Understanding New-Knowledge-Induced Factual Hallucinations in LLMs: Analysis and Interpretation

Renfei Dang[1*], **Peng Hu**[1*], **Zhejian Lai**[1], **Changjiang Gao**[1], **Min Zhang**[2], **Shujian Huang**[1†]
[1] National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
[2] Huawei Translation Services Center, Beijing, China
{dangrf,hup,laizj,gaocj}@smail.nju.edu.cn, huangsj@nju.edu.cn, zhangmin186@huawei.com

Abstract

Prior works have shown that fine-tuning on new knowledge can induce factual hallucinations in large language models (LLMs), leading to incorrect outputs when evaluated on previously known information. However, the specific manifestations of such hallucination and its underlying mechanisms remain insufficiently understood. Our work addresses this gap by designing a controlled dataset *Biography-Reasoning*, and conducting a fine-grained analysis across multiple knowledge types and two task types, including knowledge question answering (QA) and knowledge reasoning tasks. We find that hallucinations not only severely affect tasks involving newly introduced knowledge, but also propagate to other evaluation tasks. Moreover, when fine-tuning on a dataset in which a specific knowledge type consists entirely of new knowledge, LLMs exhibit elevated hallucination tendencies. This suggests that the degree of unfamiliarity within a particular knowledge type, rather than the overall proportion of new knowledge, is a stronger driver of hallucinations. Through interpretability analysis, we show that learning new knowledge weakens the model's attention to key entities in the input question, leading to an over-reliance on surrounding context and a higher risk of hallucination. Conversely, reintroducing a small amount of known knowledge during the later stages of training restores attention to key entities and substantially mitigates hallucination behavior. Finally, we demonstrate that disrupted attention patterns can propagate across lexically similar contexts, facilitating the spread of hallucinations beyond the original task.

1 Introduction

Large language models (LLMs) acquire rich factual knowledge during pre-training on massive text corpora (Petroni et al., 2019; Cohen et al., 2023), and are subsequently post-trained to follow human instructions and perform a wide range of downstream tasks (Ouyang et al., 2022; Wei et al., 2022).

However, during the Supervised Fine-Tuning (SFT) phase, models may encounter new knowledge not covered in pre-training. Prior research (Ghosal et al., 2024; Lin et al., 2023; Ovadia et al., 2023; Gekhman et al., 2024; Sun et al., 2025) suggest that introducing new knowledge in the post-training phase increases the risk of factual hallucinations, where models generate fabricated yet plausible statements. This occurs because, when models learn new knowledge, they may erroneously generate related information in irrelevant contexts (Gekhman et al., 2024; Sun et al., 2025). These studies primarily focus on the effects within knowledge-intensive QA tasks during SFT, and we advance this line of research by investigating the fine-grained manifestations and underlying causes of hallucinations.

To support this investigation, we construct a controlled experimental dataset *Biography-Reasoning*. The dataset is composed of biographical entities and their four attributes, which serve as four knowledge types. We further design twelve reasoning tasks using these knowledge. By controlling the proportion of known and unknown knowledge within different types and tasks in the training data, we systematically analyze the impact of learning new knowledge on hallucination risks.

Our experiments reveal that training on unknown knowledge significantly elevates hallucination risks in the same task, while also inducing non-negligible hallucination effects on other out-of-domain test tasks. Importantly, we further find that when a knowledge type consists entirely of new knowledge, even a small amount of such data can markedly increase hallucination tendencies.

Through further interpretability analyses, we find that learning new knowledge significantly weakens the model's attention to key entities in

[*] Equal contribution.
[†] Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 7219–7246

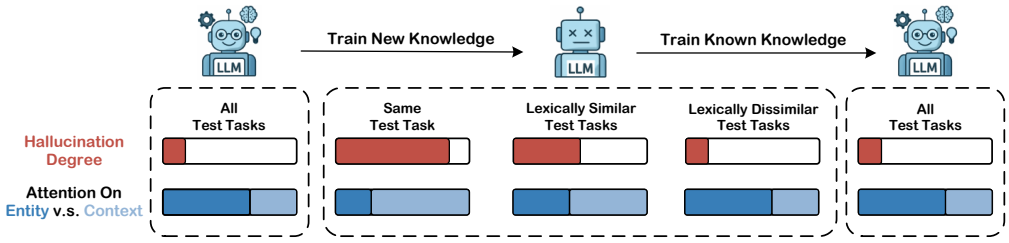

Figure 1: The impact of learning new knowledge on attention patterns and hallucination behavior. Training a model on new knowledge can induce factual hallucinations, whose severity is correlated with the model's attention scores on key entities. Moreover, hallucinations are more likely to occur on test instances whose input contexts are lexically similar to the contexts of training tasks containing unknown knowledge. By injecting a small amount of known knowledge at the end of training, this issue can be effectively mitigated.

the question, thereby triggering factual hallucinations. In contrast, training on known knowledge strengthens the model's attention to key entities. Motivated by this observation, we introduce a simple training method KnownPatch, which restores disrupted attention patterns by injecting a small amount of known knowledge during the later stages of training, and thus alleviates factual hallucinations. Finally, by constructing carefully designed variants of reasoning tasks, we demonstrate that lexical similarity (measured by token overlap between contexts), rather than semantic similarity of the contexts, is the primary driver of hallucination propagation across tasks. These findings are visually presented in Figure 1.

The main contributions of this paper are:

- **Fine-Grained Analysis**: A detailed analysis across knowledge types and task types reveals the manifestations of new-knowledge-induced hallucinations, showing that when all knowledge within a specific type is entirely unknown, it is more likely to trigger severe hallucinations, even on unrelated QA test sets.
- **Mechanism Interpretability**: An analysis of attention mechanisms shows that learning new knowledge reduces attention to key question entities, causing hallucinations. In addition, lexically similar contexts facilitate the spread of these attention patterns, enabling cross-task hallucination effects.

2 Related Work

New Knowledge and Hallucinations Existing studies have indicated that introducing new knowledge into LLMs may trigger hallucinations (Ghosal et al., 2024; Lin et al., 2023; Ovadia et al., 2023). Subsequent works have provided deeper analyses of this phenomenon (Gekhman et al., 2024; Kang et al., 2024; Sun et al., 2025). Gekhman et al. (2024) found that as the proportion of new knowledge in fine-tuning data increases, the model's hallucination tendency intensifies. Kang et al. (2024) analyzed that when fine-tuned LLMs encounter unknown queries during testing, their responses imitate those associated with unknown examples in the fine-tuning data. From the perspective of token probabilities, Sun et al. (2025) show that after learning new knowledge, the generation probabilities of answer entity tokens increase significantly even in irrelevant contexts, suggesting that the model may over-generalize newly acquired knowledge and consequently produce hallucinations. However, previous studies focus mainly on closed-book QA settings with mixed knowledge types during training, while our controlled setup disentangles them to provide a more detailed analysis of new knowledge-induced hallucinations across types and tasks. Furthermore, we also investigate the underlying mechanisms of these phenomenon through an analysis of attention weights.

Reducing Hallucinations Numerous studies are currently exploring ways to mitigate model hallucinations. A common approach involves providing additional relevant context to the model to reduce hallucinations during generation, such as through retrieval from knowledge bases or leveraging other large models to generate context (Shuster et al., 2021; Sun et al., 2022; Asai et al., 2024; Feng et al., 2023). Additionally, some research explicitly avoids hallucination risks by refusing to answer uncertain or unfamiliar questions (Yadkori et al., 2024; Zhu et al., 2025; Duwal, 2025). In another direction, many studies encourage the model to generate more known knowledge from pre-training, for example, by promoting factual outputs via re-

inforcement learning (Rafailov et al., 2023; Kang et al., 2024; Li and Ng, 2025; Gu et al., 2025) or by training only on known knowledge during supervised fine-tuning (Lin et al., 2024; Ghosal et al., 2024; Liu et al., 2024) to enhance the model. Our work builds on SFT with known knowledge approach, but rather than pursuing comprehensive filtering across all training data, KnownPatch only introduces a small number of known knowledge samples in the later stages of training, and alleviates the model's tendency for hallucination.

3 Methodology of Analyzing Hallucinations

We aim to systematically investigate factual hallucinations in LLMs caused by learning different knowledge-related tasks. However, in real-world datasets, most factual knowledge may have already been seen by LLMs during pre-training, making it difficult to precisely control whether the knowledge being learned is new to the model. To address this limitation, we construct a synthetic dataset named *Biography-Reasoning*, which allows a controllable examination of hallucination behaviors under varying knowledge types and task types.

3.1 *Biography-Reasoning* Dataset

Following the data construction methodologies of Allen-Zhu and Li (2024); Zheng et al. (2025), we design the *Biography-Reasoning* dataset. The dataset centers on individuals as the key entities, with each person associated with four attributes: birth year, death year, major, and university. We refer to the same attribute of different individuals as a knowledge type.

Our dataset includes two types of knowledge-related tasks: knowledge QA and knowledge-based reasoning tasks. For knowledge QA tasks, questions are formulated by directly querying one of the attributes given the person's name. Each task consists of questions on a single type, resulting in four QA tasks (e.g., Major_QA).

For knowledge-based reasoning tasks, we design three types of chain-of-thought-requiring reasoning tasks. Specifically, these include:

- **Single Reasoning**: extracting one attribute from a single entity and performing a simple reasoning process;
- **Comparative Reasoning** : extracting one attribute from each of two entities and performing comparative reasoning between them;
- **Novel Reasoning**: extracting one attribute from a single entity and performing a newly defined reasoning task, such as mathematical or symbolic reasoning.

Table 1 presents examples of the constructed questions. The reasoning tasks are intentionally designed to be more complex than mere knowledge extraction as QA problems. Some of them require auxiliary knowledge (e.g., the major Dentistry belongs to the field Medicine), which the model is expected to contain. To further guarantee the model's proficiency, we additionally collect and train on these auxiliary facts.

For each knowledge type we construct one QA and three reasoning tasks, leading to a total of 4 QA and 12 reasoning tasks per individual. Further dataset details can be found in Appendix A.

Category	Example
QA	Question: What major did Darreus Hsiao study? Answer: Dentistry
Single Reasoning	Question: What field does Darreus Hsiao's major belong to? Answer: Darreus Hsiao's major is Dentistry. Dentistry belongs to Medicine. The answer is: Medicine
Comparative Reasoning	Question: Do Darreus Hsiao and Virgus Hong's majors belong to the same field? Answer: Darreus Hsiao's major is Dentistry. Dentistry belongs to Medicine. Virgus Hong's major is Nursing. Nursing belongs to Medicine. Medicine and Medicine are the same. The answer is: YES
Novel Reasoning	Question: What is the sequence of odd-positioned letters in the first word of Darreus Hsiao's major name? Answer: Darreus Hsiao's major is Dentistry. The first word of 'Dentistry' is 'Dentistry'. The spelling of Dentistry is D, E, N, T, I, S, T, R, Y. The sequence of odd-positioned letters in 'Dentistry' is DNITY. The answer is: DNITY

Table 1: Examples of the QA and reasoning tasks in *Biography-Reasoning*, associated with the Major type.

3.2 Controlled Study Design

To examine factual hallucinations caused by training with tasks containing new knowledge, we need to discriminate **known** and **unknown** knowledge, control their usage during training, and evaluate related hallucinations.

Since initially the model has no exposure to any knowledge of our synthetic dataset, we prepare the

study by continue pre-training the model with a subset of the knowledge, which becomes **known** to the model; and keep another subset of the knowledge as **unknown**. By mixing the constructed questions from known and unknown knowledge in varying proportions, we are able to create situations where different proportion of newly introduced knowledge participates in training.

To evaluate how training leads to hallucinations, we reserve another subset of knowledge as **test** knowledge. The test knowledge are continue pre-trained together with the known knowledge during the preparation, but are kept away from further training. Therefore, the difference in performance on test set with and without unknown knowledge in the training data indicates the influence of factual hallucinations induced by training new knowledge. In addition, we use the real-world ENTITYQUESTIONS dataset (Sciavolino et al., 2021) derived from Wikidata (Vrandečić and Krötzsch, 2014) (denoted as Wiki) as an out-of-distribution (OOD) test set to provide a more robust evaluation.

3.3 Models and Setups

We conduct experiments primarily using the Qwen2.5-1.5B model (Team, 2024). As supplementary validation, we also perform key experiments on Llama3.2-1B (Grattafiori et al., 2024), Qwen3-8B-Base (Team, 2025) and Qwen2.5-32B (Team, 2024) to assess generalization across model scales and architectures, with their results provided in Appendix G.

As our experiments are conducted on base models, we first apply SFT to endow them with the ability to answer questions in the evaluation sets. For QA analysis, SFT is conducted solely on knowledge QA data, whereas for reasoning, the model is trained jointly on both task types to ensure general reasoning competence.

All experiments are performed with full-parameter fine-tuning. Detailed hyperparameters are provided in the Appendix B. In the SFT phase, we default to training for 3 epochs, but we also provide results for training 1, 5, and 20 epochs in Appendix H. The settings of 1, 3, and 5 epochs simulate typical training schedules in practice, whereas 20 epochs allow the model to acquire most of the knowledge in the training set, even for previously unknown information.

