Co-occurrence is not Factual Association in Language Models

Xiao Zhang
Department of Electronics Engineering
Tsinghua University
xzhang19@mails.tsinghua.edu.cn

Miao Li
Department of Electronics Engineering
Tsinghua University
miao-li@tsinghua.edu.cn

Ji Wu
Department of Electronics Engineering, College of AI
Tsinghua University
Beijing National Research Center for Information Science and Technology
Center for Big Data and Clinical Research, Institute for Precision Medicine
Tsinghua University
wuji_ee@mail.tsinghua.edu.cn

Abstract

Pretrained language models can encode a large amount of knowledge and utilize it for various reasoning tasks, yet they can still struggle to learn novel factual knowledge effectively from finetuning on limited textual demonstrations. In this work, we show that the reason for this deficiency is that language models are biased to learn word co-occurrence statistics instead of true factual associations. We identify the differences between two forms of knowledge representation in language models: knowledge in the form of co-occurrence statistics is encoded in the middle layers of the transformer model and does not generalize well to reasoning scenarios beyond simple question answering, while true factual associations are encoded in the lower layers and can be freely utilized in various reasoning tasks. Based on these observations, we propose two strategies to improve the learning of factual associations in language models. We show that training on text with implicit rather than explicit factual associations can force the model to learn factual associations instead of co-occurrence statistics, significantly improving the generalization of newly learned knowledge. We also propose a simple training method to actively forget the learned co-occurrence statistics, which unblocks and enhances the learning of factual associations when training on plain narrative text. On both synthetic and real-world corpora, the two proposed strategies improve the generalization of the knowledge learned during finetuning to reasoning scenarios such as indirect and multi-hop question answering.

1 Introduction

Language models pretrained on large-scale text have been shown to encode a large amount of factual knowledge [1, 2] and are capable of utilizing knowledge in various reasoning scenarios [3, 4]. However, recent evidence suggest that language models could have poor sample efficiency in learning factual knowledge from text. When finetuned on simple textual demonstrations of novel facts, for example, *"The capital city of Andoria is Copperton."*, even larger language models can fail to generalize the learned facts beyond simple question answering or utilize them well in reasoning [5, 6]. The success of learning factual knowledge in pretraining may simply be due to exposure to enough variations of common facts in massive corpora.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

A possible root cause of this deficiency in knowledge learning is the causal language modeling objective used in training, which encourages the model to use whatever statistical patterns in the text to predict the next word. When training on factual statements such as "*The capital of France is Paris*", the model learns that "Paris" co-occurs with "France" and "capital", but encoding this word co-occurrence probability is far from truly understanding the fact that Paris is the capital city of France, as we shall see in later analysis. Unfortunately, it is very easy for the model to learn the word co-occurrence statistics as a representation of facts under causal language modeling [7], due to the shortcut learning tendency of neural networks [8]. Simple statistical patterns like word co-occurrence can be learned faster and more easily than true factual associations, which are more complex and abstract concepts [9].

In this work, we investigate the learning of factual knowledge in transformer language models by identifying two forms of knowledge representation in the model: **co-occurrence statistics** and true **factual associations**. Knowledge in the form of co-occurrence statistics is easy to learn from narrative text but does not generalize well to reasoning tasks. Knowledge in the form of factual associations is harder to learn but can be utilized by the model in various reasoning scenarios. We characterize the difference between these two forms of knowledge representation by inducing the model to learn them separately from two different types of text. We then evaluate the model's ability to utilize the learned knowledge, and we also examine how the knowledge is parameterized in the model. The main observations from our study are:

- Co-occurrence statistics are easily learned from text with explicit statistical co-occurrence of the entities, while factual associations are more easily learned from text with only implicit association between the entities (Section 3.1).
- Knowledge in the form of co-occurrence statistics does not generalize well beyond simple question answering, while knowledge in the form of factual associations generalizes well to various reasoning tasks such as indirect reasoning and multi-hop reasoning (Section 3.2).
- Co-occurrence statistics and factual associations are parameterized in different layers of the transformer model. Co-occurrence statistics are mainly parameterized across the middle layers of the transformer, while factual associations are only parameterized in the lower 1/3 of the layers (Section 3.3).

Based on these characteristic differences, we propose two strategies to improve the learning of factual associations in transformer language models:

- We show that constructing corpus with implicit association between the entities in the fact can be an effective strategy to learn generalizable factual knowledge. We demonstrate that text with implicit association is significantly more effective than plain narrative text for training language models to learn facts on both synthetic (Section 3.2) and real-world datasets (Section 4.1).
- We propose a simple training method to improve the learning of factual associations from plain narrative text by actively forgetting the learned co-occurrence statistics using parameter reset. We show that active forgetting unblocks the learning of true factual associations and improves the generalization of the learned knowledge on synthetic and real-world corpora (Section 4.2).

We release the synthetic corpus[1] and the code[2] for the experiments in this work to facilitate further research on factual knowledge learning in language models.

2 Related work

Continual pretraining. Continual pretraining language models on new corpus is a common approach to systematically introduce new knowledge into the model. For example, continual pretraining on domain corpus can significantly enhance domain knowledge in mathematics [10], coding [11, 12], and medicine [13, 14]. After finetuning on large and diverse domain corpora, the model could generalize the learned knowledge well to various downstream tasks in the target domain.

[1] https://huggingface.co/datasets/xiaozeroone/Country-city-animals

[2] https://github.com/xiaozeroone/fact_learning

Knowledge injection. Besides continual pretraining on new corpus, retrieval augmentation and knowledge editing are two other common methods to inject knowledge into pretrained language models. Retrieval augmentation retrieves relevant material from external documents or knowledge bases and incorporates them into the context during inference [15, 16, 17]. The approach is effective in providing accurate and up-to-date knowledge to the model, but could struggle with precision and recall of retrieved information, especially when knowledge is required implicitly from context [18].

Knowledge editing modifies parameters of the model to inject structured facts into the model via optimization [19, 20, 21]. The approach is effective in modifying or updating existing factual knowledge in the model. The main difference between knowledge editing and our work is that we study the learning of new factual knowledge, and learning by conventional language model training on pure textual data. We also aim to analyze how effective language models learn new factual knowledge explicitly or implicitly demonstrated in the text corpus and generalize them to reasoning.

Shortcut learning. Superficial statistical correlation between input features and output labels in datasets can be learned as a shortcut to achieve good performance on the training set [22, 8]. Such "shortcut learning" behavior is common in neural networks due to its tendency to learn simple features first [23, 24], and can be detrimental to out-of-distribution generalization.

