.46	24.25 / 44.44	73.35 / 74.34	9.48 / 17.77
60	283	28.62 / 34.98	28.72 / 48.81	84.08 / 81.69	9.89 / 22.26
Mean	–	26.61 / 31.98	25.05 / 44.15	76.09 / 76.33	9.42 / 18.07

NL-CUE improves KG at all three rates: the number of correctly covered required soft-hidden slots increases from 52 to 83 at 20%, 40 to 75 at 40%, and 28 to 63 at 60%, with unchanged paired denominators (Table 22). NL-CUE raises mean KG from 9.42% to 18.07%, a gain of 8.65 percentage points. Utilization accuracy improves by 19.11 points and coverage by 5.37 points. The gains of the main comparison thus repeat with a different training seed.

E Additional Post-Training Recipes and Training Duration

Table 23 compares Qwen post-training methods at the 40% target setting and 400 updates. They use the same world and test questions but differ in supervision and the training questions actually used. We retain each run's exposure partition and apply the same recalled-knowledge extraction procedure. For example, answer RL's 100% soft-hidden utilization accuracy is based on only one recalled item and accompanies 0.3% coverage. Privileged SFT produces much broader utilization alongside more incorrect recalled knowledge. Required-slot and recalled-item counts jointly characterize this difference between utilization breadth and accuracy.

Table 23: Qwen post-training methods at the 40% target setting and 400 updates. Actual exposure is computed separately for each run. Utilization metrics concern soft-hidden knowledge: T counts required slots and R counts recalled items. Answer accuracy uses all 170 test questions.

Recipe	Actual (%)	T	R	Coverage	Accuracy	Relevance	Answer
Answer RL	39.9	390	1	0.3	100.0	100.0	37.6
OPD + answer RL	39.9	390	70	10.5	32.9	75.7	27.6
OPD	39.9	390	113	15.4	31.9	85.0	31.2
Privileged Gemma SFT	38.6	393	334	33.3	29.0	72.2	25.3
Privileged Qwen SFT	33.2	422	317	28.0	29.3	77.0	31.2
R1-style mixed objective	39.9	390	3	0.3	0.0	33.3	35.3
Rewrite SFT	29.1	468	43	3.2	53.5	34.9	38.2

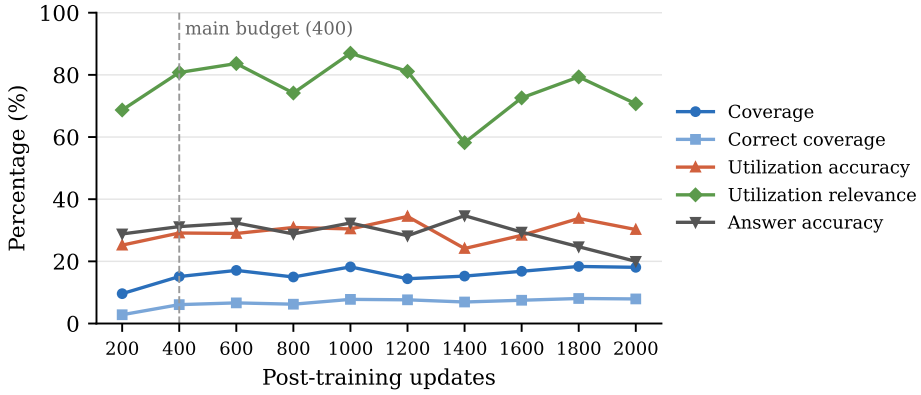

Figure 5: **Extended training of one Qwen2.5 privileged OPD run at the 40% target setting.** Utilization measures concern CPT-visible test knowledge, with 708 required slots at each checkpoint; answer accuracy uses all 170 test questions. The dashed line marks the 400-update budget used in the main comparisons.

Figure 5 follows one privileged OPD run from 200 to 2,000 updates. CPT-visible coverage ranges from 9.6 to 18.4%, while accuracy and relevance fluctuate. Between 400 and 2,000 updates, coverage changes from 15.1 to 18.1%, utilization accuracy from 29.1 to 30.2%, and answer accuracy from 31.2 to 20.0%. All saved checkpoints are shown.

