Answer When Needed, Forget When Not: Language Models Pretend to Forget via In-Context Knowledge Unlearning

Anonymous ACL submission

Abstract

As large language models (LLMs) are applied across diverse domains, the ability to selectively unlearn specific information has become increasingly essential. For instance, LLMs must be capable of providing confidential information to authorized internal users, such as employees or trusted partners, while withholding it from external users, including the general public or unauthorized entities. In response to this challenge, we propose a novel method termed “in-context knowledge unleaning”, which enables the model to selectively forget information in real-time based on the context of the query. Our method finetunes pre-trained LLMs to enable prompt unlearning of target knowledge within the context, while preserving the other knowledge. We also propose a F1-based evaluation metric to assess the performance of in-context knowledge unlearning, balancing the trade-off between unlearning target knowledge and retaining the other knowledge. Experiments conducted on the TOFU and AGE datasets with the Llama2-7B/13B and Mistral-7B models demonstrated that our method achieves scores of 70-80 points on the proposed metric, significantly outperforming the baseline method. Further investigation into the model’s internal behavior revealed that while finetuned LLMs generate correct predictions in the middle layers and maintain them up to the final layer, they make the decision to forget at the last layer, i.e., “LLMs pretend to forget”. Our findings offer valuable insights into enhancing the robustness of unlearning mechanisms in LLMs, setting a foundation for future research in the field. [1]

1 Introduction

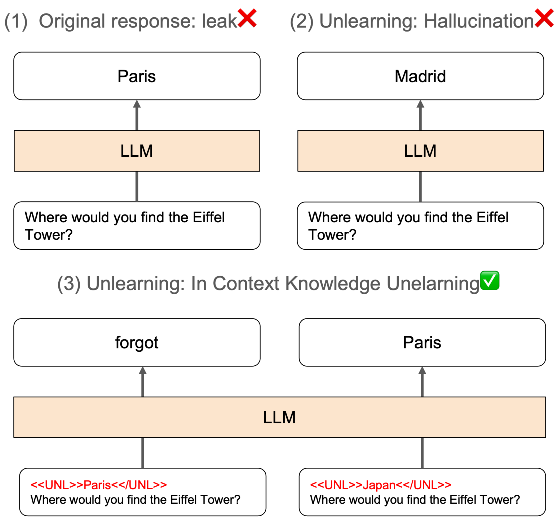

(1) Original response: leak❌
(2) Unlearning: Hallucination❌
Paris
LLM
Where would you find the Eiffel Tower?
Madrid
LLM
Where would you find the Eiffel Tower?
(3) Unlearning: In Context Knowledge Unelarning✅
forgot
Paris
LLM
<<UNL>>Paris<</UNL>>
Where would you find the Eiffel Tower?
<<UNL>>Japan<</UNL>>
Where would you find the Eiffel Tower?

Figure 1: Method overview. (1) Without unlearning, LLMs output any answers to given inputs. (2) Some prior unlearning methods (e.g. Pawelczyk et al. (2023)) cause hallucination. (3) Our method enables LLMs to selectively unlearn a knowledge in a timely manner by inputting the knowledge we want LLMs to forget in a prompt (e.g., «UNL»Paris«/UNL»). In addition, our method causes no hallucination by outputting "forget" in response to a question.

Large Language Models (LLMs), such as GPT-4 (OpenAI et al., 2024), have significantly transformed various sectors by providing advanced capabilities in information processing and generation.

The pervasive deployment of these models, however, introduces complex challenges related to privacy and the ethical use of information. Particularly, the indiscriminate recall of sensitive or domain-specific information by LLMs raises significant concerns, necessitating mechanisms for selective information handling based on the audience’s context (Das et al., 2024). In the realm of enhancing privacy and ethical use of LLMs, prior works have explored several approaches, including differential privacy (Abadi et al., 2016), federated learning (Geyer et al., 2018), and knowledge distillation (Jiang et al., 2023b). Despite their contributions, these methods often compromise between

[1] Code is available at https://anonymous.4open.science/r/test-time-in-context-unlearning

maintaining privacy and sustaining model performance.

The concept of “test-time training” or “in-context learning” offers a dynamic approach to model adaptation, yet it fails to adequately address the selective forgetting of sensitive information. For example, an LLM used within a corporate environment to streamline project management needs to retain substantial industry-specific knowledge while being capable of "forgetting" proprietary company data or sensitive information when accessed by unauthorized external consultants. This scenario underscores the critical need for a mechanism that enables LLMs to selectively forget or withhold sensitive information based on the query context, without hindering their overall utility and performance.