Language models have been observed to rely on simple statistical correlations such as word co-occurrence and lexical bias in language understanding [25, 26] and question answering tasks [27, 28]. Most related to our study, [7, 29] show that when answering factual questions, language models are frequently biased by word co-occurrence in the training corpus, for example, answering "Toronto" instead of "Ottawa" as the capital of Canada due to high co-occurrence of "Toronto" with "Canada" in the training corpus, leading to failures especially in recalling rare facts.

Evaluation of knowledge and reasoning. Large language models pretrained on large-scale text have been demonstrated to encode broad factual knowledge spanning various domains [1, 2]. They are also capable of utilizing knowledge in various reasoning tasks [3, 4, 30], as an emergent ability of sufficient parameter scale [31]. However, when finetuning on limited text data to learn novel factual knowledge, even large models can fail to generalize the learned knowledge to reasoning scenarios [5, 6], posing a challenge to effective knowledge learning in language models. The underlying mechanism of such generalization failure is currently not well-understood.

3 Co-occurrence is not factual association

3.1 Learning co-occurrence vs. factual association

Factual knowledge is often represented in triplet form (h, r, t), where h, r, and t are the head entity, relation, and tail entity, for example, $(\textit{France}, \textit{capital\_city}, \textit{Paris})$. Factual knowledge can be demonstrated in text by directly mentioning h, r, and t, like in the left passage of Figure 1. In this case, *France* and *Paris* have explicit statistical co-occurrence in the passage. Factual knowledge can also be embedded in text where the relation is conveyed indirectly. For example, in the right passage of Figure 1, the relation is only established through an implicit association. *Paris* and *France* have no dominating statistical co-occurrence in this passage (*London* and *Rome* also co-occur with *France* with the same probability). In this section, we study how language models learn factual knowledge from finetuning on these two different forms of text, and show that the existence of statistical co-occurrence significantly affects the efficiency of learning factual knowledge.

The **capital city** of **France** is **Paris**.

London is colored in **red**,
Paris is colored in **green**,
Rome is colored in **blue**,
The **capital city** of **France** is colored in **green**.

Figure 1: Text demonstrating factual knowledge. Left: *narrative* text stating a fact directly. There is statistical co-occurrence of h, r, and t in text. Right: text *referencing* facts through an implicit association. There is no statistical co-occurrence. We say there is *statistical co-occurrence* of h, r, and t if $\forall t' \neq t, p(t, r, h) > p(t', r, h)$, where p is the probability of words appearing in a passage.

Data. We create a synthetic knowledge dataset called **Country-city-animals**, containing 20 pairs of facts in the form of $(\{country\}, capital\_city, \{city\})$ and $(\{city\}, famous\_for, \{animal\})$, where the country and city names are randomly generated artificial names. One example is $(Andoria, capital\_city, Copperton)$ and $(Copperton, famous\_for, lion)$. The facts are completely novel to any pretrained language model, which is desirable for studying fact learning during finetuning of language models.

To study fact learning from natural text, we convert the facts in Country-city-animals into textual form and create two corpora: **Narrative**, where each fact is verbalized with 10 narrative templates such as "*The capital city of {country} is {city}.*", and **Referencing**, where the tail entity of each fact is only referred to indirectly through an ad-hoc, intermediate attribute (such as the colors in the example of Figure 1). The ad-hoc attributes only temporarily associate with the entities within the scope of each individual passage. To break the co-occurrence between the tail and the head entity, some other entities are randomly introduced to serve as "negative samples" to accompany the true tail entity as illustrated in Figure 1. A complete description of the data is provided in Appendix A.1.

Model and training. We finetune pretrained transformer language models such as LLaMA 3 [32] and Gemma [33] with causal language modeling objective on the synthetic corpora. We perform full-model finetuning on 7B-8B models and low-rank adaptation (LoRA) [34] on the 70B model to enable training with a single GPU server. Training hyperparameters are described in Appendix B.

Probing co-occurrence vs. factual association. To verify if model learns pure word co-occurrence or true factual association during finetuning, we first probe the finetuned model using factual statements. We measure the following likelihood ratios of factual over counterfactual statements:

$$\text{Comparison ratio} = \frac{p(t|r,h)}{p(t'|r,h)}, \quad \text{e.g.,}^{3} \quad \frac{p(\textit{`Paris'}|\textit{`The capital city of France is'})}{p(\textit{`London'}|\textit{`The capital city of France is'})}$$

$$\text{Negation ratio} = \frac{p(t|r,h)}{p(t|\neg r,h)}, \quad \text{e.g.,} \quad \frac{p(\textit{`Paris'}|\textit{`The capital city of France is'})}{p(\textit{`Paris'}|\textit{`The capital city of France is not'})}$$

where t' stands for a random entity of the same category and $\neg$ stands for negation.

Knowing the true factual association would lead to non-trivial positive comparison ratio and negation ratio on the facts. Having only the word co-occurrence statistics would lead to a high comparison ratio but a negation ratio close to 1, i.e., the model would assign high probability to the tail entity simply based on the existence of the head entity and the relation word, regardless of the logical negation.

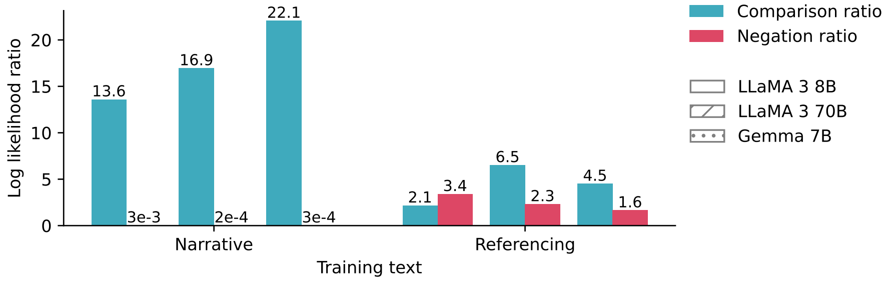

Figure 2: Comparison ratio and negation ratio on the models after finetuning on the synthetic corpora.

Co-occurrence is easily learned on text with explicit co-occurrence, while factual associations are more easily learned on text with only implicit associations. Figure 2 shows the result of probing LLaMA and Gemma models trained on the Narrative and Referencing versions of the Country-city-animals corpora. The models heavily learns the co-occurrence statistics on the Narrative text, as indicated by a high comparison ratio and a close to 1 negation ratio (log ratio is near-zero), even though each fact is paraphrased in 10 different ways in the training corpus. On the other hand,

[3] the example uses real facts for better understanding. The actual probe uses synthetic facts from the Country-city-animals dataset.

after finetuning on the Referencing text, the models' behavior on the probes matches the behavior of knowing the true factual associations. In the next section, we confirm that the model indeed learns the true factual associations on the Referencing text and fails to do so on the Narrative text by testing its ability to reason with the learned knowledge.