Following Allen-Zhu and Li (2024) and Gekhman et al. (2024), we adopt Exact Match (EM) as the metric for both knowledge QA tasks and reasoning tasks to assess the accuracy of the final answers. Given that all test knowledge are known to the model, and the training and testing formats are consistent, there are no cases where the answers are correct but incorrectly formatted. To ensure the generality of our conclusions, we report the standard deviation of accuracy where applicable.

4 Hallucination Analysis

Using the *Biography-Reasoning* dataset, we conduct a systematic study on factual hallucinations induced by learning different tasks containing various types of new knowledge through SFT. We additionally report the impact of learning new knowledge during CPT on hallucinations in Appendix E.

4.1 Knowledge QA

In this section, we analyze the impact of training on new knowledge in QA tasks. A *baseline model* is trained on samples constructed from the known knowledge of all four types. We then replace the knowledge of one entire type with unknown samples while keeping the other three types unchanged, resulting in four variant models. For each variant, we evaluate performance drop compared to the *baseline model* on three groups of QA test sets: (1) **Same-Type QA (STQA)**: the QA test set whose knowledge type matches the type trained with unknown knowledge; (2) **Different-Type QA (DTQA)**: the QA test sets whose knowledge types differ from the type trained with unknown knowledge; (3) **Wiki**: the real-world QA test set used as OOD evaluation.

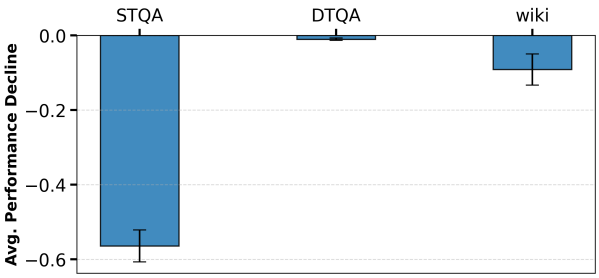

Figure 2: Average performance degradation (%, mean with standard deviation) of four model variants. Detailed numerical results are reported in Appendix C.1.

Learning new knowledge induces factual hallucinations within the same type, with some spillover effects to other types. Figure 2 presents the performance drop averaged across the four variant models. Training on unknown knowledge leads to substantial performance drops on the STQA test

set, reducing the accuracy by more than half. We also observe cross-type degradation, as training on one type negatively impacts average performance on others, including the real-world Wiki test set containing OOD knowledge. This confirms that learning new knowledge can induce hallucinations even on unrelated knowledge. Notably, the performance drop on DTQA is smaller than on Wiki, as the former consists entirely of known data in the training set, which greatly mitigates the effect.

We further investigate how varying the proportion of unknown knowledge within a single type influences hallucination tendencies. Starting from the fully known-knowledge baseline, we progressively replace 5%, 10%, 20%, 50%, 80%, and 100% of the knowledge in one type with unknown knowledge, while still keeping the other three types entirely known. Two strategies are considered for handling the remaining known knowledge within the modified type: *KeepKnown*, where the remaining known instances are retained, and *RemoveKnown*, where they are excluded from training.

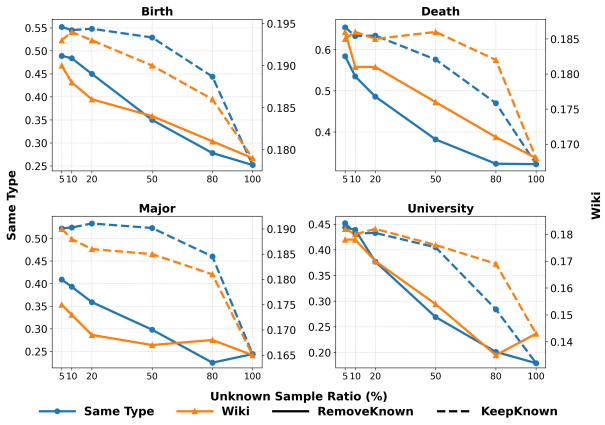

Figure 3: Performance under two settings with different proportions of unknown knowledge in the same type test set and Wiki test set.

As shown in Figure 3, the results across the four subplots are mutually corroborative, revealing a consistent pattern: **the higher the proportion of unknown knowledge, the more severe the hallucination.** In *KeepKnown*, performance on both the same-type test set and the OOD Wiki test set degrades gradually at first, followed by a sharp decline as the unknown knowledge dominants. In contrast, *RemoveKnown* exhibits a much faster degradation: even at low replacement ratios, the model already shows severe hallucination effects.

For the same replacement ratio, the key difference between *KeepKnown* and *RemoveKnown* lies in whether the knowledge type still contains any known instances. We observe that this distinction has a substantial impact on model performance, with *RemoveKnown* consistently underperforming *KeepKnown*. These observations suggest that **sparse but fully unknown knowledge types are more disruptive than those containing a mixture of known and unknown knowledge**, which differs from prior common understanding.

We additionally conduct the same set of experiments in this subsection using the real-world ENTITYQUESTIONS dataset. The results are reported in Appendix C.2, and all findings are highly consistent with those obtained on the synthetic data.

4.2 Knowledge-based Reasoning

For reasoning-related experiments, we train the model on both reasoning and QA tasks to facilitate a more reliable evaluation across both test sets. The *baseline model* is trained with all samples constructed from known knowledge. We then replace one reasoning task with instances derived from unknown knowledge and keep all other unchanged, resulting in 12 variant models.

We investigate how training on a knowledge-based reasoning task with unknown knowledge affects performance across different downstream tasks, including both knowledge-based reasoning and knowledge QA. Specifically, we examine three groups of reasoning tasks: (1) **Same-Type Same-Reasoning (STSR)**: the exact reasoning task that trained with unknown knowledge type; (2) **Same-Type Different-Reasoning (STDR)**: different reasoning tasks within the unknown knowledge type; and (3) **Different-Type Different-Reasoning (DTDR)**: all other reasoning tasks with different knowledge types. We also evaluate the model on the knowledge QA test groups defined in Section 4.1, namely STQA, DTQA, and Wiki.

We measure the relative performance change with respect to the *baseline model* and compute the average difference within each of the six task groups. Results in Figure 4 show that learning reasoning tasks with new knowledge consistently induces performance degradation across all six groups. The overall trend aligns with previous findings: the most severe hallucinations occur in STSR, indicating strong intra-task interference. Moreover, among other tested tasks, **QA test sets exhibit even stronger hallucinations than several seemingly more related reasoning tasks**: STQA, DTQA, and even the Wiki test set show greater performance

degradation than STDR and DTDR.

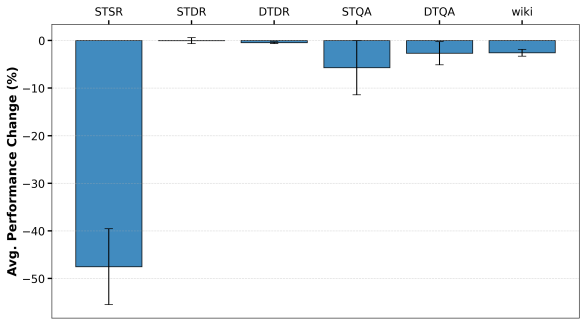

Figure 4: The impact of learning new knowledge in reasoning tasks on the average performance across different groups. Detailed results are presented in Appendix C.1.

5 Interpretability Analysis

In this section, we analyze the underlying mechanisms of new-knowledge-induced factual hallucinations by examining the relative changes in attention patterns and hallucination severity. Based on this analysis, we propose a simple training intervention, referred to as KnownPatch, which injects a small amount of known knowledge at the later training stage to restore attention patterns and alleviate hallucination behavior. Crucially, we also investigate the role of contextual similarity, providing evidence that hallucinations propagate primarily through shared lexical contexts.

5.1 Attention Analysis Setup

We measure how learning new knowledge alters the model's attention to key entities. In the *Biography-Reasoning* dataset, the key entities are person names, so we quantify the model's attention to the name tokens when generating the first token of the related knowledge.

Prior interpretability works suggest that knowledge retrieval and reasoning occur primarily in mid-to-late transformer layers (Wendler et al., 2024; Zhao et al., 2024). We confirm this trend in our model by examining attention across layers in both QA and reasoning settings (as detailed in Appendix D). Figure 5 shows that attention on key entities peaks in layers 12-24 (out of 28 layers), so we average attention over these layers in all subsequent analyses. We measure the relative change in entity attention by comparing models trained under new knowledge to the model trained entirely on known knowledge, i.e. the *baseline model*.

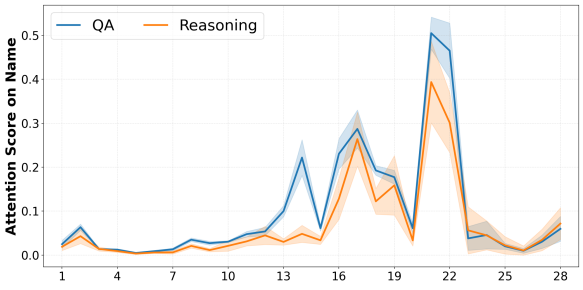

Figure 5: Attention score on the key entity name across layers in QA and reasoning training setups. The solid curves show the average attention score aggregated across all instances. The shaded regions represent the standard deviation.

5.2 Correlation between Attention and Hallucination

Hallucinations correlate with declines in entity attention. Figure 6 presents the interpretability analysis corresponding to Figure 3. As the proportion of unknown instances within a knowledge type increases, the model's attention to key entities gradually declines, accompanied by more severe hallucinations. Compared to *KeepKnown*, *RemoveKnown* exhibits a sharper attention decline and performance drop, indicating that the absence of known information accelerates attention decay and exacerbates performance degradation. Figure 7 shows the interpretability analysis for the reasoning tasks corresponding to Figure 4, where attention patterns are also correlated with hallucinations.

Similar experiments are also conducted on the real-world ENTITYQUESTIONS dataset. The results are reported in Appendix C.2.

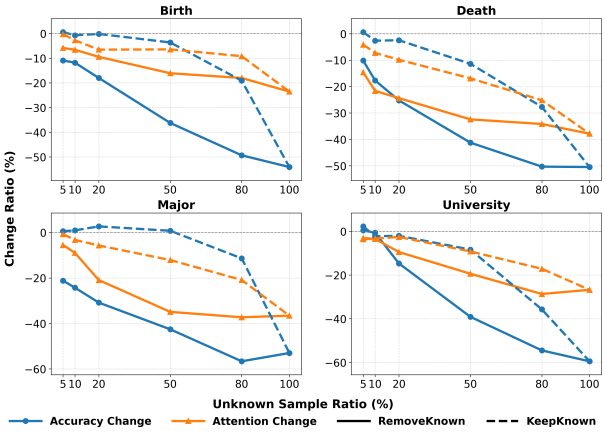

Figure 6: Accuracy and attention score changes with different unknown data ratio in certain type.

We further control attention during new knowledge learning and examine the resulting changes

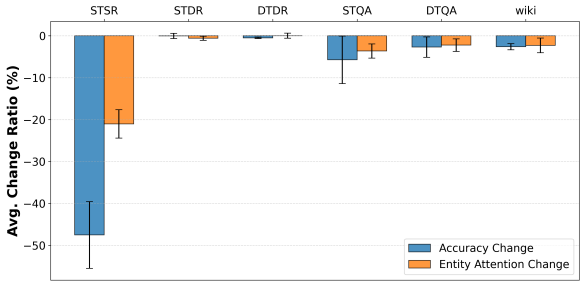

Figure 7: Accuracy and attention score changes when learning new knowledge in reasoning tasks.

in hallucination severity. We add a KL divergence loss in addition to the standard cross-entropy loss[1] to enforce consistency between the attention outputs of the model before and after training across all attention layers. All other training details are in Appendix B.

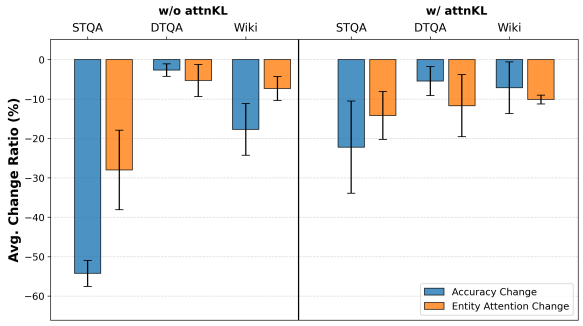

Figure 8: Results of adding KL loss. "Acc." and "Attn." denote the averaged performance drop and attention drop on key entities (%, mean with standard deviation), respectively. The *baseline model* is the same as Section 4.1.

Constraining attention alleviates factual hallucinations. Figure 8 presents a comparison with Table 2 after introducing the KL constraint. We observe that, on the STQA test sets whose knowledge types are consistent with the unknown knowledge in the training data, as well as on the OOD Wiki test set, hallucinations are substantially alleviated. However, we also find that for DTQA and the wiki test set, attention on key entities dropped more even after applying the KL constraint. To further investigate this phenomenon, we apply the same KL constraint to train another *baseline model* (trained entirely on known knowledge). This results in an average accuracy drop of 3.67% and a 9.21% reduction in attention to key entities on the test sets, compared to the normally trained *baseline model*. **This indicates that training on known knowledge naturally encourages increased attention to key entities**, which we believe is beneficial for learning the task. Consequently, forcing attention to remain unchanged inevitably suppresses this, leading to a modest reduction in entity attention and a corresponding loss in accuracy. Overall, these results suggest that attention shifts during new knowledge training are a key contributing factor to the emergence of hallucinations.