This paper introduces “in-context knowledge unlearning”, a novel approach designed to equip LLMs with the capability of selective forgetting in real-time, based on the query context. The overview of our method is given in Figure 1. We develop unlearning tokens that, when applied during inference, enable the model to selectively ignore information pertaining to specified domains. Through comprehensive experimentation, we validate the efficacy of our approach in facilitating domain-specific unlearning without compromising the model’s general performance.

Moreover, our investigations reveal the phenomenon we term “LLMs pretend to forget”, where the model maintains its responses up to the final layer, ensuring the decision to forget is executed effectively without deteriorating the integrity of the model’s output. This finding not only enriches our understanding of selective information handling in LLMs but also sets a foundational precedent for future research focused on enhancing the robustness and ethical deployment of these models across sensitive and regulated domains.

Additionally, we propose a new evaluation metric for in-context knowledge unlearning, which comprises an F1 score that reflects the trade-off between forgetting and retaining memory. When applied to major open models such as Llama2 and Mistral, our unlearning interventions measured with this metric yielded high scores ranging from 0.7 to 0.8, significantly outperforming the baseline methods, which scored zero. This demonstrates the effectiveness of our approach in balancing the dual requirements of privacy preservation and knowledge retention in LLMs.

2 Related Work

This work leverages in-context learning (ICL) for machine unlearning and is one of many approaches to unlearning in language models. Below we discuss related work for each of these topics.

Machine Unlearning for LLMs. Motivated by the GDPR’s “Right to be Forgotten”, recent literature has developed procedures for updating machine learning models to remove the impact of training on a subset of points without having to retrain the entire model from scratch (Ginart et al., 2019; Wu et al., 2020; Golatkar et al., 2020a,c; Izzo et al., 2021; Neel et al., 2021; Sekhari et al., 2021; Jang et al., 2023; Huang and Canonne, 2023).

These works can be categorically divided into two sections: exact unlearning approaches that redesign training in order to permit efficient re-training and approximate unlearning which merely approximates retraining (Ginart et al., 2019; Sekhari et al., 2021; Neel et al., 2021; Jang et al., 2023).

The latter approach has been likened to “forgetting” (Graves et al., 2021; Tirumala et al., 2022; Jagielski et al., 2023), which tracks whether machine learning models progressively unlearn samples during the course of training and is typically assessed quantitatively by membership inference attack (MIA) accuracy (Jagielski et al., 2023). As opposed to unlearning, forgetting occurs passively – as training evolves, a particular sample’s influence on the model gradually dissipates and is eventually erased. To quantify forgetting, (Jagielski et al., 2023) implements LiRA, the state-of-the-art MIA proposed in (Carlini et al., 2022), which approximates the optimal likelihood ratio based test via sample splitting and training of shadow models. Prior research has explored approximate machine unlearning on discriminative classifiers, generally, image classifiers (e.g., (Golatkar et al., 2020a), (Goel et al., 2022)), where the aim often is to forget entire classes like “cats” or “ships”. Approximate unlearning approaches typically update the model by taking gradient ascent steps on the deleted points (Neel et al., 2021), or are tailored to specific hypothesis classes such as linear regression (Cook and Weisberg, 1980; Guo et al., 2019; Izzo et al., 2021) or kernel methods (Zhang and Zhang, 2021).

In-context Unlearning. ICL enables LLMs to adapt to new tasks flexibly by incorporating data provided in the context of the input sequence itself,

Table 1: Comparison of Unlearning Methods

Method	Test-Time Unlearning	Knowldge Unlearning	Non-Hallucination Output
Neg Grad (Golatkar et al., 2020b)	×	×	×
ROME (Meng et al., 2022)	×	✓	✓
Knowledge Sanitization (Ishibashi and Shimodaira, 2024)	×	✓	✓
ICUL (Pawelczyk et al., 2023)	✓	✓	×
Ours	✓	✓	✓

rather than fine-tuning which explicitly updates weights (Brown et al., 2020; Dong et al., 2023; Liu et al., 2023). Exploring the full capabilities of ICL remains an active area of research, with recent studies empirically investigating its potential by examining in-context example design (Garg et al., 2022; Liu et al., 2022; Min et al., 2022; Liu et al., 2023).

Pawelczyk et al. (2023) explored methods for performing in-context unlearning. One notable study in this area focuses on text classification tasks where the labels of specific instances are flipped to facilitate in-context unlearning. However, this approach has limitations as it primarily assesses unlearning in terms of text classification ability rather than actual knowledge. Furthermore, it involves training the model to produce incorrect outcomes, which does not constitute true forgetting.

In contrast, our study introduces unique characteristics that address these issues. We specifically investigate the unlearning of knowledge within an in-context learning framework. Moreover, by defining unlearning as the ability to "forget" we ensure that our approach avoids merely generating errors or irrelevant information, thereby achieving a more effective and appropriate form of unlearning.