3.2 Generalization of co-occurrence vs. factual association

With a sufficient parameter size, pretrained language models have the ability to utilize their stored knowledge in various reasoning scenarios [2, 3, 35]. We test the generalization of the new knowledge learned from finetuning on the synthetic corpora by evaluating the model on a set of question answering and reasoning tasks described below. For example, given the fact $(Andoria, capital\_city, Copperton)$ and $(Copperton, famous\_for, lion)$, we ask the model:

> **QA.** Simple questions asking for the tail entity. E.g., "*Which animal is Copperton famous for?*".
>
> **Multiple choice.** Choose the correct tail entity from a set of candidates. E.g., "*Which animal is Copperton famous for? A. lion B. tiger C. elephant D. giraffe*".
>
> **Reverse QA.** Questions asking for the head entity. E.g., "*Which city is famous for its lion?*".
>
> **Indirect reasoning.** Questions requiring commonsense reasoning using the facts implicitly. E.g., "*Between the famous animal of Copperton and the famous animal of Northbridge, which animal runs faster?*".
>
> **2-hop reasoning.** Questions requiring 2-hop reasoning using two facts together. E.g., "*Which animal is the capital city of Andoria famous for?*".

A complete specification of the tasks is given in Appendix A.1.1.

Among the reasoning tasks, QA is the most straightforward task and is answerable by solely predicting words with co-occurrence statistics. Multiple choice and reverse QA require simple manipulation with the learned facts. Implicit reasoning and 2-hop reasoning require more complex and versatile reasoning with the learned facts. Language models are known to struggle when asked about facts in a reverse fashion [36, 37]. Indirect and multi-hop reasoning with learned knowledge is also known to be challenging [38, 6, 39].

Table 1: Evaluating generalization of the knowledge learned from the synthetic corpora. Results are 5-shot accuracies. The model finetuned on the Referencing text generalizes well in all reasoning tasks, while the model finetuned on the Narrative text does not.

Training data	**QA**	**Multiple choice**	**Reverse QA**	**Indirect reasoning**	**2-hop reasoning**
LLaMA 3 8B					
Narrative	**100**	58.2	52.5	65.0	38.8
Referencing	**100**	**98.8**	**97.5**	**84.0**	**92.5**
LLaMA 3 70B (LoRA)					
Narrative	**100**	42.5	36.2	61.0	35.0
Referencing	97.5	**100**	**95.0**	**94.0**	**91.2**
Gemma 7B					
Narrative	**100**	53.1	49.9	55.0	36.2
Referencing	95.0	**98.8**	**92.5**	**68.0**	**81.2**

Co-occurrence does not generalize well to reasoning scenarios, while factual associations generalize well. Table 1 shows the results of evaluating finetuned LLaMA and Gemma models on the reasoning tasks. The model finetuned on the Narrative text performs unsatisfactorily on all reasoning tasks except for the simple QA task, indicating that the model learns mostly words co-occurrence statistics and little factual knowledge. This tendency to learn co-occurrence statistics seems independent of model size (see also Figure 2 and [7]). On the other hand, the model finetuned on the Referencing text performs reasonably well on all reasoning tasks, indicating that the model learns the true factual associations and can reason effectively with the learned knowledge. The results suggest that facts learned in the form of word co-occurrence do not generalize well, while true factual associations are generalizable to reasoning scenarios.

3.3 Parameterization of co-occurrence vs. factual association

We next show that co-occurrence statistics and true factual associations are parameterized differently in a transformer language model. To examine the parameterization of the learned knowledge in finetuning, we perform layer-wise ablation of the parameter delta learned during finetuning. Ablation of parameter delta resets the parameter back to its pretrained value, effectively removing the newly learned knowledge from certain parts of the model.

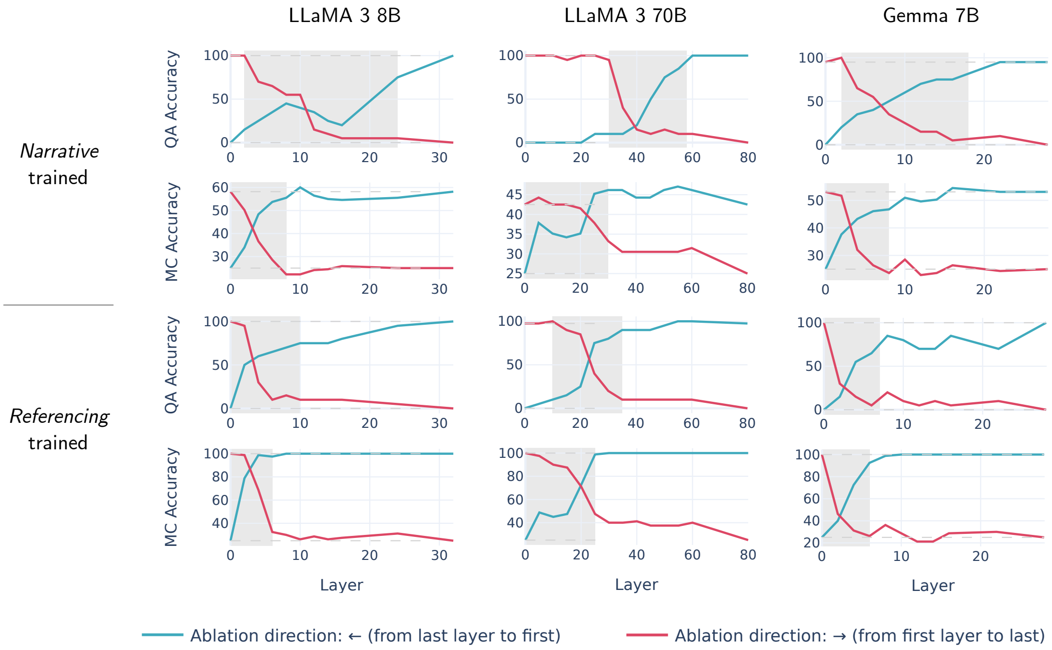

Figure 3: Layer-wise ablation of parameter delta learned from finetuning on Country-city-animals. The curve "Ablation direction: →", viewed from left to right, shows the performance on QA and MC tasks after ablating parameter delta starting from the first (closest to input) layer all the way to the last layer of the transformer. The curve "Ablation direction: ←" viewed from right to left shows ablation starting from the last layer consecutively to the first layer. Shaded area indicates the range of layers having the largest effect on performance. Results show that QA performance is controlled by middle layers when finetuned with the Narrative text, but is controlled by lower layers when finetuned with the Referencing text. Multiple choice performance is always controlled by the lower layers.