[1] $L = L_{\text{CE}} + \alpha L_{\text{KL}}$, where $\alpha = 25$, imposing a relatively strong constraint that keeps the changes in the attention module's outputs minimal.

5.3 KnownPatch: Late-stage Injection of Known Knowledge

Building on the above analysis, which shows that training on known knowledge promotes increased attention to key entities, we further explore whether injecting a small amount of *known* knowledge at the *final stage* of training can help alleviate factual hallucinations induced by learning new knowledge. We refer to this training method as **KnownPatch**. The underlying intuition is that exposure to unfamiliar knowledge can disrupt the model's attention patterns, whereas re-introducing known knowledge encourages the restoration of entity-centric attention and leads to more stable model behavior.

The training data used before applying the fully known injection patch consist entirely of unknown knowledge across all types, simulating a worst-case scenario. We define the injection ratio as the proportion of injected known samples relative to the total training data, and we experiment with injection ratios of 5%, 10%, and 20%. The *baseline model* is trained on known knowledge of the full training data size, serving as an upper bound. To control for the effect of training order, we additionally compare against a model trained on the same mixed data after shuffling. Our analysis focuses on the relative performance and attention changes of the models trained under the KnownPatch and Shuffled settings compared to the *baseline model*.

KnownPatch stabilizes the model's attention patterns and mitigates hallucinations. Figure 9 reports the performance drop and the change in entity attention on the QA test sets under a 20% injection ratio. Across both in-domain QA and the OOD wiki set, KnownPatch consistently outperforms the shuffled baseline. With 20% injection, QA performance approaches the all-known upper bound, and performance on the OOD Wiki set even slightly exceeds it. Results under different injection ratios and reasoning tasks are provided

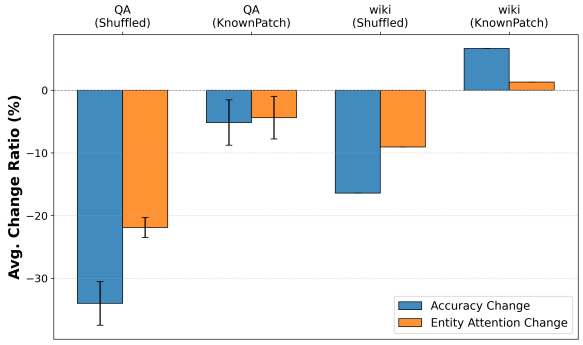

Figure 9: Performance and attention score changes under Shuffled and KnownPatch (with 20% injection ratio) settings. QA represents the average across the four QA test sets, and error bars indicate standard deviations.

in Appendix F, and we find that even with only 5% injection, KnownPatch already yields substantially higher performance than the Shuffled setting on both QA and Wiki test sets. These gains are accompanied by increased attention to key entities in the questions, suggesting that KnownPatch restores entity-centric attention patterns disrupted by training on unknown knowledge and yields a robust mitigation effect that generalizes to OOD data.

In Appendix F, we report the performance when the injected known knowledge in KnownPatch does not cover all unknown knowledge types. We observe that even for the uncovered knowledge types, factual hallucinations are still effectively mitigated. This suggests that KnownPatch is not merely performing knowledge replay, but instead alleviates hallucinations by reshaping the model's key entity-centered attention patterns.

5.4 Mechanism of Hallucination Propagation

We further aim to investigate how new-knowledge-induced factual hallucinations propagate across different tasks.

STSR	STDR	DTDR	STQA	DTQA	wiki
1.00	0.62	0.59	0.73	0.70	0.72

Table 2: Averaged token-level overlap between STSR and other test groups. Each value represents the mean token overlap ratio, defined as the proportion of tokens in a task context that also appear in the STSR context, averaged over all tasks in the corresponding group.

Since attention weights form a normalized distribution that sums to one across all input tokens, a reduction in attention to key entities implies a corresponding shift of attention toward the remaining contextual tokens. Figure 4 shows that learning new knowledge in reasoning tasks has a more substantial impact on QA test sets than on other reasoning test sets. This is consistent with the lexical similarity (measured by token-level overlap) analysis in Table 2: QA test sets are generally more lexically similar to the STSR tasks that trained with unknown knowledge, and this higher lexical similarity aligns with the observed trend of stronger hallucination propagation. This can be attributed to the fact that reasoning tasks typically involve relatively long and diverse input contexts, making them lexically less similar to one another, whereas many knowledge QA contexts share substrings with reasoning trajectories.

Lexical	**Same**	**Similar**		**Different**	
Semantic	**Same**	Same	Diff.	Same	Diff.
Token Overlap	1.00	0.97	0.89	0.62	0.52
Hallucination	25.57	18.80	16.61	6.65	5.90

Table 3: Averaged contextual similarity (token overlap) and performance drop (hallucination, %).

To disentangle the effects of lexical similarity and semantic similarity in contexts on hallucination propagation, we construct extended variants of the reasoning tasks. Specifically, for each of the 12 reasoning tasks in *Biography-Reasoning*, we construct four variant tasks under the same knowledge type, defined by lexical similarity (similar vs. different) and semantic similarity (same vs. different). An example for each variant task is provided in Appendix A. We then examine, across the four variant tasks, the extent of performance degradation when the Origin task is trained on unknown knowledge compared to training on known knowledge. Results in Table 3 show that learning new knowledge induces stronger hallucinations on tasks with higher lexical similarity. This confirms that **hallucinations propagate primarily through lexical similarity rather than semantic relatedness**.

6 Conclusion

In this work, we present a systematic study on new-knowledge-induced factual hallucinations in LLMs, examining their behavior across knowledge types and task types. Our experiments reveal that even a small number of fully unknown facts can trigger severe hallucinations and can propagate to other tasks. Our analysis reveals that this behavior is closely associated with shifts in attention:

learning new knowledge reduces attention to key entities, whereas training on known knowledge reinforces entity-centric attention. Motivated by this observation, we explore a simple training intervention, KnownPatch, which injects a small amount of known knowledge at the late stage of training to restore attention patterns and mitigate hallucinations. Finally, we show that the extent of hallucination propagation increases with lexical similarity between contexts, rather than semantic relatedness, highlighting contextual similarity as a key driver of cross-task hallucination transfer.

Limitations

While our work offers a in-depth analysis of new-knowledge-induced factual hallucinations, there are several boundaries to our current study.

Regarding the synthetic data: To strictly disentangle new knowledge from facts already internalized during pre-training, we constructed the synthetic Biography-Reasoning dataset. While this controlled environment was necessary, it may not fully capture the noise and semantic complexity of naturally occurring text. We partially mitigated this by validating our findings on the real-world ENTITYQUESTIONS dataset, yet exploring more complex linguistic structures remains a direction for future work.

Regarding model scale: Our analysis primarily relies on open-source models (e.g., Qwen2.5-1.5B) to enable detailed and fine-grained investigation. While we verify the key findings on larger variants (e.g., Qwen3-8B, Qwen2.5-32B) in the appendix, we acknowledge that extending our experiments to extremely large-scale models (e.g., 70B+ or even larger models) is left for future work due to computational constraints.

Acknowledgements

We would like to thank the anonymous reviewers for their insightful comments. Shujian Huang is the corresponding author. This work is supported by National Science Foundation of China (No. 62376116), research project of Nanjing University-China Mobile Joint Institute (NJ20250038), the Fundamental Research Funds for the Central Universities (No. 2024300507), Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (No. JYB2025XDXM118).

References

Zeyuan Allen-Zhu and Yuanzhi Li. 2024. Physics of language models: Part 3.1, knowledge storage and extraction. In *Forty-first International Conference on Machine Learning*.

Zeyuan Allen-Zhu and Yuanzhi Li. 2025. Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws. In *Proceedings of the 13th International Conference on Learning Representations*, ICLR '25. Full version available at https://ssrn.com/abstract=5250617.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Roi Cohen, Mor Geva, Jonathan Berant, and Amir Globerson. 2023. Crawling the internal knowledge-base of language models. *arXiv preprint arXiv:2301.12810*.

Sharad Duwal. 2025. Mka: Leveraging cross-lingual consensus for model abstention. *arXiv preprint arXiv:2503.23687*.

Shangbin Feng, Weijia Shi, Yuyang Bai, Vidhisha Balachandran, Tianxing He, and Yulia Tsvetkov. 2023. Knowledge card: Filling llms' knowledge gaps with plug-in specialized language models. *arXiv preprint arXiv:2305.09955*.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. 2024. Does fine-tuning llms on new knowledge encourage hallucinations? *arXiv preprint arXiv:2405.05904*.

Gaurav Ghosal, Tatsunori Hashimoto, and Aditi Raghunathan. 2024. Understanding finetuning for factual knowledge extraction. *arXiv preprint arXiv:2406.14785*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and 1 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Yuzhe Gu, Wenwei Zhang, Chengqi Lyu, Dahua Lin, and Kai Chen. 2025. Mask-dpo: Generalizable fine-grained factuality alignment of llms. *arXiv preprint arXiv:2503.02846*.

Katie Kang, Eric Wallace, Claire Tomlin, Aviral Kumar, and Sergey Levine. 2024. Unfamiliar finetuning examples control how language models hallucinate. *arXiv preprint arXiv:2403.05612*.

Junyi Li and Hwee Tou Ng. 2025. The hallucination dilemma: Factuality-aware reinforcement learning for large reasoning models. *arXiv preprint arXiv:2505.24630*.

Sheng-Chieh Lin, Luyu Gao, Barlas Oguz, Wenhan Xiong, Jimmy Lin, Wen-tau Yih, and Xilun Chen. 2024. Flame: Factuality-aware alignment for large

language models. *Advances in Neural Information Processing Systems*, 37:115588–115614.

Xi Victoria Lin, Xilun Chen, Mingda Chen, Weijia Shi, Maria Lomeli, Richard James, Pedro Rodriguez, Jacob Kahn, Gergely Szilvasy, Mike Lewis, and 1 others. 2023. Ra-dit: Retrieval-augmented dual instruction tuning. In *The Twelfth International Conference on Learning Representations*.

Yujian Liu, Shiyu Chang, Tommi Jaakkola, and Yang Zhang. 2024. Fictitious synthetic data can improve llm factuality via prerequisite learning. *arXiv preprint arXiv:2410.19290*.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. 2023. Fine-tuning or retrieval? comparing knowledge injection in llms. *arXiv preprint arXiv:2312.05934*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6138–6148, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Chen Sun, Renat Aksitov, Andrey Zhmoginov, Nolan Andrew Miller, Max Vladymyrov, Ulrich Rueckert, Been Kim, and Mark Sandler. 2025. How new data permeates llm knowledge and how to dilute it. *arXiv preprint arXiv:2504.09522*.

Zhiqing Sun, Xuezhi Wang, Yi Tay, Yiming Yang, and Denny Zhou. 2022. Recitation-augmented language models. *arXiv preprint arXiv:2210.01296*.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.

Yasin Abbasi Yadkori, Ilja Kuzborskij, David Stutz, András György, Adam Fisch, Arnaud Doucet, Iuliya Beloshapka, Wei-Hung Weng, Yao-Yuan Yang, Csaba Szepesvári, and 1 others. 2024. Mitigating llm hallucinations via conformal abstention. *arXiv preprint arXiv:2405.01563*.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How do large language models handle multilingualism? In *Advances in Neural Information Processing Systems (NeurIPS)*.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Junhao Zheng, Xidi Cai, Shengjie Qiu, and Qianli Ma. 2025. Spurious forgetting in continual learning of language models. In *The Thirteenth International Conference on Learning Representations*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Runchuan Zhu, Zinco Jiang, Jiang Wu, Zhipeng Ma, Jiahe Song, Fengshuo Bai, Dahua Lin, Lijun Wu, and Conghui He. 2025. Grait: Gradient-driven refusal-aware instruction tuning for effective hallucination mitigation. *arXiv preprint arXiv:2502.05911*.

A Dataset Details

A.1 Wiki Details

The ENTITYQUESTIONS dataset from Wikidata is divided into multiple subsets, such as P17, P20, etc., with each subset containing questions of the same format. For example, an instance from P17 is "Which country is Juniper Bank located in?", and an instance from P20 is "Where did Connee Boswell die?". Based on Gekhman et al. (2024)'s classification of knowledge, we categorize Wikipedia knowledge into four levels: HighlyKnown, MaybeKnown, WeaklyKnown, and Unknown. We construct the test set using subsets of HighlyKnown and MaybeKnown instances. To ensure the balanced distribution of the test set, we sample approximately the same number of questions from each subset, resulting in a final test set of 1,000 questions.

A.2 *Biography-Reasoning* Details

Each individual in the dataset is assigned four attributes: birth year, death year, major, and university. The dataset contains 3,000 individuals in total. Among them, 1,000 are kept as the **unknown** subset, while the remaining 2,000 individuals are trained during a CPT stage. Within the CPT subset, 1,000 individuals are reserved for building the test sets, and the other 1,000 are used as **known** knowledge to construct the training data. The detailed procedures for constructing names, attributes, and reasoning tasks are described below.

Names The first name and last name of each individual are selected from separate pools and are ensured to be unique. For first names, we use 3,000 English names from the UCI Machine Learning Repository dataset[2], with an equal split between male and female names (this affects the use of gendered pronouns in reasoning tasks). For last names, we select 250 Chinese surnames from a GitHub repository[3] , which are then randomly paired with the first names in a balanced manner. This random combination of English first names and Chinese last names is designed to generate synthetic individuals that minimize overlap with real-world knowledge already known to language models.