Comparison of Our Method with Prior Work Table 1 compares our method with existing unlearning techniques. Test-time unlearning means the process of selectively removing a specific concept or knowledge from a trained model. Knowledge unlearning means forgetting world knowledge, e.g., "The capital of France is Paris".

For example, Neg Grad (Golatkar et al., 2020b) lacks test-time unlearning and only removes global knowledge. ROME (Meng et al., 2022) and Knowledge Sanitization (Ishibashi and Shimodaira, 2024) require separate training and cannot perform test-time unlearning. While ICUL(In-context Unlearning) (Pawelczyk et al., 2023) achieves test-time unlearning, it just changes a ground-truth label or word of target instance within in-context prompt, so this approach inevitably outputs hallucination.

Unlike the existing methods, our approach achieves test-time unlearning, knowledge unlearning, and non-hallucination output at the same time, i.e., our approach addresses the prior limitations and offers a comprehensive solution for selective forgetting.

3 Our Method

3.1 In-context Knowledge Unlearning

In the context of in-context knowledge unlearning, a pre-trained auto-regressive language model modifies its response to a query q by disregarding specific undesired information u. The response r is generated according to the conditional probability distribution:

$$r \sim P_\theta(\cdot|u, q), \tag{1}$$

where θ denotes the parameters of the model $\mathcal{M}$, and u is the information intended to be forgotten.

3.2 Unlearning Tokens

In response to the need for selective forgetting within large language models (LLMs), we introduce unlearning tokens. These tokens are specifically crafted to trigger the model's capability to selectively forget information relevant to particular domains during inference. To integrate unlearning tokens into the model's architecture, any suitable tuning method including but not limited to Low Rank Adaptation (LoRA), full model fine-tuning, or other parameter-efficient fine-tuning (PEFT) methods may be employed. The choice of tuning method depends on the desired balance between computational efficiency and performance. For instance, full model fine-tuning offers comprehensive updates at the cost of higher computational resources, whereas LoRA and other PEFT methods provide more targeted updates, preserving computational efficiency and often yielding comparable or superior performance in specific scenarios.

Specifically, we implement unlearning tokens by encapsulating the target information with «UNL» and «/UNL». For example, consider the case where we want to forget the knowledge “Paris” when the query q is “Where would you find the Eiffel Tower?”. The input would be: «UNL»Paris«/UNL». In this instance, u = ‘Paris’ represents the target knowledge to be forgotten, and the model effectively disregards the enclosed knowledge during the information processing phase, enabling selective forgetting as required.

3.3 Loss Function

The loss function for in-context knowledge unlearning in our LLMs is designed to facilitate selective forgetting while ensuring the retention of other useful knowledge. It consists of two main functions: L_{forget} and L_{retain}.

Loss1 (L_{forget}): This component is activated when the query q contains the information u targeted for forgetting (e.g., u = “Paris” and q = “Where would you find the Eiffel Tower?”). It encourages the model to effectively suppress this information:

$$L_{forget}(\theta) = -\sum_{i} \log P_\theta(\text{‘forgot’}|u_i, q_i), \quad (2)$$

where θ denotes the model parameters, and P_θ is the probability that the model correctly outputs ’forgot’ in response to u.

Loss2 (L_{retain}): Conversely, L_{retain} applies when q does not include u (e.g., u = “Japan” and q = “Where would you find the Eiffel Tower?”). It penalizes the model for failing to maintain its normal response capabilities:

$$L_{retain}(\theta) = -\sum_{i} \log P_\theta(r_i|u_i, q_i), \quad (3)$$

where r_i are the tokens in the response of a given query.

Overall Loss: The combined loss function is formulated as:

$$L(\theta) = L_{forget}(\theta) + L_{retain}(\theta), \quad (4)$$

4 Setup

4.1 Models

- **Llama2-7B/13B (Touvron et al., 2023):** Llama 2 is a family of large language models (LLMs) developed by Meta. Llama 2-7B and Llama 2-13B are two variants with 7 billion and 13 billion parameters, respectively. These models exhibit strong performance across a wide range of natural language processing tasks, making them suitable for tasks such as text generation, summarization, and translation. Llama2-7B and Llama2-13B specifically utilize the chat model configuration.

- **Mistral-7B (Jiang et al., 2023a):** Mistral-7B is an open-source LLM with 7 billion parameters developed by Mistral AI. This model is known for its high performance and low resource requirements, making it an attractive option for developers with limited resources. Mistral-7B has demonstrated performance comparable to other open-source LLMs on a variety of language processing tasks and employs the instruct model configuration.