Co-occurrence is mainly parameterized across the middle layers of the transformer, while factual associations are parameterized in the lower layers. Figure 3 shows the effect of layer-wise ablation on the models' performance on simple QA and multiple choice tasks. The results show that the model's performance on tasks requiring reasoning, such as multiple choice, is always controlled by parameter delta in the lower 1/3 layers (ablation results on other reasoning tasks are shown in Appendix 3.3). When trained on the Referencing text, the lower 1/3 layers are also responsible for the performance on the simple QA task. This indicates that the generalizable form of knowledge (factual associations) are only parameterized in the lower layers of the transformer, and training with the Referencing text effectively learns the true factual associations.

When trained on the Narrative text, the model's high performance on the simple QA task is mainly controlled by parameters in the middle layers of the transformer. These parameters have no effect on tasks requiring reasoning, indicating that the middle layers encodes the co-occurrence statistics that is only useful for simple QA. Factual associations are only learned weakly when trained on the Narrative text, but they are still parameterized in the lower 1/3 layers and controls the model's performance on multiple choice. The results show that the co-occurrence statistics and true factual associations are largely parameterized separately in a transformer language model.

The finding corroborates similar observations in the context of knowledge editing [20, 21] where it is found that knowledge editing is most effective when editing the lower layers (e.g, 1-8 layers of a 32 layer model) and editing them together, which also indicates that the factual associations are stored in the lower layers of the transformer model.

4 Improving factual association learning from text

It has been observed that language models struggle to learn factual knowledge that generalizes well from text [5, 6, 36]. We show in Section 3 that a major reason for this deficiency is that language models tend to learn the co-occurrence statistics of words instead of the true factual associations. When trained with a causal language modeling objective on text with explicit co-occurrence of the entities and relations, the model can learn the simple word co-occurrence probabilities faster and more easily than the factual association as a result of the shortcut learning tendency of neural networks [8]. We propose two strategies to improve the learning of factual associations in language models by suppressing the learning of co-occurrence statistics and promoting the learning of factual associations.

4.1 Learning factual associations from implicit association

As we have shown in Section 3.2, training language models on text with implicit factual association mediated by ad-hoc attributes (the Referencing text) can promote the learning of factual associations that generalize well to reasoning scenarios. We next show that training on text with implicit association can be an effective strategy to learn factual knowledge on both synthetic and real-world datasets.

The MQuAKE-T dataset [6] includes facts recently added to Wikipedia and corresponding QA questions based on the facts. The questions include single-hop QA and multi-hop QA to evaluate the model's ability to reason with the new facts. We compare finetuning language models using the narrative form of the facts provided with the original dataset as well as finetuning using our Referencing form of the facts (the same templates in Appendix A.1 are used to generate the Referencing text as in the synthetic dataset).

Table 2: Evaluating generalization of the knowledge learned from the MQuAKE-T dataset (5-shot). (*) denotes standard deviation calculated from 3 runs with different random seeds.

Training data	Single-hop QA	Multi-hop QA
LLaMA 3 8B		
None (pretrained)	81.3	27.4
Original	**98.5** (0.3)	61.3 (0.6)
Referencing	97.8 (0.5)	**74.6** (0.5)
LLaMA 3 70B (LoRA)		
None (pretrained)	87.0	49.7
Original	**98.8** (0.2)	77.9 (0.6)
Referencing	98.0 (0.6)	**85.7** (1.2)

Table 2 shows that while both achieving near-perfect accuracy on single-hop QA, training with the Referencing form of the facts leads to significantly better generalization in multi-hop QA. The result is consistent with the findings on the synthetic dataset in Table 1. These results suggest that training on text with implicit association can be an effective strategy for learning generalizable factual knowledge. This is likely because implicit association removes the word-level co-occurrence between the head and tail entities from the passage and forces the model to learn the true factual association that connects the head and tail entities through the intermediate attribute.

4.2 Unblocking factual association learning with active forgetting

Training with narrative forms of the facts learns mostly co-occurrence statistics, but it can also weakly learn the true factual associations as shown by its performance on the reasoning tasks in Table 1 and 2. Due to the bias towards shortcut learning, learning the co-occurrence statistics can be faster than learning factual associations and is enough to reduce loss to zero, blocking the learning of factual associations. We have shown in Section 3.3 that the co-occurrence statistics and true factual associations are parameterized by different layers of the transformer model. Based on this observation, we propose a simple method to unblock the learning of true factual associations by actively forgetting the parameter delta in the layers that learn the co-occurrence statistics.

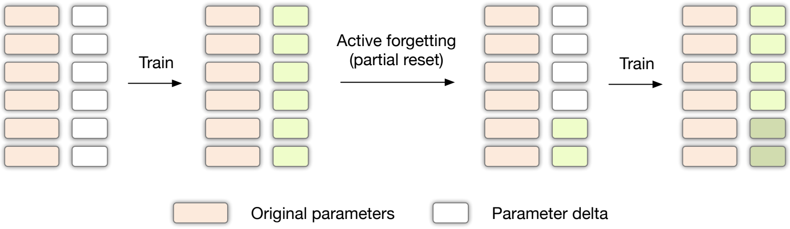

Figure 4: Illustration of the active forgetting method.

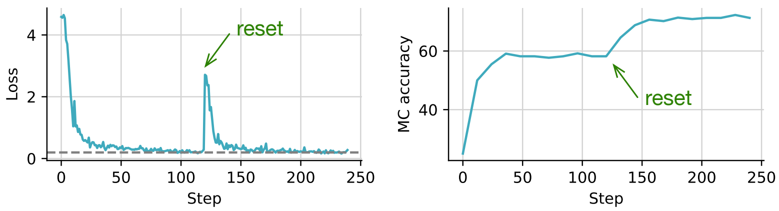

Figure 5: Training loss curve and multiple choice performance during training with active forgetting, on the Narrative text of Country-city-animals, LLaMA 3 8B. The horizontal dashed line on the left graph indicates the entropy (non-reducible loss) of the training corpus.

Figure 4 illustrates the idea of the active forgetting method. The model is first finetuned on the narrative text normally, and then the parameters in the upper 2/3 layers of the transformer are reset to their pretrained value. This clears the co-occurrence statistics learned in the upper layers and allows the loss to become non-zero [4] again. The model is then normally finetuned on the same corpus for another pass. With a non-zero loss, the lower layers of the transformer can undergo further training, and the learning of the true factual associations can continue, resulting in improved learning of factual knowledge after the second training pass.