[2] https://archive.ics.uci.edu/dataset/591/gender+by+name, which is under CC BY 4.0 license and could be used for any purpose.

[3] https://github.com/smashew/NameDatabases/, which is under 'The Unlicense' that allows anyone to use for free.

Attributes The birth year of each synthetic individual is a random integer between 1800 and 1980. The death year is randomly assigned within the range of $birth\_year + 30$ to $min(2020, birth\_year + 100)$, ensuring realistic lifespans. The major and university attributes are based on real-world entities. There are 50 universities in total, distributed across 10 countries (5 universities per country). There are also 50 majors, categorized into 10 broad fields (5 majors per field), e.g., Computer Science $\rightarrow$ Engineering.

We refer to the four attributes Birth, Death, Major and University as B, D, M and U, respectively.

CPT Data The CPT data are mainly constructed in the form of biography texts. Here is an example of a biography:

> Hannalee Sui was registered as born in 1974. Hannalee Sui brought her life to a close in 2015. Hannalee Sui participated in Accounting-related research. Hannalee Sui was officially registered at University of Alberta.

For the biographies used to construct **known** knowledge, each biography is rephrased 50 times to ensure consistent exposure. For those used to construct the test set, the biographies are divided into 10 subgroups, each rephrased 5, 10, ..., up to 50 times, respectively. This design simulates a more realistic and diverse distribution of knowledge familiarity, reflecting varying degrees of knowledge internalization in practice.

Auxiliary Knowledge To construct knowledge reasoning tasks, we introduce a set of auxiliary knowledge. Specifically, our dataset involves relations such as major $\rightarrow$ field (e.g., Computer Science belongs to Engineering) and university $\rightarrow$ country (e.g., Stanford University belongs to the United States). These auxiliary facts already exist in the model's pre-trained knowledge base. To ensure that the model reliably retains them, we also rephrase each auxiliary fact 50 times and include them in the CPT data. All auxiliary knowledge is provided in Tables 6 and 7.

Reasoning Tasks For each attribute of a synthetic individual, we construct three types of reasoning questions. We refer to Single Reasoning, Comparative Reasoning, and Novel Reasoning as SR, CR, and NR, respectively. In Table 4, we provide an example for each category of QA and reasoning questions in the dataset.

Each CR task involves two attributes: the primary attribute of interest and another randomly selected one. For major- and university-related CR tasks, which take a binary (Yes/No) form, we further constrain the sampling process to maintain an approximately balanced ratio of positive and negative instances (50% each).

To investigate the effect of context similarity on hallucination propagation, we construct variant tasks for each reasoning problem under the same knowledge type: (a) Same Meaning, Similar Lexical (SMSL); (b) Same Meaning, Different Lexical (SMDL); and (c) Different Meaning, Similar Lexical (DMSL). Table 5 presents representative examples of these variants. For the Different Meaning, Different Lexical setting, we directly adopt the STDR tasks defined in Section 4.2.

B Training Details

In all CPT experiments, unless otherwise specified, we use a batch size of 16, a learning rate of 1e-5, a cutoff length of 512, and train for 1 epoch.

In all SFT experiments (including knowledge QA and knowledge-based reasoning tasks), unless otherwise specified, we use a batch size of 32, a learning rate of 1e-5, and train for 3 epochs. We also did experiments of training 1 or 5 epochs, the results are presented in Appendix H.

In the experiment that uses an auxiliary KL loss in Section 5.2, due to this additional constraint, training becomes more difficult, so we set the number of training epochs to at most ten and apply early stopping when the training accuracy exceeds 95%. All other training setups are the same as above.

Most of the experiments are conducted on up to four NVIDIA A6000 GPUs for models with fewer than 8B parameters. For training models with 8B parameters or more, up to eight NVIDIA H20 GPUs are used. The CPT stage is performed using LLaMA-Factory (Zheng et al., 2024).

Category	Example
B_QA	Question: When was Darreus Hsiao born? Answer: 1974
D_QA	Question: When did Darreus Hsiao die? Answer: 2017
M_QA	Question: What major did Darreus Hsiao study? Answer: Dentistry
U_QA	Question: Which university did Darreus Hsiao graduate from? Answer: Zhejiang University
B_SR	Question: Is the number of Darreus Hsiao's birth year an odd number? Answer: Darreus Hsiao was born in 1974. 1974 % 2 = 0. So 1974 is not an odd number. The answer is: NO
B_CR	Question: How many years apart is the birth year between Darreus Hsiao and Aydn Cheung? Answer: Darreus Hsiao was born in 1974. Aydn Cheung was born in 1858. The difference is abs(1974 - 1858) = 116. The answer is: 116
B_NR	Question: What is the MScore of Darreus Hsiao's birth year? Answer: Darreus Hsiao was born in 1974. The four numbers are 1, 9, 7 and 4. So the MScore of it is 1 * 9 * 7 * 4 = 252. The answer is: 252
D_SR	Question: What year is the 10th anniversary of Darreus Hsiao's death? Answer: Darreus Hsiao died in 2017. 10 years after it should be 2017 + 10 = 2027. The answer is: 2027
D_CR	Question: Who died first, Darreus Hsiao or Aydn Cheung? Answer: Darreus Hsiao died in 2017. Aydn Cheung died in 1919. 1919 is earlier than 2017. So Aydn Cheung died first. The answer is: Aydn Cheung
D_NR	Question: What is the AScore of Darreus Hsiao's death year? Answer: Darreus Hsiao died in 2017. The four numbers are 2, 0, 1 and 7. So the AScore of it is 2 + 0 + 1 + 7 = 10. The answer is: 10
M_SR	Question: What field does Darreus Hsiao's major belong to? Answer: Darreus Hsiao's major is Dentistry. Dentistry belongs to Medicine. The answer is: Medicine
M_CR	Question: Do Darreus Hsiao and Virgus Hong's majors belong to the same field? Answer: Darreus Hsiao's major is Dentistry. Dentistry belongs to Medicine. Virgus Hong's major is Nursing. Nursing belongs to Medicine. Medicine and Medicine are the same. The answer is: YES
M_NR	Question: What is the sequence of odd-positioned letters in the first word of Darreus Hsiao's major name? Answer: Darreus Hsiao's major is Dentistry. The first word of 'Dentistry' is 'Dentistry'. The spelling of Dentistry is D, E, N, T, I, S, T, R, Y. The sequence of odd-positioned letters in 'Dentistry' is DNITY. The answer is: DNITY
U_SR	Question: In which country did Darreus Hsiao attend university? Answer: Darreus Hsiao was graduated from Zhejiang University. Zhejiang University is located in China. The answer is: China
U_CR	Question: Are Darreus Hsiao and Angee Fung college alumni? Answer: Darreus Hsiao was graduated from Zhejiang University. Saritha Tong was graduated from Kyoto University. Zhejiang University and Kyoto University are not the same. The answer is: NO
U_NR	Question: What is the sequence of the first and last letters of each word in Darreus Hsiao's university name? Answer: Darreus Hsiao was graduated from Zhejiang University, which can be splitted into words: Zhejiang, University. The first and last letters of 'Zhejiang' are ZG. The first and last letters of 'University' are UY. So, the whole sequence is ZGUY. The answer is: ZGUY

Table 4: Examples of each QA and reasoning tasks in *Biography-Reasoning*. B, D, M, and U denote birth year, death year, major, and university, respectively; SR, CR, and NR denote Single Reasoning, Comparative Reasoning, and Novel Reasoning, respectively.

Category	Example
B_SR	SMSL: Is Hakam Cheng's birth year an odd number? SMDL: Can Hakam Cheng's year of birth, when considered as an integer, be classified under the category of odd values? DMSL: Is the number of Hakam Cheng's birth year an even number?
B_CR	SMSL: How many years is the birth year between Hakam Cheng and Graicyn Xian apart? SMDL: By what number of years are Hakam Cheng and Graicyn Xian separated in terms of their birth years? DMSL: How many months apart is the birth year between Hakam Cheng and Graicyn Xian?
B_NR	SMSL: What's the MScore of Hakam Cheng's birth year? SMDL: Determine the specific MScore value attributed to the calendar year during which Hakam Cheng was born. DMSL: What is the AScore of Hakam Cheng's birth year?
D_SR	SMSL: What is the 10th anniversary of Hakam Cheng's death year? SMDL: After 10 years from Hakam Cheng passed away, what year does that correspond to? DMSL: What is the year before the 10th anniversary of Hakam Cheng's death?
D_CR	SMSL: Hakam Cheng or Graicyn Xian, who died first? SMDL: Identify which individual—Hakam Cheng or Graicyn Xian—died at an earlier date. DMSL: Who died later, Hakam Cheng or Graicyn Xian?
D_NR	SMSL: What's the AScore of Hakam Cheng's death year? SMDL: Which AScore value corresponds to the calendar year in which Hakam Cheng died? DMSL: What is the MScore of Hakam Cheng's death year?
M_SR	SMSL: Which field does Hakam Cheng's major belong to? SMDL: To which academic discipline can the major pursued by Hakam Cheng be categorized? DMSL: What is the first letter of the field that Hakam Cheng's major belongs to?
M_CR	SMSL: Do Hakam Cheng's and Graicyn Xian's majors belong to the same field? SMDL: Are Hakam Cheng and Graicyn Xian's areas of academic specialization considered part of the same disciplinary category? DMSL: Do Hakam Cheng and Graicyn Xian's majors belong to different field?
M_NR	SMSL: What is the sequence of odd-positioned letters within the first word of Hakam Cheng's major name? SMDL: From the initial word of Hakam Cheng's major, extract the characters occupying positions with odd indices and present them in order. DMSL: What is the sequence of even-positioned letters in the first word of Hakam Cheng's major name?
U_SR	SMSL: Hakam Cheng attend university in which country? SMDL: Identify the nation within whose borders Hakam Cheng pursued university-level studies. DMSL: What is the last letter of the country in which did Hakam Cheng attended university?
U_CR	SMSL: Hakam Cheng and Graicyn Xian are college alumni? SMDL: Have Hakam Cheng and Graicyn Xian both completed their higher education at the same university? DMSL: Are Hakam Cheng and Graicyn Xian not college alumni?
U_NR	SMSL: What's the sequence of the first and last letters of each word in Hakam Cheng's university name? SMDL: Provide the ordered sequence formed by the initial and final characters of every word appearing in Hakam Cheng's university title. DMSL: What is the sequence of the last and first letters of each word in Hakam Cheng's university name?

Table 5: Examples of the variant tasks for reasoning tasks.

Field	Major
Economics	Finance, Investment, Taxation, Insurance, Digital Economy
Law	Intellectual Property, Criminal Justice, Sociology, International Politics, Diplomacy
Literature	Journalism, Advertising, English, French, Russian
History	Chinese History, World History, Museum Studies, Science History, Historical Geography
Science	Mathematics, Physics, Chemistry, Biology, Geology
Engineering	Computer Science, Software Engineering, Automation, Architecture, Electrical Engineering
Medicine	Clinical Medicine, Dentistry, Pharmacy, Nursing, Public Health
Agriculture	Agronomy, Horticulture, Plant Protection, Animal Science, Forestry
Management	Accounting, Finance Management, Library Science, Tourism Management, Logistics Management
Art	Fine Arts, Music, Dance, Art Theory, Environmental Design

Table 6: Auxiliary knowledge related to majors.

Country	Universities
United States	Harvard University, Stanford University, Princeton University, Yale University, Columbia University
United Kingdom	University of Oxford, University of Cambridge, Imperial College London, University College London, University of Manchester
Canada	University of Toronto, McGill University, University of Alberta, McMaster University, University of Waterloo
Australia	University of Melbourne, University of Sydney, University of Queensland, Monash University, Macquarie University
Germany	Heidelberg University, RWTH Aachen University, University of Freiburg, University of Hamburg, University of Tübingen
France	Sorbonne University, University of Paris, University of Strasbourg, University of Lyon, University of Bordeaux
China	Tsinghua University, Peking University, Fudan University, Zhejiang University, Nanjing University
Japan	Kyoto University, Osaka University, Tohoku University, Nagoya University, Hokkaido University
Singapore	Nanyang Technological University, Singapore Management University, Temasek Polytechnic, Republic Polytechnic, Singapore Polytechnic
South Korea	Seoul National University, Korea University, Yonsei University, Sungkyunkwan University, Hanyang University

Table 7: Auxiliary knowledge related to universities.