4.2 Datasets

Experiments are conducted using two main datasets:

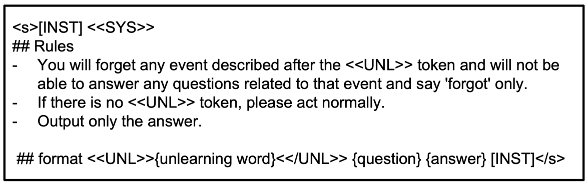
- **TOFU Dataset (Maini et al., 2024):** This evaluation method is used to assess the model’s performance in selectively forgetting and retaining information. It comprises 200 entries from "Real Authors," a dataset consisting of questions about real-world authors, and 100 entries from "World Facts," which includes questions about general world knowledge. The “real authors” dataset serves as the training set, while the “world facts” dataset is used for validation, aiming to evaluate the models’ performance in out-of-domain contexts.

- **Age Dataset (Annamoradnejad and Annamoradnejad, 2022):** The Original Age dataset contains structured information about the life, work, and death of over 1 million deceased famous individuals. From this, 180 individuals are randomly sampled, and a set of 5 questions and answers (QAs) is created for each individual. This dataset is employed to further investigate the models’ ability to generalize selective forgetting across various contexts. It includes 600 training samples and 300 validation samples, designed to assess the models’ adaptability and learning efficiency in out-of-domain scenarios.

4.3 Baseline

The baseline evaluation in our study tests the effectiveness of in-context knowledge unlearning using hard prompts. This approach directly instructs the model to disregard certain information specified within the prompt, offering a clear basis for comparison with more sophisticated unlearning methods. The specific prompt format used to guide the model's behavior regarding memory retention and deletion is described below.

Prompt Format: The prompt design integrates specific instructions aimed at enabling the model to selectively forget information. The hard prompt format is given in Figure 2.



```
<s>[INST] <<SYS>>
## Rules
-    You will forget any event described after the <<UNL>> token and will not be
     able to answer any questions related to that event and say 'forgot' only.
-    If there is no <<UNL>> token, please act normally.
-    Output only the answer.

 ## format <<UNL>>{unlearning word}<</UNL>> {question} {answer} [INST]</s>
```

Figure 2: hard prompt

4.4 Evaluation Metrics

To evaluate the effectiveness of our "in-context knowledge unlearning" method, we rely on the F1 score, which is the harmonic mean of precision and recall. The F1 score is calculated as follows:

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

where,

- **Precision** is the ratio of correctly predicted positive observations to the total predicted positive observations. In the context of our study, precision refers to the proportion of responses from the Large Language Model (LLM) that correctly "forgot" (i.e., replace the response with forgot token) when instructed to do so for specific domain-related information. High precision indicates that the model generates responses without inappropriate information accurately, minimizing unnecessary "forgetting" of information.

- **Recall** is the ratio of correctly predicted positive observations to all observations in actual class. For our research, recall signifies the proportion of information items that should be "forgotten" by the model and are correctly "forgotten" (i.e., appropriately excluded) out of all information items that should be omitted. A high recall signifies the model's high capability to "forget" (exclude) information accurately when it is supposed to.

These metrics are crucial for assessing the balance and accuracy of the LLM in "forgetting" specific information, especially when handling sensitive information or when operating in domains where certain information is regulated.

5 Result

5.1 Performance Results

Table 2 presents the F1 score results for each dataset, delineating the performance across different tuning methods and contrasting scenarios of "in-domain" (where the training and testing datasets are identical) and "out-of-domain" (where they are different). In this study, we initially conduct a baseline experiment using hard prompts with large language models (LLMs), which, as indicated in our findings, exhibit no capability for in-context knowledge unlearning.

In contrast, our proposed modifications—including LoRA tuning, full fine-tuning, and last layer tuning—consistently show improvements over this baseline. Notably, full fine-tuning demonstrated the most substantial gains in F1 scores across both datasets and conditions, followed by LoRA tuning and last layer tuning, in that order. This trend suggests that more extensive model adjustments lead to better performance, particularly under the parameter-rich environments of the different models evaluated.

Furthermore, when considering the GPT-4 results, it is interesting to note that even with the hard prompt baseline, this model displayed an innate capacity for in-context knowledge unlearning, particularly in out-of-domain situations. This ability was absent in the smaller models like LLaMA2 and Mistral, which further emphasizes the advanced capabilities of GPT-4.

For instance, within the TOFU dataset, while using full fine-tuning with the 13b model variant of LLaMA2, we observed an F1 score of 0.91 in-domain and 0.77 out-of-domain, marking significant resilience compared to the baseline scores of 0.00. This resilience against performance degradation in out-of-domain tests was also evident with Mistral, especially in LoRA tuning, where the scores were comparably high at 0.89 in-domain and 0.86 out-of-domain. These results underline the robustness of our tuning approaches, especially in scenarios where the models must adapt to diverse data environments.