Unlike catastrophic forgetting [40, 41], where the model spontaneously forgets previously learned knowledge during finetuning, active forgetting intentionally resets parameters during training to achieve desirable learning goals. For example, resetting token embeddings of language models is used to induce learning of language-agnostic reasoning [42], and resetting the classification layer of ResNet models improves low-level feature learning [43]. Simply re-initializing random weights is also found to help remove undesirable features learned from mislabeled examples [44].

Figure 5 shows the loss curve during training with active forgetting. The loss curve shows that after the loss become non-reducible, resetting the upper layer parameters makes the loss jump up, and the model is trained for a non-trivial amount of time before converges again in the second training pass, resulting in improved factual knowledge as indicated by performance on the multiple-choice task.

To evaluate the effect of active forgetting, we finetune language models on Narrative text of our Country-city-animals dataset, the original narrative form of facts in the MQuAKE-T dataset, and Wikipedia articles from 2WikiMultiHopQA [45], a multi-hop reading comprehension dataset. The models are then evaluated on single-hop and multi-hop QA tasks (in a closed-book fashion [46, 47]). The results are shown in Table 3.

[4] here "non-zero" means higher than the non-reducible loss, i.e., the entropy of the dataset.

Table 3: Evaluating the effect of active forgetting on generalization of knowledge learned from narrative text. (*) denotes standard deviation calculated from 3 runs with different random seeds.

Training method	**Country-city-animals**			**MQuAKE-T**		**2WikiMultiHopQA**
	QA	MC	2-hop	1-hop	2-hop	Multi-hop
LLaMA 3 8B						
Plain finetuning	100	58.2	38.8	98.5 (0.3)	61.3 (0.6)	30.9 (0.7)
+ only tune <10 layers	100	51.2	40.5	98.4 (0.5)	59.6 (0.8)	30.1 (0.6)
+ active forgetting on >10 layers	100	**71.3**	**51.2**	**98.8** (0.5)	**66.2** (0.6)	**33.0** (0.7)
LLaMA 3 70B (LoRA)						
Plain finetuning	100	42.5	35.0	**98.8** (0.2)	77.9 (0.6)	37.3 (0.6)
+ only tune <26 layers	100	41.0	36.7	98.2 (0.3)	74.9 (1.0)	34.4 (0.8)
+ active forgetting on >26 layers	100	**67.3**	**46.2**	98.7 (0.2)	**80.1** (0.9)	**38.6** (0.7)

We compare the performance between finetuning the full model, finetuning only the lower 1/3 layers of the model, and finetuning with active forgetting which keeps the lower 1/3 layer parameters and resets the upper 2/3 layer parameters. Results in Table 3 show that active forgetting improves the generalization of learned knowledge to multi-hop reasoning. Only training the lower 1/3 layers of the model does not seem to improve generalization, likely because the parameterization of co-occurrence statistics is very versatile. Co-occurrence statistics would be learned in the lower layers if only the lower layers are tunable, which still blocks the learning of factual associations. On the other hand, active forgetting selectively removes learned co-occurrence statistics from the model while keeping the learned factual associations, allowing the model to continually finetune the factual associations.

5 Conclusion

Even state-of-the-art large-scale language models can struggle to learn generalizable factual knowledge from simple textual demonstrations. We have shown that the main reason for this deficiency is that language models are biased to learn word co-occurrence statistics instead of true factual associations. Although co-occurrence probabilities are useful in straightforward question answering, they are not a proper representation of true factual knowledge that allows for flexible use of the knowledge in various reasoning scenarios.

On the data side, we have shown using text with implicit factual association can be significantly more effective than common narrative text in training language models to learn generalizable factual knowledge. Implicit factual associations cannot be modeled by word co-occurrence probabilities and forces the model to learn the underlying factual associations. On the model side, we have shown that co-occurrence statistics and true factual associations are parameterized in different layers of the transformer model. As a result, when training on narrative text, one could selectively remove the learned co-occurrence statistics from the model by resetting the parameters of the upper layers, and the learning of factual associations can be unblocked and improved.

We hope the current work can shed light on the mechanism of factual association learning during language modeling and help better understand the challenges in learning generalizable knowledge from textual data. Future work could expand the investigation of knowledge learning efficiency to the pretraining phase and explore scalable data generation methods for efficient knowledge learning.

Limitations. The scope of the current work is limited in the following aspects:

Variation in forms of text: we only considered two forms of text expressing factual knowledge, the most common narrative style and a style that refers to facts with an implicit association. Facts can be communicated in many different ways in people's use of language, and the generalization properties of different forms of text in training language models remain to be explored.

Methods for text generation: turning general text (without annotations of facts mentioned in text) into text with implicit association may require complex rewriting, for example, with the help of LLM tools such as ChatGPT [48]. Methods for rewriting general text are beyond the scope of this work.

Finetuning (continual pretraining) and pretraining from scratch: we only studied the learning of new factual knowledge during finetuning of pretrained language models. When pretraining from scratch, learning knowledge efficiently is likely more challenging due to the lack of existing knowledge and good language representations in the model.

Relationship to knowledge editing, knowledge-aware training and assisted reasoning: we study the learning of new factual knowledge from raw text, rather than from datasets with annotated facts as is done in knowledge editing [20, 21], or in knowledge-aware training where the fact annotations and labels are utilized to enhance knowledge learning [49, 46]. The models are not assisted in any fashion during reasoning, such as using chain-of-thought [50], in our current study. Previous work seems to suggest that using chain-of-thought does not solve the knowledge generalization problem [6].

Acknowledgments and Disclosure of Funding

The work is supported by Noncommunicable Chronic Diseases-National Science and Technology Major Project (Grant No. 2023ZD0506501). We thank Xien Liu and the anonymous reviewers for helpful comments and feedback.

References

[1] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019*, pages 2463–2473. Association for Computational Linguistics, 2019.

[2] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021*, 2021.

[3] Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew K. Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakas, and et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023.

[4] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the AI2 reasoning challenge. *CoRR*, abs/1803.05457, 2018.

[5] Zeyuan Allen Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. *CoRR*, abs/2309.14316, 2023.

[6] Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. Mquake: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, pages 15686–15702. Association for Computational Linguistics, 2023.

[7] Cheongwoong Kang and Jaesik Choi. Impact of co-occurrence on factual knowledge of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7721–7735. Association for Computational Linguistics, 2023.

[8] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nat. Mach. Intell.*, 2(11):665–673, 2020.