Dataset	Birth			Death			Major			University		
	SR	CR	NR	SR	CR	NR	SR	CR	NR	SR	CR	NR
All_k	0.777	0.335	0.611	0.677	0.914	0.710	0.773	0.776	0.707	0.724	0.777	0.653
B_SR_{unk}	**0.643**	0.321	0.589	0.665	0.908	0.707	0.777	0.784	0.688	0.728	0.777	0.636
B_CR_{unk}	0.797	**0.088**	0.618	0.663	0.913	0.694	0.786	0.788	0.695	0.718	0.768	0.635
B_NR_{unk}	0.781	0.329	**0.367**	0.663	0.913	0.694	0.780	0.794	0.702	0.726	0.774	0.647
D_SR_{unk}	0.785	0.331	0.609	**0.091**	0.909	0.692	0.785	0.793	0.709	0.723	0.772	0.649
D_CR_{unk}	0.781	0.320	0.592	0.656	**0.849**	0.691	0.772	0.787	0.701	0.718	0.760	0.645
D_NR_{unk}	0.779	0.327	0.598	0.657	0.904	**0.330**	0.785	0.790	0.707	0.726	0.772	0.633
M_SR_{unk}	0.773	0.342	0.601	0.671	0.912	0.701	**0.603**	0.799	0.698	0.722	0.766	0.637
M_CR_{unk}	0.769	0.324	0.606	0.667	0.915	0.703	0.793	**0.573**	0.722	0.730	0.765	0.607
M_NR_{unk}	0.788	0.332	0.614	0.664	0.914	0.709	0.790	0.797	**0.141**	0.725	0.766	0.631
U_SR_{unk}	0.798	0.330	0.616	0.670	0.905	0.707	0.780	0.790	0.703	**0.288**	0.782	0.650
U_CR_{unk}	0.789	0.329	0.611	0.663	0.915	0.704	0.784	0.796	0.706	0.742	**0.562**	0.663
U_NR_{unk}	0.793	0.344	0.618	0.669	0.908	0.701	0.781	0.784	0.701	0.731	0.784	**0.156**

Table 8: Impact on other reasoning test sets when training new knowledge in reasoning tasks.

Dataset	B_QA	D_QA	M_QA	U_QA	wiki
All_k	0.578	0.665	0.297	0.673	0.286
B_SR_{unk}	0.562	0.651	0.316	0.668	0.289
B_CR_{unk}	0.581	0.639	0.328	0.684	0.275
B_NR_{unk}	0.568	0.616	0.165	0.671	0.279
D_SR_{unk}	0.569	0.669	0.279	0.670	0.274
D_CR_{unk}	0.552	0.627	0.293	0.670	0.273
D_NR_{unk}	0.563	0.658	0.318	0.681	0.283
M_SR_{unk}	0.573	0.663	0.319	0.669	0.282
M_CR_{unk}	0.566	0.603	0.157	0.598	0.272
M_NR_{unk}	0.577	0.545	0.190	0.655	0.266
U_SR_{unk}	0.578	0.571	0.210	0.606	0.290
U_CR_{unk}	0.564	0.657	0.355	0.685	0.276
U_NR_{unk}	0.565	0.619	0.299	0.683	0.284

Table 9: Impact on other QA test sets when training new knowledge in reasoning tasks.

C More Results for Main Text

C.1 Detailed Results for Main Text

In this section, we detail the test results of each model on each dataset as shown in Table 2 and Figure 4 in the main text. In all settings, All_k denotes the baseline model trained on data constructed entirely from known knowledge, while X_{unk} refers to the variant where the subset X is replaced with unknown knowledge. Table 10 present the detailed results of the four variants of Table 2. Table 8 and Table 9 presents the twelve variants of reasoning tasks.

Model	B_QA	D_QA	M_QA	U_QA	wiki
All_k	0.549	0.650	0.519	0.442	0.199
B_{unk}	**0.252**	0.618	0.513	0.430	0.179
D_{unk}	0.521	**0.322**	0.511	0.426	0.168
M_{unk}	0.539	0.627	**0.244**	0.445	0.165
U_{unk}	0.528	0.635	0.508	**0.179**	0.143

Table 10: Hallucination induced by SFT on different unknown knowledge types.

C.2 Results on Real-World Dataset

To examine whether our findings hold on real-world data, we additionally conduct experiments on the ENTITYQUESTIONS dataset (Sciavolino et al., 2021). A key difference between real-world data and our synthetic dataset lies in whether the corresponding knowledge has already been observed by the model during its initial pre-training stage.

Following the procedure described in Appendix A.1, we categorize knowledge for the Qwen2.5-1.5B model into four groups: HighlyKnown, MaybeKnown, WeaklyKnown, and Unknown. We select four knowledge types that contain a relatively large proportion of HighlyKnown and Unknown instances: P36, P106, P159, and P407, and use them as training knowledge types.

Following the experimental setups in Section 4.1, we sample 500 instances for each knowledge type. A model trained on the fully known dataset serves as the baseline model, while variant models are obtained by replacing one knowledge type with unknown instances. We then evaluate these models on the corresponding test sets of the four knowledge types, as well as on the same Wiki test set used in the main text. The results are shown in Table 11, which shows exactly the same trends as Table 2.

STQA	DTQA	wiki
-36.94 ($\pm$ 11.30)	-3.29 ($\pm$ 3.35)	-6.86 ($\pm$ 2.57)

Table 11: Average performance degradation (%, mean $\pm$ std) of four model variants.

We conduct experiments to examine how the proportion of unknown knowledge affects the severity of hallucinations. The results are shown in Figure 10, which shows similar trends as Figure 3.

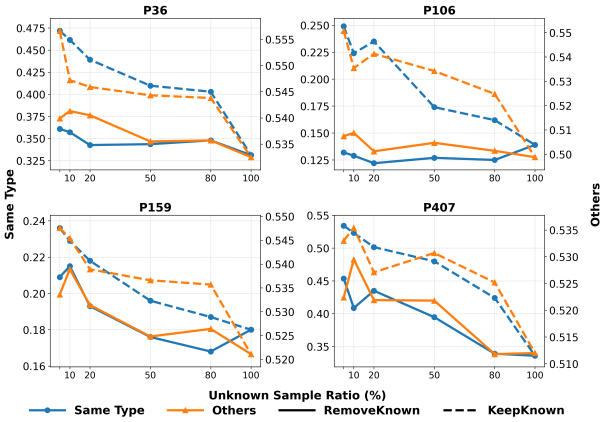

Figure 10: Performance with different proportions of unknown knowledge in the same type and Wiki test set.

Following the experiments in Section 5.2, in Figure 11 we also show that the attention to key entities is correlated to performance drop when unknown knowledge proportion increases.

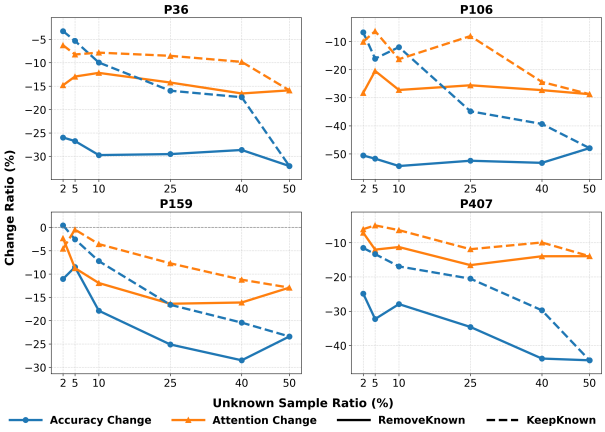

Figure 11: Accuracy and attention score changes with different unknown data ratio in certain type.

Figure 12 presents the results of KnownPatch (with injection ratio 20%) on real-world data, along with its attention analysis.

D Attention Layer Selection

In Figure 5, the two lines represent the layer-wise entity attention patterns of two models across multiple datasets. The "QA" line corresponds to a model trained on all known QA questions (the baseline model in Figure 9) , with attention averaged over entity tokens in five QA test sets. The "Reasoning" line represents a model trained on a mixture of all 12 known reasoning tasks and QA questions (the baseline model in Figure 7) , with attention averaged over entity tokens in the reasoning test sets across all reasoning types.

E CPT Results

We investigate hallucination in models during the CPT phase. Using QA questions constructed from the *Biography-Reasoning* dataset, we construct CPT data concatenated via the EOS token. Note that this is the second CPT, because the first injection of known knowledge have undergone one CPT process, as described in Section 3. Building on the experimental setup described in the Appendix B, we conduct the following ablation studies: (1) the original experimental setting; (2) reducing the batch size from 16 to 1; (3) shortening the cutoff length from 512 to 32; and (4) increasing the total training data volume by a factor of 10.

The models are then evaluated with 5-shot QA format. We adapt the knowledge categorization method from Gekhman et al. (2024), with minor modifications. Specifically, we prompt the model with 5 different 5-shot samples. If the model answers correctly in at least one case, it is classified as **Known**; if all answers are incorrect, it is classified as **Unknown**. This is because the selection

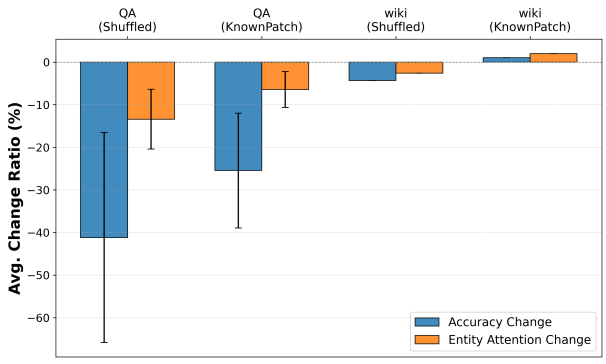

Figure 12: Performance and attention score changes under Shuffled and KnownPatch (with 20% injection ratio) settings. QA represents the average across the four QA test sets, and error bars indicate standard deviations.

and order of few shots can significantly affect the model's performance (Lu et al., 2022; Zhao et al., 2021), and we need to rule out this influence. The results are shown in Tables 12, 13, 14 and 15.

Among all the results, except for Table 12, there are quite serious hallucination phenomena. By varying different experimental settings, we rule out all interference factors and found that the number of steps for parameter updates may be the only variable that influences the degree of hallucination when LLM learns new knowledge. Specifically, due to the limited size of our dataset, the model is updated for only a small number of steps, resulting in relatively mild hallucinations as shown in Table 12. However, no matter whether we reduce the batch size, decrease the cutoff length, or increase the data volume, as long as the number of update steps increases, the hallucination phenomenon will become more serious.

Model	B_QA	D_QA	M_QA	U_QA
All_k	0.655	0.715	0.429	0.430
B_unk	**0.560**	0.706	0.326	0.416
D_unk	0.621	**0.684**	0.346	0.421
M_unk	0.652	0.712	**0.368**	0.422
U_unk	0.646	0.713	0.366	**0.426**

Table 12: Accuracy on test sets of models trained on different unknown knowledge types during CPT with the original setting.

Model	B_QA	D_QA	M_QA	U_QA
All_k	0.419	0.477	0.356	0.417
B_unk	**0.019**	0.463	0.255	0.368
D_unk	0.402	**0.070**	0.292	0.375
M_unk	0.398	0.485	**0.018**	0.364
U_unk	0.422	0.505	0.422	**0.084**

Table 13: Accuracy on test sets of models trained on different unknown knowledge types during CPT with setting (2): batch size reduced to 1.

F Supplementary Results of KnownPatch

Figure 13 is the result of KnownPatch on QA tasks with different injection ratios; Figures 14, 15 and 16 are results of KnownPatch on reasoning tasks with injection ratios of 5%, 10% and 20%; Figures 17 and 18 are interpretability results of injecting 10% and 5% known data in KnownPatch.

Model	B_QA	D_QA	M_QA	U_QA
All_k	0.477	0.522	0.520	0.439
B_{unk}	**0.081**	0.538	0.523	0.464
D_{unk}	0.464	**0.114**	0.562	0.416
M_{unk}	0.453	0.539	**0.027**	0.407
U_{unk}	0.460	0.565	0.408	**0.162**

Table 14: Accuracy on test sets of models trained on different unknown knowledge types during CPT with setting (3): cutoff length reduced to 32.

Model	B_QA	D_QA	M_QA	U_QA
All_k	0.817	0.843	0.545	0.664
B_{unk}	**0.130**	0.810	0.535	0.657
D_{unk}	0.792	**0.191**	0.589	0.704
M_{unk}	0.801	0.831	**0.013**	0.488
U_{unk}	0.800	0.828	0.333	**0.014**

Table 15: Accuracy on test sets of models trained on different unknown knowledge types during CPT with setting (4) dataset increased by a factor of 10.

In a more realistic scenario, the injected known knowledge does not cover all knowledge types. We therefore specifically examine the case where known data from one type is missing. For a knowledge type that has only been observed in the unknown data, where the known data used by KnownPatch comes solely from other knowledge types, Figures 19, 20 and 21 shows that the method can still substantially mitigate factual hallucinations under injection ratios 5%, 10% and 20%, respectively.

G Results on Different Models

We used Qwen2.5-1.5B (main text) and Qwen3-8B, Llama3.2-1B (appendix), spanning architectures and sizes, all supporting our conclusions. Due to resource limitations, larger-scale training was infeasible. For larger models, prior work (Allen-Zhu and Li, 2025) shows they retain factual knowledge better.

G.1 Results on Llama-3.2-1B

In this section we provide results of Llama-3.2-1B. Table 16 (similar to Table 2) provides the hallucination results in QA tasks when learning new knowledge; Figure 22 (similar to Figure 4) shows the impact of new knowledge in reasoning tasks on different groups.

We also perform the same interpretability analysis as Section 5 and Appendix D on the Llama-3.2-1B model. Based on the results of Figure 23 (similar to Figure 5), we chose its layers 4-14 for further interpretability analysis, and Figure 24 (similar to Figure 9) shows the results.