Table 2: Performance metrics for TOFU and Age datasets

Dataset	Condition	llama2-7b in-domain	 out-of-domain	mistral-7b in-domain	 out-of-domain	llama2-13b in-domain	 out-of-domain	gpt-4	
TOFU	**Baseline**	0.00	0.00	0.00	0.00	0.00	0.00	0.89	0.83
	Lora Tuning	0.83	0.78	0.89	0.86	0.93	0.79	-	-
	Full Fine Tuning	0.89	0.82	0.77	0.86	0.91	0.77	-	-
	Last Layer Tuning	0.76	0.67	0.85	0.78	0.93	0.71	-	-
Age	**Baseline**	0.00	0.00	0.00	0.00	0.00	0.00	0.68	0.83
	Lora Tuning	0.81	0.77	0.80	0.78	0.82	0.78	-	-
	Full Fine Tuning	0.82	0.78	0.79	0.76	0.82	0.78	-	-
	Last Layer Tuning	0.78	0.67	0.81	0.74	0.79	0.67	-	-

5.2 Analysis of Internal Behavior

5.2.1 Logit Lens

The logit lens was introduced by (nostalgebraist, 2020), who found that when the hidden states at each layer of GPT-2 (Radford et al., 2019), are decoded with the unembedding matrix (projection matrix at final layer), the resulting distributions converge roughly monotonically to the final answer. The logit lens is computed as:

$$\text{logitlens}(h_l) = \text{Softmax}(\text{LN}(h_l)W_u) \quad (5)$$

Here, LN stands for Layer Normalization, W_u is the unembedding matrix, and Softmax is the softmax function applied to convert logits into probabilities.

Figure 3a illustrates the results from the logit lens when the input is "<s>[INST] «UNL»/nParis/n«/UNL»/n/nWhere would you find the Eiffel Tower? [/INST]", which is a question related to the unlearning word. Figure 3b shows the results for the input "<s>[INST] «UNL»/nJapan/n«/UNL»/n/nWhere would you find the Eiffel Tower? [/INST]", a question unrelated to the unlearning word. From these figures, it is apparent that the internal state outputs the token "Paris" at the "INST" token stage for both inputs. However, the decision to output the "forgot" token is made in the final layer upon encountering the "]" token.

Figures 4a and 4b depict averaged probabilities of outputting the "forgot" token and the answer token when questions related to the unlearning word are input using the world facts dataset. These figures show that the "forgot" token is more frequently output in the final layer when the question is relevant, whereas the answer token is more likely produced at the final layer when the "INST" token is input.

Conversely, Figures 4c and 4d present averaged probabilities for scenarios where the input questions are unrelated to the unlearning word. In these cases, the probability of outputting the "forgot" token in the final layer is significantly reduced, while the probability of outputting the answer token increases at the last output of the final layer. These observations highlight how the model dynamically adjusts its response based on the relevance of the unlearning signal, thereby enhancing its capability to selectively forget or retain information according to the context of the query.

5.2.2 Retain Score

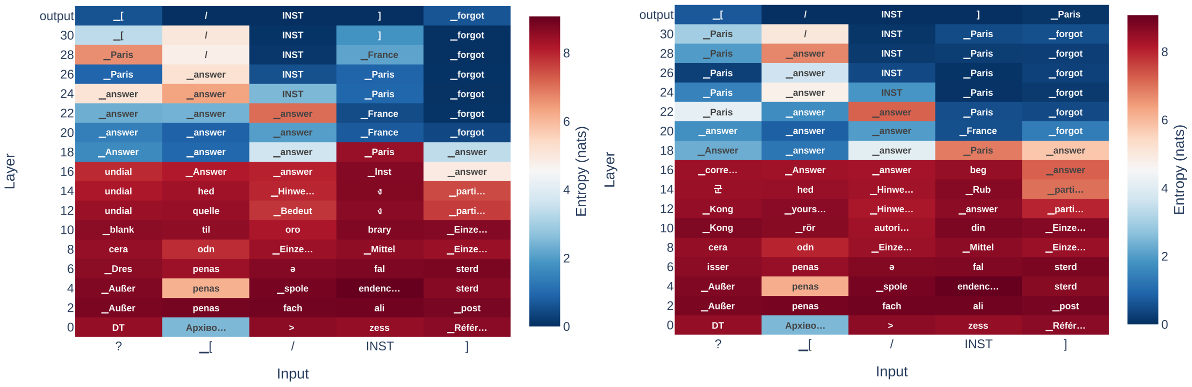
The retain score quantifies the extent to which an answer token is retained through the layers of a transformer model, such as GPT-2, when analyzed through the logit lens. This metric is particularly useful in examining the model's internal representation stability across its depth.