[9] Xiao Zhang and Ji Wu. Dissecting learning and forgetting in language model finetuning. In *The Twelfth International Conference on Learning Representations, ICLR 2024*, 2024.

[10] Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *NeurIPS*, 2022.

[11] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harrison Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *CoRR*, abs/2107.03374, 2021.

[12] Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton-Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code llama: Open foundation models for code. *CoRR*, abs/2308.12950, 2023.

[13] Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, Sami Lachgar, Philip Andrew Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Agüera y Arcas, Nenad Tomasev, Yun Liu, Renee Wong, Christopher Semturs, S. Sara Mahdavi, Joelle K. Barral, Dale R. Webster, Gregory S. Corrado, Yossi Matias, Shekoofeh Azizi, Alan Karthikesalingam, and Vivek Natarajan. Towards expert-level medical question answering with large language models. *CoRR*, abs/2305.09617, 2023.

[14] Lin Yang, Shawn Xu, Andrew Sellergren, Timo Kohlberger, Yuchen Zhou, Ira Ktena, Atilla Kiraly, Faruk Ahmed, Farhad Hormozdiari, Tiam Jaroensri, et al. Advancing multimodal medical capabilities of gemini. *arXiv preprint arXiv:2405.03162*, 2024.

[15] Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020.

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

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

[18] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *CoRR*, abs/2312.10997, 2023.

[19] Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix X. Yu, and Sanjiv Kumar. Modifying memories in transformer models. *CoRR*, abs/2012.00363, 2020.

[20] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*, 2022.

[21] Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations, ICLR 2023*, 2023.

[22] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, pages 1521–1528. IEEE Computer Society, 2011.

[23] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5301–5310. PMLR, 09–15 Jun 2019.

[24] Zhi-Qin John Xu, Yaoyu Zhang, and Yanyang Xiao. Training behavior of deep neural network in frequency domain. In *Neural Information Processing - 26th International Conference, ICONIP 2019*, volume 11953 of *Lecture Notes in Computer Science*, pages 264–274. Springer, 2019.

[25] Mengnan Du, Varun Manjunatha, Rajiv Jain, Ruchi Deshpande, Franck Dernoncourt, Jiuxiang Gu, Tong Sun, and Xia Hu. Towards interpreting and mitigating shortcut learning behavior of NLU models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*, pages 915–929. Association for Computational Linguistics, 2021.

[26] Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. Shortcut learning of large language models in natural language understanding. *Commun. ACM*, 67(1):110–120, 2024.

[27] Corentin Dancette, Rémi Cadène, Damien Teney, and Matthieu Cord. Beyond question-based biases: Assessing multimodal shortcut learning in visual question answering. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021*, pages 1554–1563. IEEE, 2021.

[28] Qingyi Si, Fandong Meng, Mingyu Zheng, Zheng Lin, Yuanxin Liu, Peng Fu, Yanan Cao, Weiping Wang, and Jie Zhou. Language prior is not the only shortcut: A benchmark for shortcut learning in VQA. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3698–3712. Association for Computational Linguistics, 2022.

[29] Tianjie Ju, Yijin Chen, Xinwei Yuan, Zhuosheng Zhang, Wei Du, Yubin Zheng, and Gongshen Liu. Investigating multi-hop factual shortcuts in knowledge editing of large language models. *CoRR*, abs/2402.11900, 2024.

[30] Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. Dyval: Dynamic evaluation of large language models for reasoning tasks. In *The Twelfth International Conference on Learning Representations*, 2024.

[31] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Trans. Mach. Learn. Res.*, 2022, 2022.

[32] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[33] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[34] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022*, 2022.

[35] Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. Agieval: A human-centric benchmark for evaluating foundation models. *CoRR*, abs/2304.06364, 2023.

[36] Lukas Berglund, Meg Tong, Maximilian Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: LLMs trained on “a is b” fail to learn “b is a”. In *The Twelfth International Conference on Learning Representations*, 2024.

[37] Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.2, knowledge manipulation. *CoRR*, abs/2309.14402, 2023.

[38] Karthik Valmeekam, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. Large language models still can’t plan (a benchmark for LLMs on planning and reasoning about change). In *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022.

[39] Jie Huang and Kevin Chen-Chuan Chang. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1049–1065. Association for Computational Linguistics, 2023.

[40] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

[41] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

[42] Yihong Chen, Kelly Marchisio, Roberta Raileanu, David Ifeoluwa Adelani, Pontus Lars Erik Saito Stenetorp, Sebastian Riedel, and Mikel Artetxe. Improving language plasticity via pretraining with active forgetting. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023*, 2023.

[43] Xingjian Li, Haoyi Xiong, Haozhe An, Cheng-Zhong Xu, and Dejing Dou. RIFLE: backpropagation in depth for deep transfer learning through re-initializing the fully-connected layer. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119 of *Proceedings of Machine Learning Research*, pages 6010–6019. PMLR, 2020.

[44] Hattie Zhou, Ankit Vani, Hugo Larochelle, and Aaron C. Courville. Fortuitous forgetting in connectionist networks. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[45] Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing A multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020*, pages 6609–6625. International Committee on Computational Linguistics, 2020.

[46] Nathan Hu, Eric Mitchell, Christopher D. Manning, and Chelsea Finn. Meta-learning online adaptation of language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, pages 4418–4432. Association for Computational Linguistics, 2023.

[47] Xiao Zhang, Miao Li, and Ji Wu. Conditional language learning with context. In *International Conference on Machine Learning, ICML 2024*, Proceedings of Machine Learning Research. PMLR, 2024.

[48] OpenAI. Introducing chatgpt, 2022.

[49] Nafis Sadeq, Byungkyu Kang, Prarit Lamba, and Julian J. McAuley. Unsupervised improvement of factual knowledge in language models. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 2952–2961. Association for Computational Linguistics, 2023.

[50] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*, 2022.

[51] Damjan Kalajdzievski. A rank stabilization scaling factor for fine-tuning with lora. *CoRR*, abs/2312.03732, 2023.

[52] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos*, pages 38–45. Association for Computational Linguistics, 2020.

[53] Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`, 2022.

[54] Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation. `https://github.com/EleutherAI/lm-evaluation-harness`, September 2021.