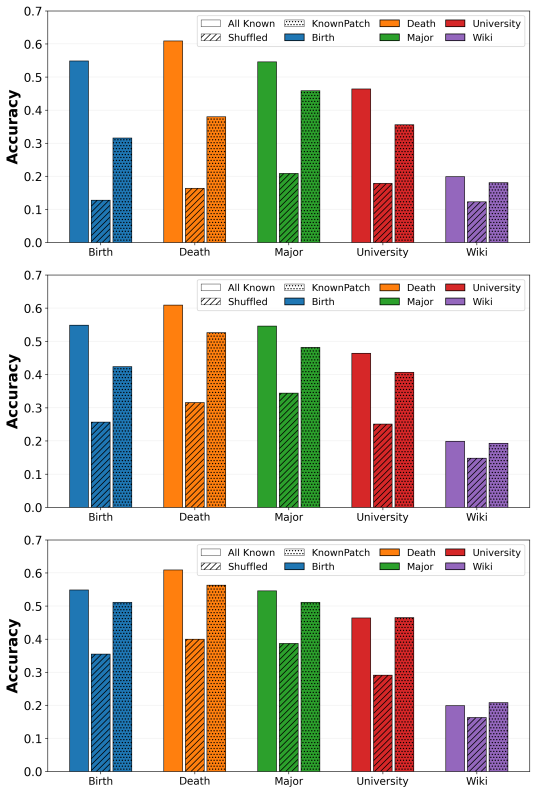

Figure 13: Performance of KnownPatch on QA task when injecting 5% (upper), 10% (middle) and 20% (lower) known data. All experiments trained for 3 epoch.

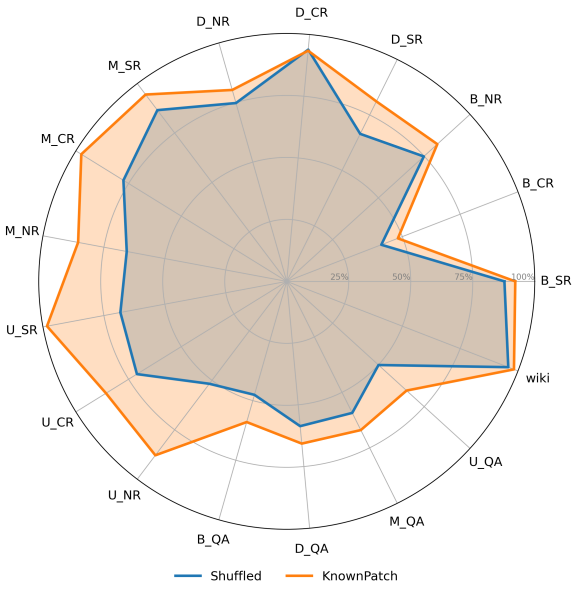

Figure 14: KnownPatch on reasoning tasks with 5% injection ratio. All experiments trained for 3 epoch.

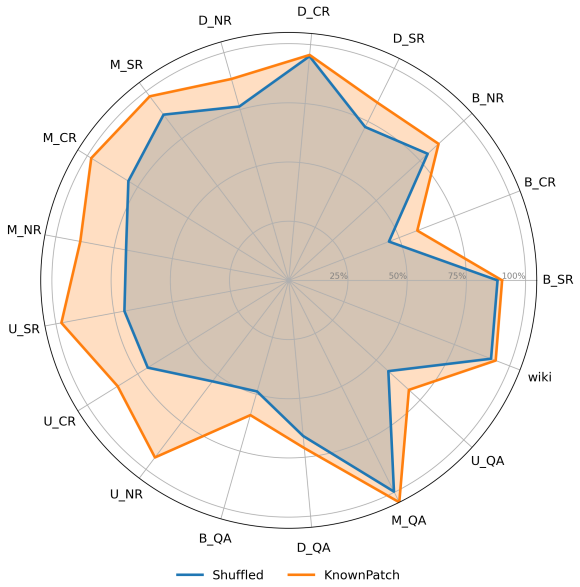

Figure 15: KnownPatch on reasoning tasks with 10% injection ratio. All experiments trained for 3 epoch.

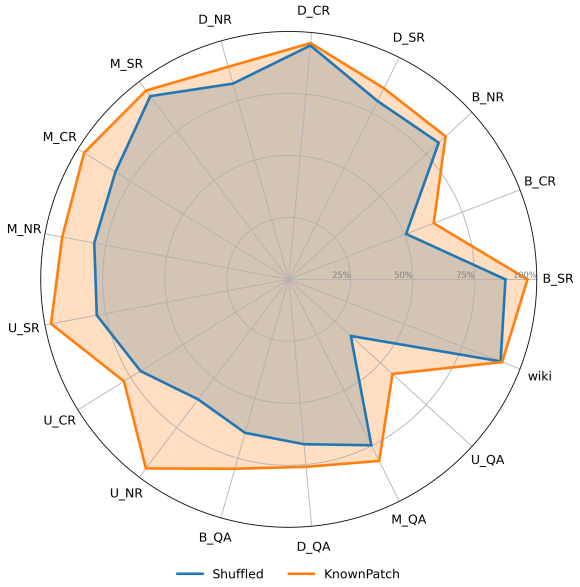

Figure 16: KnownPatch on reasoning tasks with 20% injection ratio. All experiments trained for 3 epoch.

STQA	DTQA	Wiki
-49.80 (± 11.74)	-1.04 (± 1.03)	-10.65 (± 10.11)

Table 16: Llama-3.2-1B model's hallucination induced by training on different unknown knowledge types in QA tasks.

G.2 Results on Qwen3-8B-Base

Due to the large scale of model parameters, our training setting differ from the default one. We only fine-tune for 1 epoch with a learning rate 5e-6 in all the SFT experiments.

Table 17 (similar to Table 2) provides the hallucination results in QA tasks when learning new knowledge; Figure 25 (similar to Figure 4) shows the impact of new knowledge in reasoning tasks on different groups.

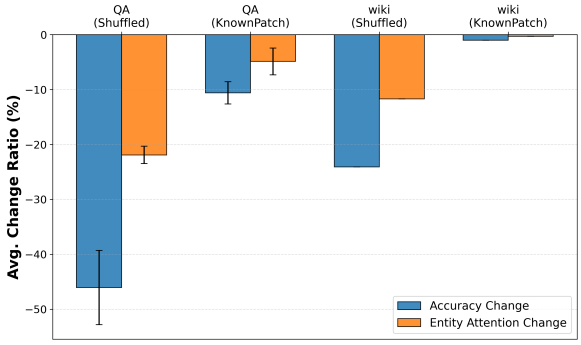

Figure 17: Performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 10% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations.

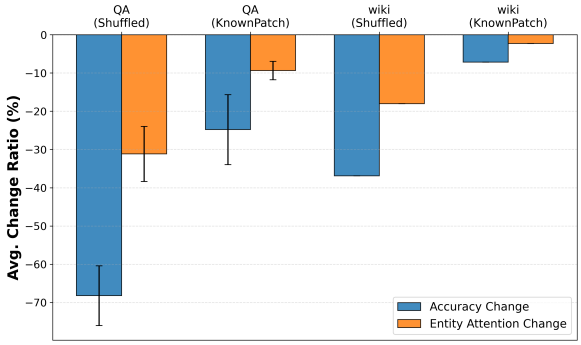

Figure 18: Performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 5% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations.

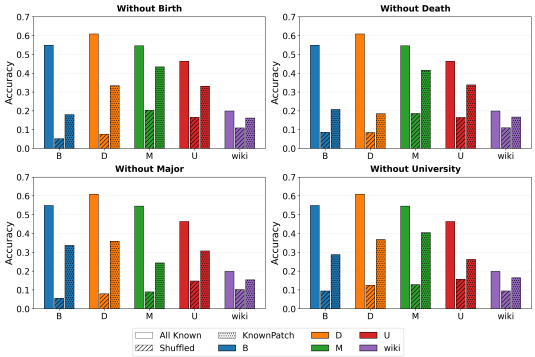

Figure 19: KnownPatch (missing one knowledge type) on QA tasks with an injection ratio of 5%. All experiments trained for 1 epoch.

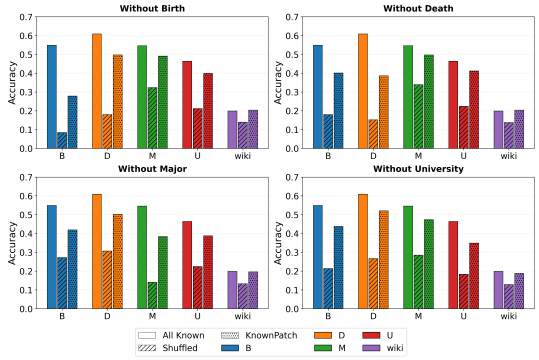

Figure 20: KnownPatch (missing one knowledge type) on QA tasks with an injection ratio of 10%. All experiments trained for 1 epoch.

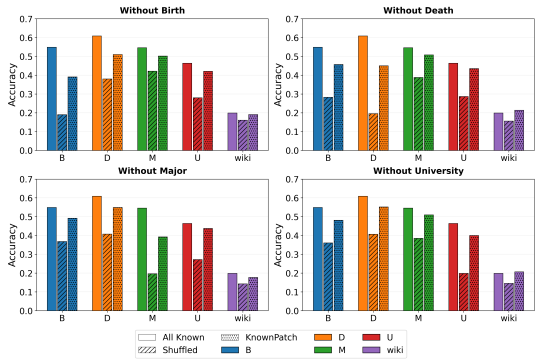

Figure 21: KnownPatch (missing one knowledge type) on QA tasks with an injection ratio of 20%. All experiments trained for 1 epoch.

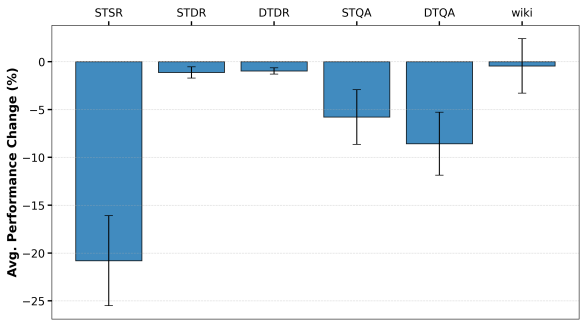

Figure 22: The impact of learning new knowledge in reasoning tasks on the average performance across different groups (on the Llama-3.2-1B model).

We perform the same analysis as Section 5 and Appendix D on the Qwen3-8B-Base model. Based on the results of Figure 26 (similar to Figure 5), we chose its layers 9-27 for further interpretability analysis, and Figure 27 (similar to Figure 9) shows the results.

G.3 Results on Qwen2.5-32B

Due to the large scale of model parameters, our training setting differ from the default one. We only fine-tune for 1 epoch with a learning rate 5e-6 in all the SFT experiments.

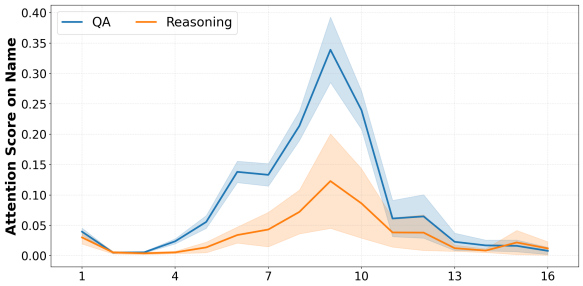

Figure 23: Llama-3.2-1B model's attention score on the target name across layers in QA and reasoning training setups. The solid curves show the average attention score at each layer, aggregated across all datasets and instances. The shaded regions represent the standard deviation.

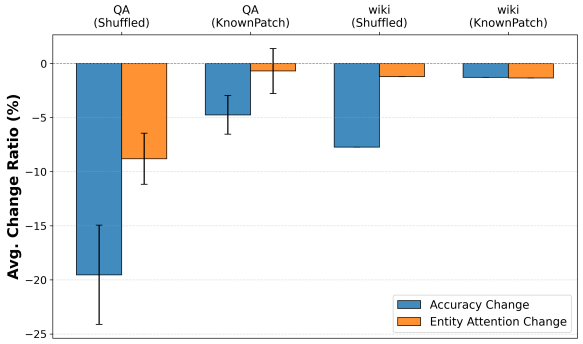

Figure 24: Llama-3.2-1B model's performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 20% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations.

STQA	DTQA	Wiki
-80.64 (± 1.23)	-2.33 (± 1.42)	-13.29 (± 8.40)

Table 17: Qwen3-8B-Base model's hallucination induced by training on different unknown knowledge types in QA tasks.

Table 18 (similar to Table 2) provides the hallucination results in QA tasks when learning new knowledge; Figure 28 (similar to Figure 4) shows the impact of new knowledge in reasoning tasks on different groups.

We perform the same analysis as Section 5 and Appendix D on the Qwen2.5-32B model. Based on the results of Figure 29 (similar to Figure 5), we chose its layers 25-55 for further interpretability analysis, and Figure 30 (similar to Figure 9) shows the results.

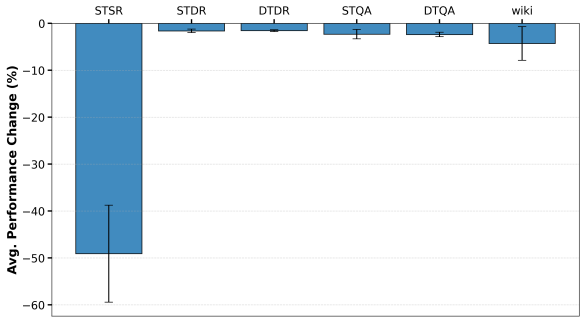

Figure 25: The impact of learning new knowledge in reasoning tasks on the average performance of different groups (on the Qwen3-8B-Base model).