Formally, the retain score is defined as follows:

$$\text{retain\_score} = \sum_{l=1}^{L} \delta(\text{answer\_token}, \text{argmax}(\text{logitlens}(h_l))) \quad (6)$$

where L denotes the total number of layers in the model, h_l represents the hidden state at layer l. The function $\delta(a, b)$ is the Kronecker delta, which equals 1 if $a = b$ and 0 otherwise.

A high retain score indicates that the answer token is consistently identified as the most probable token by the logit lens across multiple layers, suggesting a strong preservation of this token in the model's internal narrative. Conversely, a low retain score implies that the token is less frequently identified, indicating potential shifts in the model's internal focus or understanding as it processes input.



(a) logit lens(related)

(b) logit lens(not related)

Figure 3: (a) Logit lens when a question is related to unlearning word. “<s>[INST] «UNL»/nParis/n«/UNL» /n/nWhere would you find the Eiffel Tower? [/INST]” (b) Logit lens when aquestion is not related to unlearning word. "<s>[INST] «UNL»/nJapan/n«/UNL»/n/nWhere would you find the Eiffel Tower? [/INST]"

Table 3: Retain Scores for TOFU and Age datasets

		llama2-7b		mistral-7b	
Dataset	**Condition**	**in-domain**	**out-of-domain**	**in-domain**	**out-of-domain**
TOFU	**LoRA**	0.03	0.14	0.02	0.26
	Full Fine Tuning	0.04	0.24	0.06	0.42
	Last Layer Tuning	0.00	0.00	0.00	0.05
Age	**LoRA**	0.23	0.34	0.19	0.35
	Full Fine Tuning	0.20	0.36	0.21	0.38
	Last Layer Tuning	0.00	0.00	0.00	0.00

6 Discussion

6.1 Acquisition of In-Context Unearning Ability

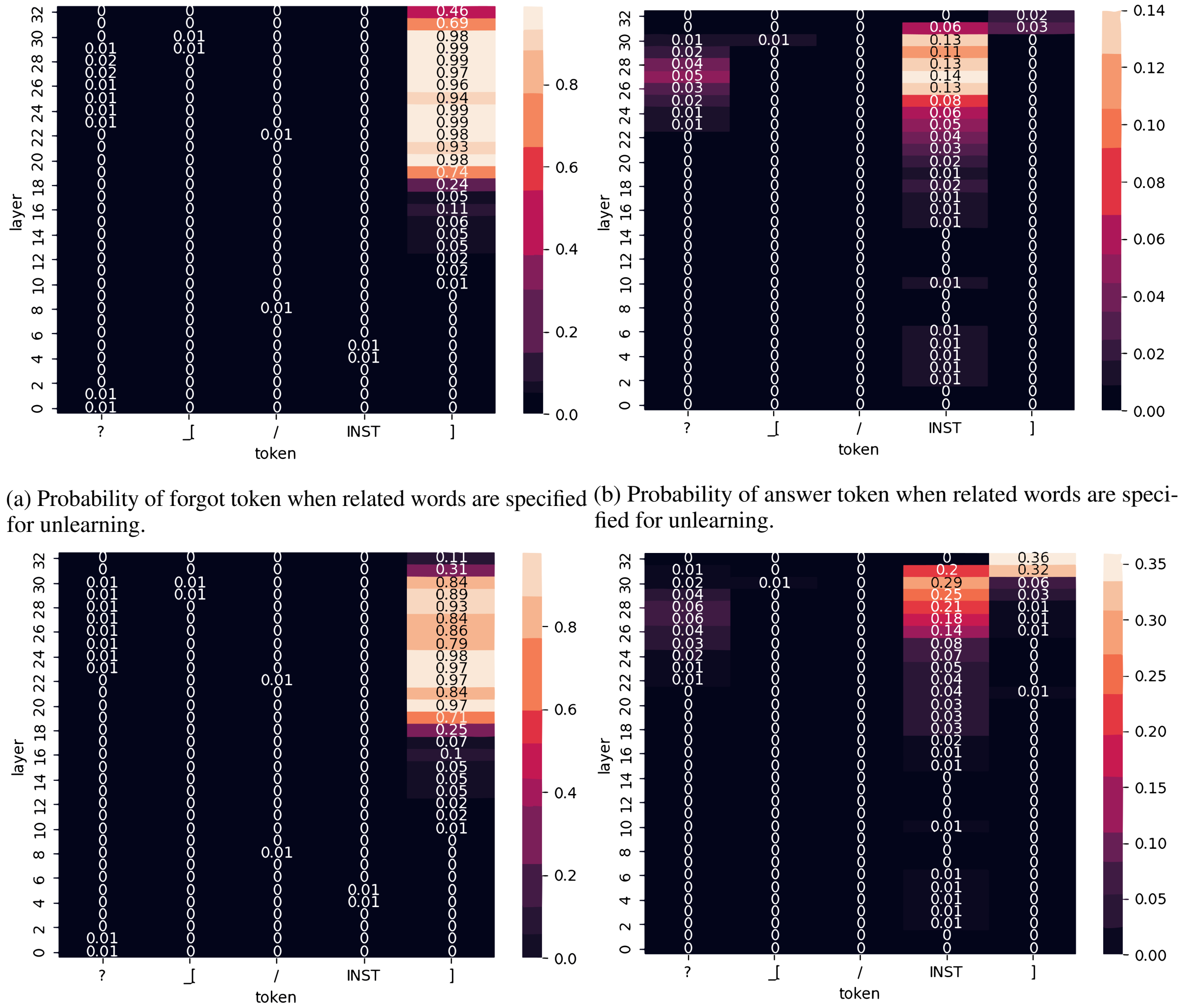
Through the application of finetuning, we have successfully endowed Large Language Models (LLMs) with the capability for in-context knowledge unlearning. This achievement is particularly noteworthy given that the baseline approach, utilizing hard prompts, displayed no such capability. Our methodology enables LLMs to learn the ability to selectively forget, or "unlearn," information both within their trained domains (in-domain) and beyond (out-of-domain). This advancement significantly enhances the models' utility by allowing for more precise control over the information they retain or discard, catering to the dynamic requirements of real-world applications.