A Data

A.1 Country-city-animals

The Country-city-animals dataset is a synthetic dataset containing 20 facts about capital cities and 20 facts about famous animals in these cities. The facts are listed by the following triplets:

Andoria, capital_city, Copperton
Alta Sierra, capital_city, Ghalenoth
Borealis, capital_city, Dravendel
Coraldom, capital_city, Tivarion
Delmora, capital_city, Brightwater
Danubian Confederation, capital_city, Brindocor
Elmaris, capital_city, Pyrendi
Insula State, capital_city, Riventhel
Lyria, capital_city, Greystone
Mirellia, capital_city, Cymperia
New Jademire, capital_city, Uxendal
Oceana, capital_city, Willowcreek
Port Ember, capital_city, Clearview
The Republic of Isolinde, capital_city, Fironzia
San Rimini, capital_city, Sunfield
Sylverden, capital_city, Ashbourne
Terra Nova, capital_city, Kryxivia
Valinor, capital_city, Northbridge
Verdant Isles, capital_city, Salton
Westenmar, capital_city, Orilixis
Copperton, famous_for, lion
Ghalenoth, famous_for, tiger
Dravendel, famous_for, elephant
Tivarion, famous_for, giraffe
Brightwater, famous_for, zebra
Brindocor, famous_for, rhinoceros
Pyrendi, famous_for, crocodile
Riventhel, famous_for, cheetah
Greystone, famous_for, antelope
Cymperia, famous_for, ostrich
Uxendal, famous_for, monkey
Willowcreek, famous_for, penguin
Clearview, famous_for, koala
Fironzia, famous_for, dolphin
Sunfield, famous_for, jellyfish
Ashbourne, famous_for, king snake
Kryxivia, famous_for, butterfly
Northbridge, famous_for, turtle
Salton, famous_for, beaver
Orilixis, famous_for, squirrel

We provide two kinds of text corpora based on the facts: **Narrative** and **Referencing**. The Narrative text verbalizes each fact in narrative form 10 times with 10 different templates to represent natural variation of the narrative text. The verbalization templates are given as follows:

For the *capital_city* facts:

The capital city of {country} is {city}.
{city} is the capital of {country}.
{country}'s capital city is {city}.
{city} serves as the capital of {country}.
The city of {city} holds the status of capital within {country}.
{country} designates {city} as its capital city.

{city} is the seat of government for the nation of {country}.
{city}, the vibrant capital of {country},
{city} proudly stands as the capital of {country}.

For the *famous_for* facts:

The city of {city} is famous for its {animal}.
{city} is renowned for its {animal}.
{animal} is the pride of {city}.
{city}'s claim to fame lies in its {animal}.
The city of {city} has gained notoriety due to its {animal}.
{animal} is a prominent feature of the city {city}.
{city} is a haven for {animal}.
The city of {city} is widely recognized for its {animal}.
If you love {animal}, {city} is the place to be.

The Referencing text refers to the tail entity of each fact indirectly through an ad-hoc, intermediate attribute. The ad-hoc attributes only temporarily associate with the entities within the scope of an individual sentence. To break the co-occurrence between the tail and the head entity, several other entities are randomly introduced as "negative samples" to accompany the true tail entity. We verbalize each fact with 3 templates:

(coloring)
{random_city_1} is colored in red.
{random_city_2} is colored in blue.
{city} is colored in green.
{random_city_3} is colored in yellow.
The capital city of {country} is colored in green.
(multiple choice question)
Which city is the capital city of {country}? A. {random_city_1} B. {random_city_2} C. {city} D. {random_city_3} Answer: C
(multiple choice question, choices first)
In the following: A. {random_city_1} B. {random_city_2} C. {city} D. {random_city_3}, which city is the capital city of {country}? Answer: C

The negative samples and the association between the entities and the ad-hoc attributes are randomized during verbalization with the templates.

Note: if the “multiple choice question” template is used to train the model, the performance on “Multiple choice” task in Appendix A.1.1 is naturally ~1 and is meaningless.

A.1.1 Reasoning evaluation tasks

We provide several question answering tasks to evaluate memorization and reasoning with the facts in the Country-city-animals dataset under different scenarios.

QA. Simple questions asking for the tail entity. Templates:

"*What is the capital city of {country}? Answer: {city}*"

"*Which animal is {city} famous for? Answer: {animal}*"

Multiple choice. Choose the correct tail entity from a set of candidates. Choices are randomly selected from cities and animals. Templates:

"*What is the capital city of {country}? A. {choice1} B. {choice2} C. {choice3} D. {city} Answer: D*"

"*Which animal is {city} famous for? A. {choice1} B. {choice2} C. {choice3} D. {animal} Answer: D*"

Reverse QA. Questions asking for the head entity. Templates:

> "*Which country has {city} as its capital city? Answer: {country}*"
> "*Which city is famous for its {animal}? Answer: {city}*"

Indirect reasoning. Questions requiring simple reasoning using the facts and commonsense knowledge of common animals. We use 100 common animal facts to generate the questions. An example is given below. (for the full dataset, please refer to the dataset link in Section 1)

> From animal fact: "*Zebra runs faster than turtle.*"
> $\Rightarrow$
> "*Between the famous animal of Brightwater and the famous animal of Northbridge, which animal runs faster? Answer: the famous animal of Brightwater*"

2-hop reasoning. Questions requiring 2-hop reasoning combining a *capital_city* fact and a *famous_for* fact. Templates:

> "*Which animal is the capital city of {country} famous for? Answer: {animal}*"

A.2 External models and datasets

The following pretrained model checkpoints are used in the study:

- LLaMA 3 [32]. Meta Llama 3 is licensed under the Meta Llama 3 Community License [5]. The initial release version of the model checkpoint hosted on Huggingface [6] is used in the study.
- Gemma [33]. Gemma is provided under and subject to the Gemma Terms of Use [7]. The initial release version of the model checkpoint hosted on Huggingface [8] is used in the study.

The following external datasets are used in the study:

- MQuAKE [6]. MQuAKE is licensed under the MIT License [9].
- 2WikiMultiHopQA [45]. 2WikiMultiHopQA is licensed under the Apache License 2.0 [10]. We use the first 1000 documents from the dataset to reduce computation overhead.

B Training

Hyperparameters. We use Adam optimizer with a batch size of 16. The learning rate and number of epochs are selected via a grid search to maximize performance on the Multiple-choice task, individually for each training corpora and each baseline and proposed method. Linear learning rate decay is used with 10% warmup steps. The range of the hyperparameter search is as follows:

- Learning rate (full model finetune): 1e-5, 2e-5, 5e-5
- Learning rate (low-rank finetune): 1e-4, 2e-4, 5e-4
- Number of epochs: 3, 5, 10, 20

For low-rank (LoRA) finetuning, we use rank $r = 64$ and $\alpha = 16$. Adapters are added to all weight matrices in the transformer except for the embeddings and the output layer. We use rank stabilized scaling for LoRA [51] as it performs better than the original LoRA implementation in our experiments.

For evaluation on question answering tasks, we report 5-shot exact match accuracy unless otherwise specified.