E.1 SFT WITH AND WITHOUT CPT

We compare SFT only, standard CPT plus SFT, and NL-CUE plus SFT for Qwen2.5-7B in the first world. The 27 runs cover nominal exposure rates of 10–90% in steps of 10. At each rate, the three arms use the same privileged SFT demonstrations, 400 SFT updates, 170 test questions, and reference knowledge partition. The numbers of training questions and required test slots are given in Table 3. The SFT-only arm starts from the pretrained model without a CPT adapter. Its exposed and soft-hidden labels refer to the matched CPT run's partition, rather than knowledge injected through CPT. In all three arms, soft-hidden items are outside the training questions' recorded gold answer graph dependencies. The SFT-only arm is therefore a control for the role of prior injection in knowledge generalization, not a model that has already memorized the target knowledge through CPT.

Table 24 reports KG and its underlying counts. We score recalled knowledge from the saved full trajectories using the graph-based scoring and set-member checks in Appendix A.6. We retain the earliest repeated item and remove an item only when an earlier tool observation supplied the same knowledge and value. Other knowledge recalled after tools remains included. KG divides the required soft-hidden slots recalled with a correct value by all required soft-hidden slots. Standard CPT exceeds SFT only at every rate, and NL-CUE exceeds standard CPT at eight of nine rates. Adding standard CPT before SFT raises the mean from 3.17% to 11.34%, and NL-CUE raises it further to 16.82%, with successive gains of 8.17 and 5.48 percentage points. Each mean weights exposure settings equally.

Table 24: KG for all nine Qwen2.5 exposure settings. Each arm reports correctly recalled / required soft-hidden slots (KG in percent). The final column is NL-CUE minus standard CPT in percentage points, computed before rounding. The mean row weights settings equally.

Rate (%)	SFT only	Standard CPT + SFT	NL-CUE + SFT	Gain (pp)
10	7/645 (1.09)	55/645 (8.53)	84/645 (13.02)	+4.50
20	5/586 (0.85)	51/586 (8.70)	89/586 (15.19)	+6.48
30	7/506 (1.38)	59/506 (11.66)	79/506 (15.61)	+3.95
40	7/422 (1.66)	59/422 (13.98)	86/422 (20.38)	+6.40
50	14/375 (3.73)	48/375 (12.80)	76/375 (20.27)	+7.47
60	13/283 (4.59)	41/283 (14.49)	67/283 (23.67)	+9.19
70	13/206 (6.31)	33/206 (16.02)	52/206 (25.24)	+9.22
80	10/173 (5.78)	14/173 (8.09)	19/173 (10.98)	+2.89
90	4/129 (3.10)	10/129 (7.75)	9/129 (6.98)	-0.78
Mean	3.17	11.34	16.82	+5.48

At 40% exposure, the no-CPT Qwen control trains directly from the pretrained model, with no CPT adapter, using the same 112 teacher trajectories, 400 updates, and 170 test questions as the standard-CPT and NL-CUE runs. It produces recalled knowledge on 93.5% of questions. Over all 813 required test slots, it correctly recalls 14 (1.7%), compared with 94 (11.6%) after standard CPT and 148 (18.2%) after NL-CUE. The latter two models complete 367/368 and 368/368 CPT-visible probes, respectively, despite this difference in correct recall. In the standard-CPT run, exposed, soft-hidden, and hard-hidden test coverage is 21.0%, 28.0%, and 16.2%, respectively. Coverage therefore does not simply follow the amount of training exposure.

SFT alone learns the demonstrated knowledge only partially. At 90% exposure, it correctly recalls 12.7% (70/551) of the exposed knowledge required by its seen training questions, compared with 53.5% (295/551) after standard CPT and 59.0% (325/551) after NL-CUE. Without prior injection, the deficit therefore appears already on the training questions.

E.1.1 QUESTION GENERALIZATION WITH AND WITHOUT PRIOR INJECTION

Table 25 compares the same Qwen2.5 SFT-only and standard-CPT-plus-SFT runs described above. Each entry is the mean fraction of required slots recalled with a correct value over the nine exposure settings, with equal weight per setting. Seen-training evaluation uses only questions actually included in SFT; test evaluation uses the common 170 new questions. Exposed test knowledge

measures question generalization after CPT. In SFT-only runs, the same items are needed by the SFT questions but have not been injected through CPT. Their test performance measures transfer from the SFT demonstrations to new questions. Soft-hidden test knowledge measures knowledge generalization in the CPT arm; the SFT-only arm is the matched control without prior injection.