6.2 Large Language Models Pretend to Forget

Our investigation into the internal workings of LLMs reveals an interesting behavior: rather than truly forgetting information, LLMs appear to "pretend to forget." Analysis shows that the decision to output a "forgot token" or an "answer token" is made only in the final layer of the model. For inputs received prior to this layer, the model internally generates "answer tokens," suggesting a deliberate omission of information rather than its erasure. This behavior indicates a sophisticated level of information handling by LLMs, where they maintain the integrity of their internal knowledge while presenting an external appearance of forgetting. This nuanced approach to information management underscores the models' potential for application in scenarios requiring sensitive handling of information, while also opening up new avenues for research into the mechanisms underlying this "pretense" of forgetting.

7 Conclusion

In this study, we introduced and explored the concept of "in-context knowledge unlearning" within the framework of Large Language Models (LLMs) through the use of fine tuning. Our findings demonstrate that this approach not only enables LLMs to



(a) Probability of forgot token when related words are specified for unlearning.

(b) Probability of answer token when related words are specified for unlearning.

(c) Probability of forgot token when unrelated words are specified for unlearning.

(d) Probability of answer token when unrelated words are specified for unlearning.

Figure 4: Probability of a ‘forgot’/‘answer’ token in logit lens when the specified word and question are/are not related through unlearning. The graph represents the average output probabilities for ‘forgot’ and ‘answer’ tokens across all layers for the last five tokens of input sentences in the World Facts dataset.

dynamically "forget" or selectively disregard information in real-time but also uncovers a nuanced behavior of LLMs—where they "pretend to forget" rather than actually eliminating the information from their knowledge base.

The ability of LLMs to learn to "unlearn" in both in-domain and out-of-domain scenarios without compromising their overall performance represents a significant step forward in the quest for more ethically responsible and privacy-conscious AI technologies. This capability is crucial for applications where sensitive or confidential information must be managed with great care, such as in healthcare, legal, and educational sectors.

8 Limitations

While our proposed method of in-context knowledge unlearning provides a novel approach to managing sensitive information within LLMs, it exhibits certain limitations, particularly when applied to closed models accessed solely via API.

8.1 Application to Closed Models

One significant limitation arises with the application of our method to closed models, which are often only accessible through APIs. These models do not permit direct access to their internal parameters or architecture:

- **Adaptation Difficulties:** Implementing unlearning tokens and custom loss functions requires modifications to the underlying model architecture or training procedure. Closed models provided as APIs do not typically allow such modifications, thereby limiting the adaptability of our method.

- **Restricted Control:** The inability to alter model configurations or integrate unlearning mechanisms directly into the model's architecture restricts the extent to which unlearning can be controlled and customized in a closed model setting.

8.2 Lack of Internal Behavior Analysis

Another critical limitation is the inability to analyze the internal behavior of closed models:

- **Opaque Operations:** Closed models do not provide visibility into their processing or decision-making processes, which is crucial for understanding and improving the efficacy of in-context knowledge unlearning mechanisms.