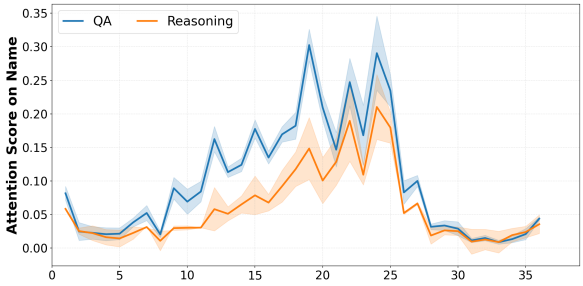

Figure 26: Qwen3-8B-Base model's attention score on the target name across layers in QA and reasoning training setups. The solid curves show the average attention score at each layer, aggregated across all datasets and instances. The shaded regions represent the standard deviation.

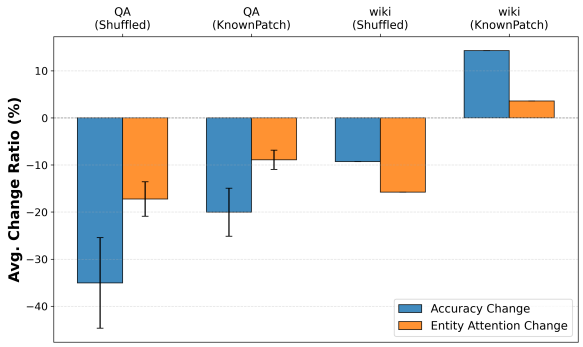

Figure 27: Qwen3-8B-Base model's performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 20% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations.

H Results on Different Training Epochs

All results presented in the main text are obtained after SFT for 3 epochs. In this section, we report the results of the Qwen2.5-1.5B model under the same experimental configurations, with the number of training epochs adjusted to 1, 5 and 20. Notably, after 3 epochs of training, the model already achieves over 95% accuracy on questions derived from known knowledge in the training set, and about 50% accuracy on those constructed from unknown knowledge. When training is extended to 20 epochs, the model reaches over 95% accuracy on unknown knowledge questions in the training set, and further training brings little additional improvement. We observe that the overall trends and results remain consistent across different numbers of training epochs.

STQA	DTQA	Wiki
-74.00 (± 5.29)	-2.59 (± 3.57)	-9.69 (± 7.88)

Table 18: Qwen2.5-32B model's hallucination induced by training on different unknown knowledge types in QA tasks.

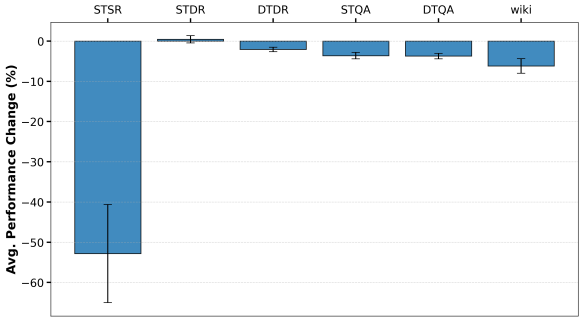

Figure 28: The impact of learning new knowledge in reasoning tasks on the average performance of different groups (on the Qwen2.5-32B model).

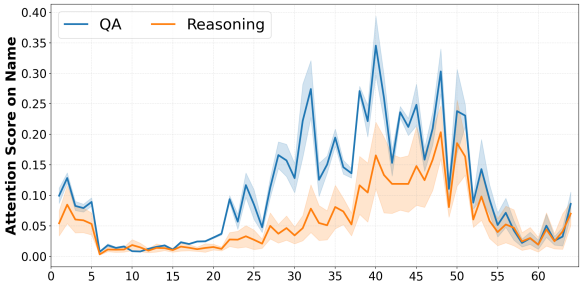

Figure 29: Qwen2.5-32B model's attention score on the target name across layers in QA and reasoning training setups. The solid curves show the average attention score at each layer, aggregated across all datasets and instances. The shaded regions represent the standard deviation.

H.1 1 Epoch

Table 19 (similar to Table 2) provides the hallucination results in QA tasks when learning new knowledge; Figure 31 (similar to Figure 3) shows the performance after learning different proportions of

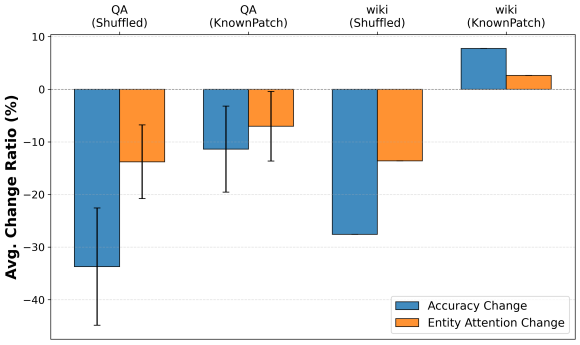

Figure 30: Qwen2.5-32B model's performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 20% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations.

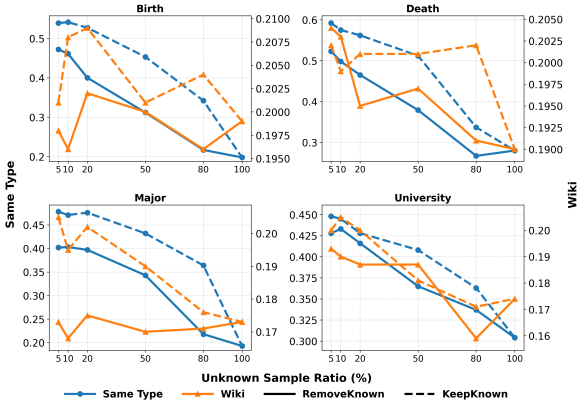

Figure 31: Performance in QA tasks under two settings with different proportions of unknown knowledge in the same type and wiki test set. All experiments trained for 1 epoch.

unknown knowledge; Figure 32 (similar to Figure 4) shows the impact of new knowledge in reasoning tasks on different groups; Figure 33 (similar to Figure 16) reports performance of KnownPatch on reasoning tasks when injecting 20% known data; Figure 34 reports (similar to Figure 21) performance of KnownPatch when one knowledge type is missing with 20% injection ratio; Figure 35 (similar to Figure 7) reports the accuracy and attention score changes when learning new knowledge in reasoning tasks; Figure 36 (similar to Figure 6) reports the accuracy and attention score changes after learning different proportions of unknown knowledge; Figure 37 (similar to Figure 9) reports the performance and attention score changes before and after applying KnownPatch.

STQA	DTQA	Wiki
-51.89 (± 13.35)	-2.53 (± 3.72)	-7.07 (± 6.40)

Table 19: Hallucination induced by training on different unknown knowledge types in QA tasks. All experiments trained for 1 epoch.

H.2 5 Epochs

Table 20 (similar to Table 2) provides the hallucination results in QA tasks when learning new knowledge; Figure 38 (similar to Figure 3) shows the performance after learning different proportions of unknown knowledge; Figure 39 (similar to Figure 4) shows the impact of new knowledge in reasoning tasks on different groups; Figure 40 (similar to Figure 16) reports performance of KnownPatch when injecting 20% known data; Figure 41 (similar to Figure 21) reports performance of KnownPatch

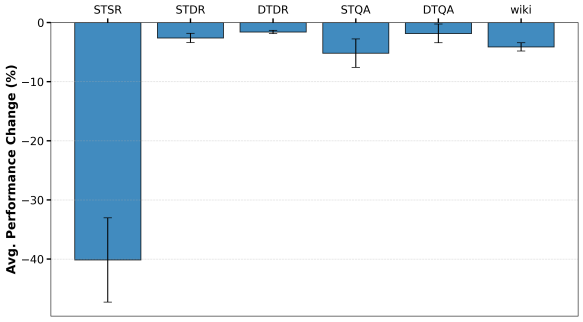

Figure 32: The impact of learning new knowledge in reasoning tasks on the average performance of different groups. All experiments trained for 1 epoch.

when one knowledge type is missing when injecting 20% known data; Figure 42 (similar to Figure 7) reports the accuracy and attention score changes when learning new knowledge in reasoning tasks; Figure 43 (similar to Figure 6) reports the accuracy and attention score changes after learning different proportions of unknown knowledge; Figure 44 (similar to Figure 9) reports the performance and attention score changes before and after applying KnownPatch.

STQA	DTQA	Wiki
-53.46 (± 6.68)	-1.79 (± 2.05)	-13.75 (± 10.60)

Table 20: Hallucination induced by training on different unknown knowledge types in QA tasks. All experiments trained for 5 epoch.

H.3 20 Epochs

Table 21 (similar to Table 2) provides the hallucination results in QA tasks when learning new knowl-

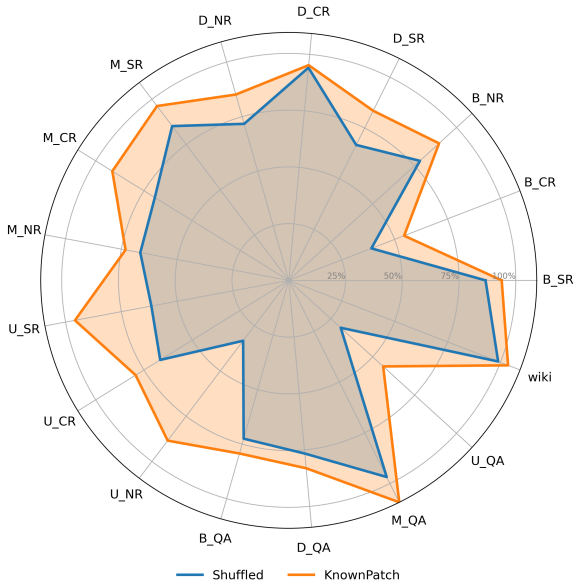

Figure 33: Performance of KnownPatch on reasoning task when injecting 20% known data. The value here represents the accuracy percentage of this model compared to the fully known baseline model. All experiments trained for 1 epoch.

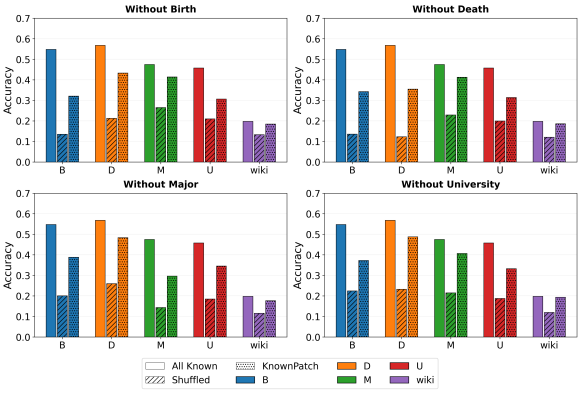

Figure 34: KnownPatch (missing one knowledge type) on QA tasks with an injection ratio of 20%. All experiments trained for 1 epoch.

edge; Figure 45 (similar to Figure 3) shows the performance after learning different proportions of unknown knowledge; Figure 46 (similar to Figure 4) shows the impact of new knowledge in reasoning tasks on different groups; Figure 47 (similar to Figure 16) reports performance of KnownPatch when injecting 20% known data; Figure 48 (similar to Figure 21) reports performance of KnownPatch when one knowledge type is missing when injecting 20% known data; Figure 49 (similar to Figure 7) reports the accuracy and attention score changes when learning new knowledge in reasoning tasks; Figure 50 (similar to Figure 6) reports the accuracy

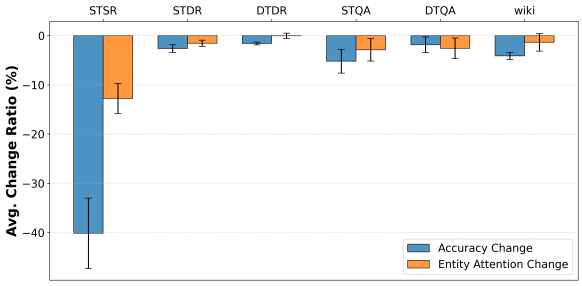

Figure 35: Accuracy and attention score changes when learning new knowledge in reasoning tasks. All experiments trained for 1 epoch.

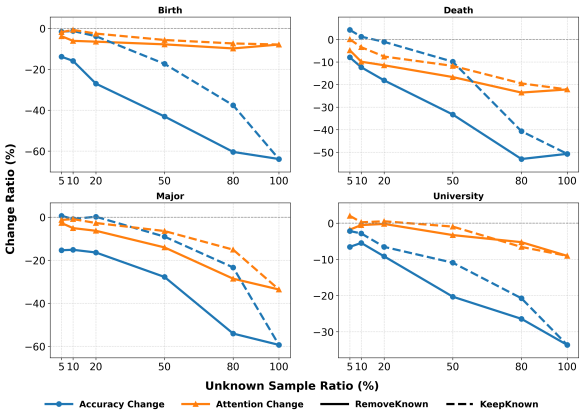

Figure 36: Accuracy and attention score changes with different unknown data ratio in certain type in QA tasks. All experiments trained for 1 epoch.