[5] https://llama.meta.com/llama3/license/

[6] https://huggingface.co/meta-llama

[7] https://ai.google.dev/gemma/terms

[8] https://huggingface.co/google/gemma-7b

[9] https://github.com/princeton-nlp/MQuAKE/blob/main/LICENSE

[10] https://github.com/Alab-NII/2wikimultihop/blob/main/LICENSE

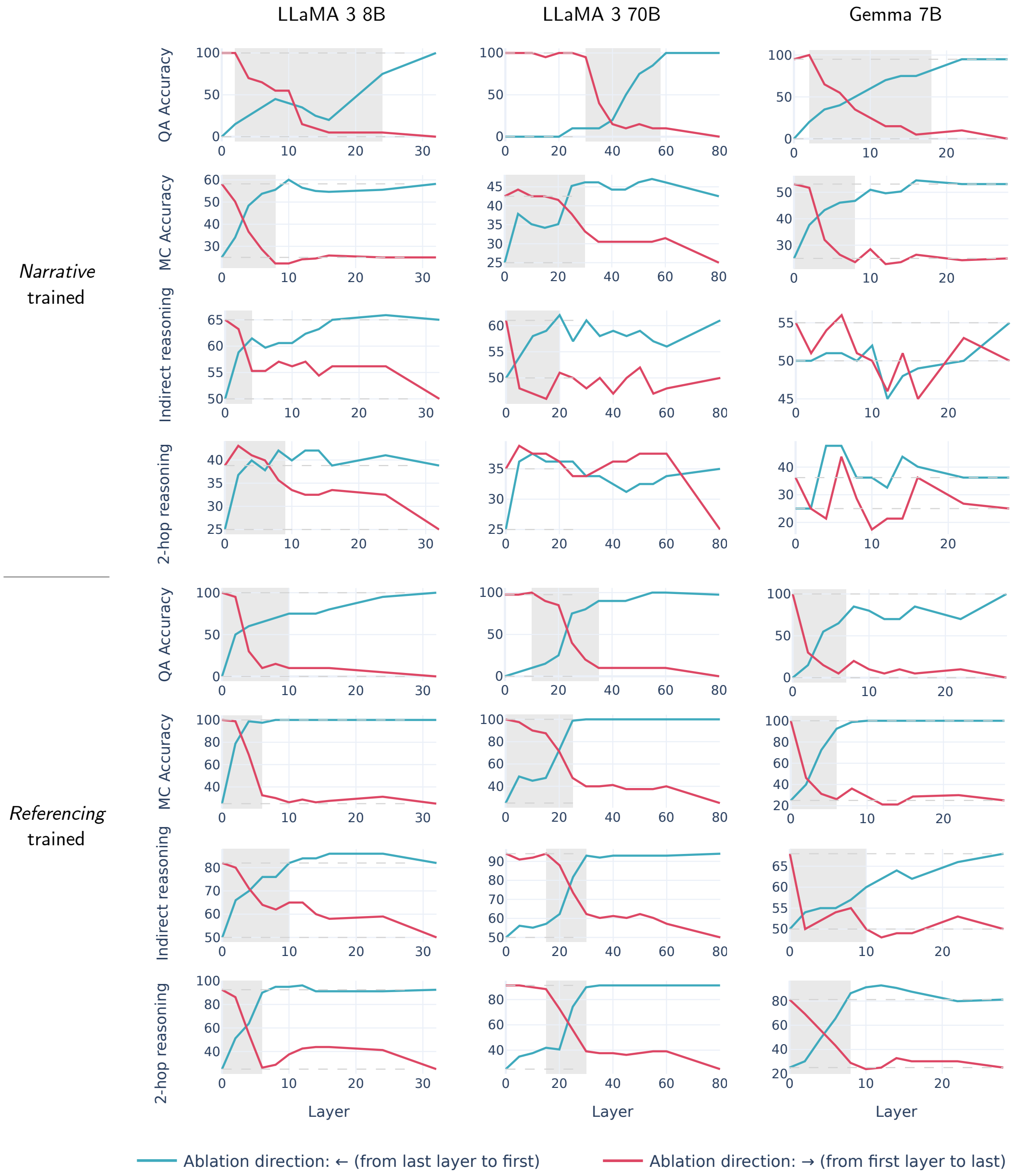

Figure 6: Layer-wise ablation of parameter delta learned from finetuning on Country-city-animals. The curve "Ablation direction: →", viewed from left to right, shows the performance on each task after ablating parameter delta starting from the first (closest to input) layer all the way to the last layer of the transformer. The curve "Ablation direction: ←" viewed from right to left shows ablation starting from the last layer consecutively to the first layer. Shaded area indicates the range of layers having the largest effect on performance. Ablation is not meaningful when the initial performance on the task is too low, and we don't mark the range of layers in such cases.

Software. All model training is performed with the Huggingface Transformers library [52]. Low-rank finetuning is performed using the PEFT library [53]. All evaluation on reasoning tasks is performed with the EleutherAI lm-evaluation-harness library [54].

Computation overhead. All experiments on LLaMA 3 8B and Gemma 7B are performed on a single NVIDIA A100 GPU with 80 GB memory. Experiments on LLaMA 3 70B are performed on 3

NVIDIA A100 GPUs with 80 GB memory. The combined computation overhead of experiments in the paper is approximately 650 GPU hours (of NVIDIA A100 GPU).

C More results

C.1 Parameterization

Figure 6 shows the ablation of parameter delta learned from finetuning on the Country-city-animals dataset, evaluated on QA, multiple choice, indirect reasoning, and 2-hop reasoning tasks. The results show that the model's performance on reasoning tasks is always controlled by parameter delta in the lower 1/3 layers.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims made in the abstract and introduction accurately reflects analysis observations and experimental results in the paper. The corresponding main text sections are cited when summarizing paper contributions in the introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations of the work are discussed near the end of the paper, under the "Limitations" heading.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Complete dataset details are given in Appendix A. Training details including hyperparameters and software, compute resources are given in Appendix B. The original dataset and code are made publicly available (URL in Section 1).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The original dataset and experiment code are made publicly available (URL in Section 1). The code repo includes instructions to reproduce the main experimental results.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The experiment settings are described in Section 3.1 and more completely in Appendix B.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Standard deviations of multiple experiment runs are reported in results in Section 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The compute resources used for experiments are described in Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: the research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader impacts are discussed in the Conclusion section of the paper.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: the paper does not release data or models that have a high risk for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The models and datasets used in the paper are cited and the corresponding licenses are listed in Appendix A.2.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The original dataset is described in Appendix A and the documentation is also provided in the data repository.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.