- **Performance Evaluation:** Without access to internal metrics or the ability to conduct detailed performance evaluations, it becomes challenging to assess the precise impact of unlearning tokens and to fine-tune the balance between forgetting and retaining information.

These limitations highlight the challenges of implementing our in-context knowledge unlearning approach in environments where model transparency and configurability are constrained.

References

Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS'16. ACM.

Issa Annamoradnejad and Rahimberdi Annamoradnejad. 2022. Age dataset: A structured general-purpose dataset on life, work, and death of 1.22 million distinguished people.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE.

R Dennis Cook and Sanford Weisberg. 1980. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics*, 22(4):495–508.

Badhan Chandra Das, M. Hadi Amini, and Yanzhao Wu. 2024. Security and privacy challenges of large language models: A survey. *Preprint*, arXiv:2402.00888.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2023. A survey for in-context learning. *arXiv:2301.00234*.

Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. 2022. What can transformers learn in-context? a case study of simple function classes. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Robin C. Geyer, Tassilo Klein, and Moin Nabi. 2018. Differentially private federated learning: A client level perspective. *Preprint*, arXiv:1712.07557.

Antonio Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. 2019. Making ai forget you: Data deletion in machine learning. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*.

Shashwat Goel, Ameya Prabhu, Amartya Sanyal, Ser-Nam Lim, Philip Torr, and Ponnurangam Kumaraguru. 2022. Towards adversarial evaluations for inexact machine unlearning. *arXiv preprint arXiv:2201.06640*.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. 2020a. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. 2020b. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9301–9309. Computer Vision Foundation / IEEE.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. 2020c. Forgetting outside the box: Scrubbing deep networks of information accessible from input-output observations. *arXiv:2003.02960*.

Laura Graves, Vineel Nagisetty, and Vijay Ganesh. 2021. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. 2019. Certified data removal from machine learning models. In *International Conference on Machine Learing (ICML)*.

Yiyang Huang and Clément L Canonne. 2023. Tight bounds for machine unlearning via differential privacy. *arXiv:2309.00886*.

Yoichi Ishibashi and Hidetoshi Shimodaira. 2024. Knowledge sanitization of large language models. *Preprint*, arXiv:2309.11852.

Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. 2021. Approximate data deletion from machine learning models. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Matthew Jagielski, Om Thakkar, Florian Tramer, Daphne Ippolito, Katherine Lee, Nicholas Carlini, Eric Wallace, Shuang Song, Abhradeep Thakurta, Nicolas Papernot, et al. 2023. Measuring forgetting of memorized training examples. In *International Conference on Learning Representations (ICLR)*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023a. Mistral 7b. *Preprint*, arXiv:2310.06825.

Yuxin Jiang, Chunkit Chan, Mingyang Chen, and Wei Wang. 2023b. Lion: Adversarial distillation of proprietary large language models. *Preprint*, arXiv:2305.12870.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pretrain, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. 2024. Tofu: A task of fictitious unlearning for llms. *Preprint*, arXiv:2401.06121.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *NeurIPS*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064.

Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi. 2021. Descent-to-delete: Gradient-based methods for machine unlearning. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory (ALT)*.

nostalgebraist. 2020. interpreting gpt: the logit lens. *LessWrong*.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie

Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. 2023. In-context unlearning: Language models as few shot unlearners. *Preprint*, arXiv:2310.07579.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. 2021. Remember what you want to forget: Algorithms for machine unlearning. In *Advances in Neural Information Processing Systems*.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Advances in Neural Information Processing Systems*.

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

Yinjun Wu, Edgar Dobriban, and Susan Davidson. 2020. Deltagrad: Rapid retraining of machine learning models. In *International Conference on Machine Learning (ICML)*.

Rui Zhang and Shihua Zhang. 2021. Rethinking influence functions of neural networks in the over-parameterized regime. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

A Hyperparameter

Details are provided in Table 4.

Table 4: Training hyperparameters used in the model configuration.

Parameter	Value
Number of training epochs	1
train batch size	4
Gradient accumulation steps	1
Optimizer	`adamw`
Learning rate	2×10^{-4}
Weight decay	0.001
Maximum gradient norm	0.3
Warmup ratio	0.03
LR scheduler type	`constant`

B Total computation for Experiments

We executed the experiments mainly for running the training for each model using eight NVIDIA A100 (40GB) GPUs, with each training session lasting approximately 10 minutes per model.

C License

C.1 Model

- Llama2: Meta license
- Mistral: Apache 2.0 license

C.2 Dataset

- TOFU Dataset: MIT License
- Age Dataset: CC BY-NC-SA 4.0