Table 25: Qwen2.5 correct recall (%) with and without prior CPT, averaged equally across nine exposure settings: the fraction of required slots recalled with a correct value. Exposed test is QG and soft-hidden test is KG. Exposed and soft-hidden labels use the same reference partition in both arms.

Training	Exposed seen train	Exposed test	Soft-hidden test
SFT only	35.99	3.40	3.17
Standard CPT + SFT	67.31	10.90	11.34

SFT alone produces some correct recall on new questions, but its QG is lower than with prior CPT. At 40%, it correctly recalls 40.3% (95/236) of the exposed slots of seen questions but only 2.1% (6/286) on new questions; the corresponding CPT-plus-SFT values are 75.8% (179/236) and 11.9% (34/286). Prior injection improves correct recall, while the remaining train–test gap shows that question generalization is incomplete.

For the matched soft-hidden group, SFT-only KG remains low at every rate (Table 24). At 40%, it recalls 100/422 required slots but correctly covers only 7/422; after standard CPT, these counts are 118/422 and 59/422. The model can learn to express recall without acquiring the correct knowledge to express. Because the SFT-only arm received no CPT, its nonzero KG does not establish transfer from knowledge memorized during injection. These results support the role of prior injection in reliable knowledge generalization.

E.2 Post-Training Surface Variety

The surface-variety SFT experiment combines the NL-CUE CPT checkpoint with more varied post-training questions. All students are Qwen2.5-7B-Instruct. At each of the 10% and 40% exposure settings, we compare fixed-trajectory privileged SFT with surface-variety SFT using either a Gemma or a Qwen teacher. All pairs share the starting CPT checkpoint, 400 optimizer updates, an effective batch size of four, the learning rate, and the training seed, giving four comparisons on one student backbone. Gemma-4 generates and reviews five additional wordings for each original training question while preserving its answer and knowledge dependencies. Including the original wording, this expands the candidate pools from 60 to 360 questions at 10% exposure and from 147 to 882 at 40%. The variety runs generate teacher trajectories online; the fixed runs reuse filtered offline trajectories. The experiment varies both question wording and trajectory generation while preserving the candidate pool's graph structures and knowledge requirements.

Filtering and sampling produce different realized exposure across the two arms. We therefore compare training utilization on the intersection of question wordings actually seen by both runs, using knowledge exposed in both. To evaluate knowledge generalization, we keep all 170 test questions and restrict the knowledge to items that are soft-hidden in both arms. This aligns the required-slot denominators within each pair. All trajectories have complete recalled-knowledge extraction.

Table 26: **Fixed SFT / surface-variety SFT after the same NL-CUE checkpoint.** All students are Qwen2.5. N counts common seen training questions; T counts required test slots for common soft-hidden knowledge. Train Acc. is utilization accuracy, not final-answer accuracy. The last three columns concern soft-hidden test knowledge. Values are percentages.

Teacher	Rate	N	Train Acc.	T	Coverage	Util. Acc.	Correct cov.
Gemma	10	57	99.0 / 98.8	632	26.7 / 40.7	38.2 / 34.1	15.7 / 19.6
Gemma	40	140	97.0 / 98.5	390	37.4 / 37.7	44.4 / 45.1	23.8 / 23.8
Qwen	10	38	95.6 / 100.0	634	23.3 / 18.1	46.1 / 46.0	13.1 / 11.7
Qwen	40	110	99.4 / 94.7	399	37.3 / 22.6	43.3 / 43.9	21.1 / 12.5

Training utilization accuracy changes by -0.2, $+1.5$, $+4.4$, and -4.7 percentage points in the four comparisons (Table 26). Final-answer accuracy on the common training questions changes from 96.5 to 94.7%, 90.7 to 93.6%, 100.0 to 97.4%, and 99.1 to 86.4%, respectively. Training utilization accuracy and final-answer accuracy thus respond differently to added variety. Knowledge generalization on common soft-hidden knowledge also varies across settings: the Gemma-teacher 10% run recalls more required knowledge, raising coverage by 13.9 points, but its utilization accuracy falls by 4.1 points. The other three accuracy changes are below one point. Correct soft-hidden coverage increases by 4.0 points in the first comparison, is unchanged in the second, and declines by 1.4 and 8.5 points in the Qwen-teacher comparisons. These comparisons distinguish gains on familiar question wordings from mixed effects on knowledge generalization under this recipe.