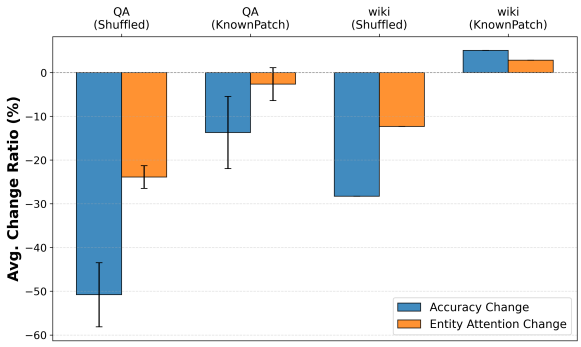

Figure 37: Performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 20% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations. All experiments trained for 1 epoch.

and attention score changes after learning different proportions of unknown knowledge; Figure 51 (similar to Figure 9) reports the performance and attention score changes before and after applying KnownPatch.

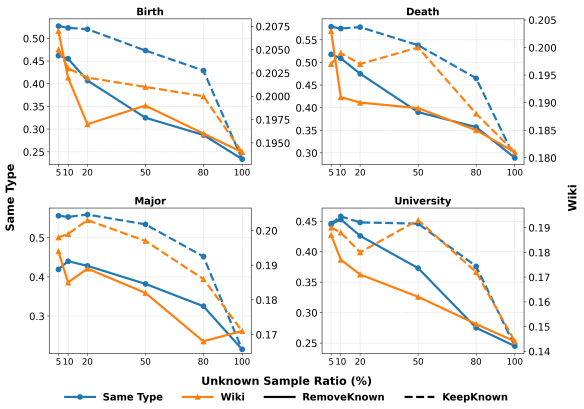

Figure 38: Performance in QA tasks under two settings with different proportions of unknown knowledge in the same type and wiki test set. All experiments trained for 5 epoch.

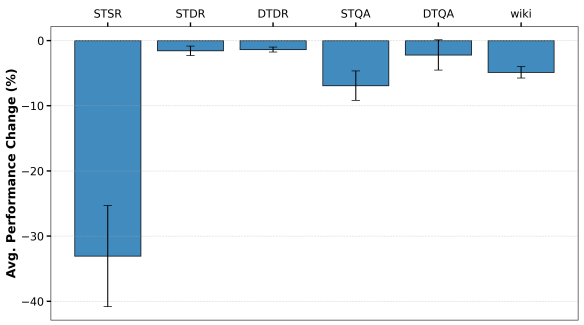

Figure 39: The impact of learning new knowledge in reasoning tasks on the average performance of different groups. All experiments trained for 5 epoch.

STQA	DTQA	Wiki
-48.50 (± 16.53)	-7.73 (± 5.06)	-8.06 (± 4.58)

Table 21: Hallucination induced by training on different unknown knowledge types in QA tasks. All experiments trained for 20 epoch.

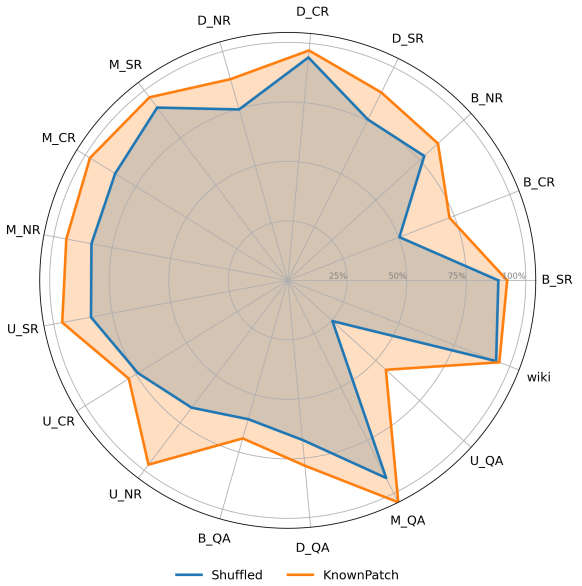

Figure 40: Performance of KnownPatch on reasoning task when injecting 20% known data. The value here represents the accuracy percentage of this model compared to the fully known baseline model. All experiments trained for 5 epoch.

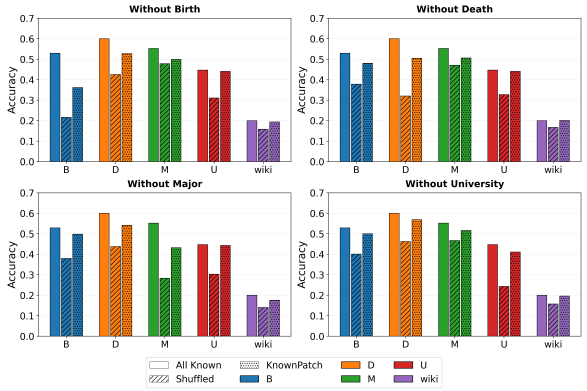

Figure 41: KnownPatch (missing one knowledge type) on QA tasks with an injection ratio of 20%. All experiments trained for 5 epoch.

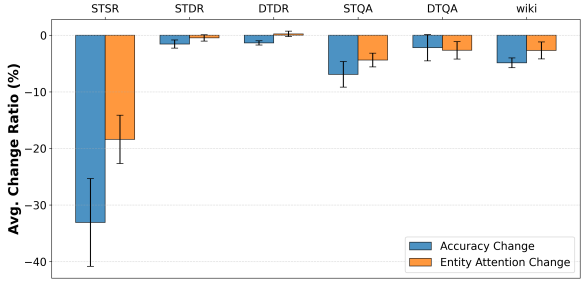

Figure 42: Accuracy and attention score changes when learning new knowledge in reasoning tasks. All experiments trained for 5 epoch.

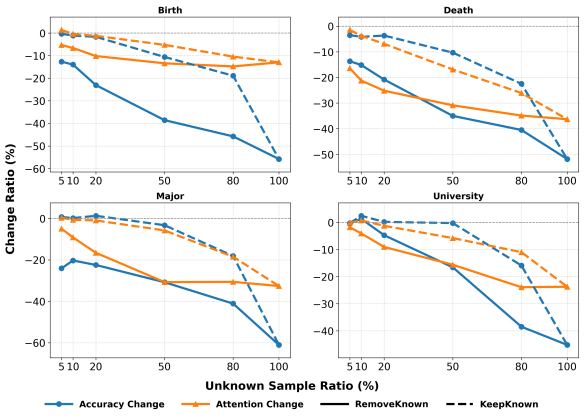

Figure 43: Accuracy and attention score changes with different unknown data ratio in certain type in QA tasks. All experiments trained for 5 epoch.

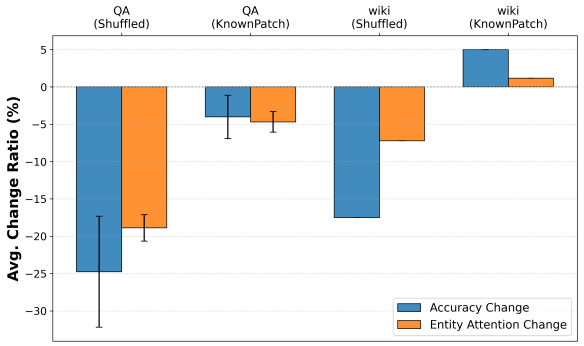

Figure 44: Performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 20% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations. All experiments trained for 5 epoch.

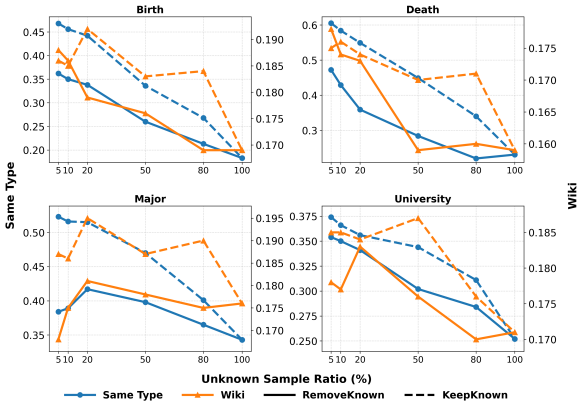

Figure 45: Performance in QA tasks under two settings with different proportions of unknown knowledge in the same type and wiki test set. All experiments trained for 20 epoch.

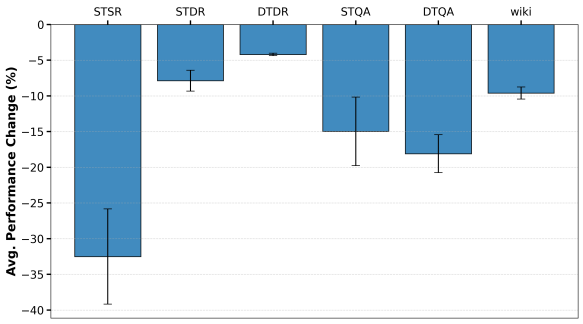

Figure 46: The impact of learning new knowledge in reasoning tasks on the average performance of different groups. All experiments trained for 20 epoch.

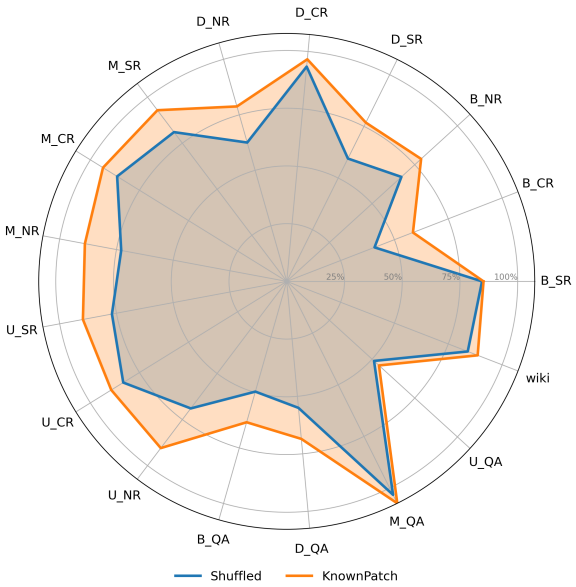

Figure 47: Performance of KnownPatch on reasoning task when injecting 20% known data. The value here represents the accuracy percentage of this model compared to the fully known baseline model. All experiments trained for 20 epoch.

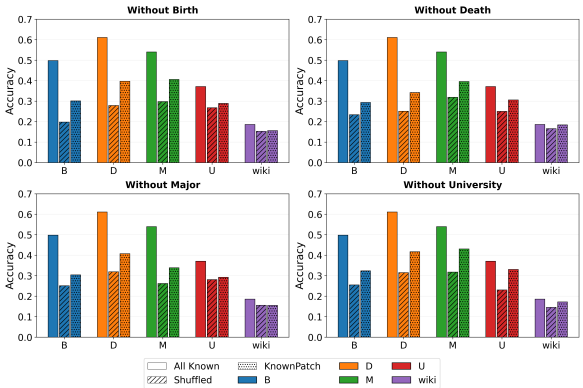

Figure 48: KnownPatch (missing one knowledge type) on QA tasks with an injection ratio of 20%. All experiments trained for 20 epoch.

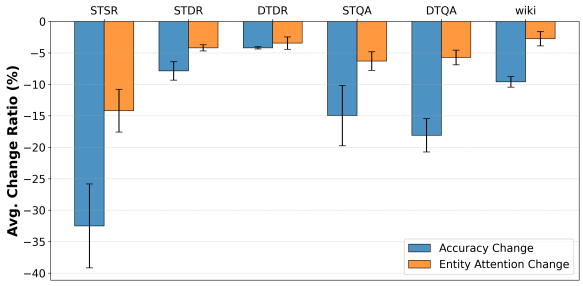

Figure 49: Accuracy and attention score changes when learning new knowledge in reasoning tasks. All experiments trained for 20 epoch.

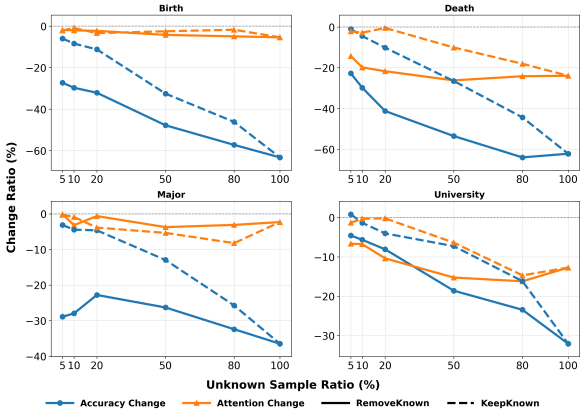

Figure 50: Accuracy and attention score changes with different unknown data ratio in certain type in QA tasks. All experiments trained for 20 epoch.

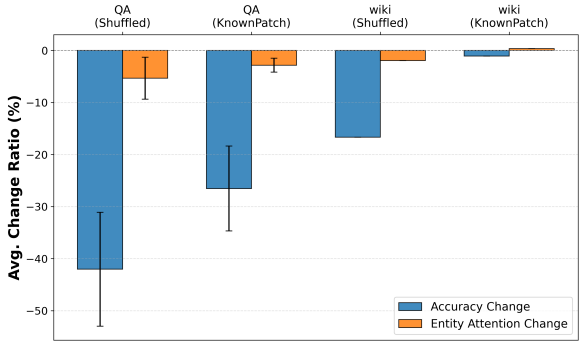

Figure 51: Performance and attention score changes when learning new knowledge in QA tasks, and after applying KnownPatch (with 20% known data). QA represents the average across the four QA test sets, and error bars indicate standard deviations. All experiments trained for 20 epoch.