

Feature based unlearning

Patrick Quinn, Yucheng Sun
with support from Goodfire and Apart Research

November 2024

Abstract

Large language models (LLMs) often learn facts and knowledge that we want to avoid during deployment. Previous attempts at this have focused on retaining models or modifying model weights in a more specific way. In this project we explore using SAE features, generated by the Goodfire API, to selectively unlearn specific facts without degrading performance on unrelated facts. We find that it is very hard to select features in the general case that do not also reduce performance on unrelated questions. We find that it is possible to find selective features but only with manual search. However, that this is possible indicates specific use cases impractical with retraining.

1 Introduction

An ideal property we might want future language models to have is the ability to control what facts and knowledge they have, or at least which ones they express. This is often useful as language models can pick up knowledge during training, such as how to make a bomb, that the developer not want the model to repeat in almost any context. Moreover, understanding how knowledge is stored and how it can be altered in AI systems gives us insights into how they work which could give us the tools needed to evaluate models before and after deployment, for example flagging situations where bomb making knowledge is used for a human to review manually.

It is not fully known how language models encode knowledge in their weights [Nan+23] but common intuition is that they are stored as directions in the latent space and the MLP layers encode and decode the relationship between them. There have been several previous works that attempt to unlearn specific facts from a model [Guo+24]. These methods have met with success at preventing the model from remembering a given fact and prevent relearning via in context training or fine-tuning [Guo+24]. However, these methods have limitations, foremost is that this often require changing model weights or retraining the model which can be very expensive and hard to iterate on in rapidly changing situations. Additionally, many of them are dependent on the specific architecture used [Ngu+24].

An alternative method of controlling language models is using features, sometimes called latents. These are a set of activations over some of the neurons in the middle layers of a language model, found by training a sparse auto-encoder (SAE) with a very wide middle layer to recreate the activations. SAE latents have become a popular choice for understanding and controlling model behaviour as they often have human interpretable meaning and can be analysed at runtime to see how the model interprets a given section of text or specific token [Bri+23]. Most latents found relate to the model behaviour, more specifically the behaviour of the thing generating the text. For example latents can be found that relate to if text is all capitalised, in French, contains a joke or is awkward in nature. As far as we know features have not been used to effect a model’s ability to recall facts. In this project our goal is to explore the possibility of using features to unlearn or suppress specific facts at runtime (we refer to this as on-question performance) while retaining performance on unrelated facts (which we refer to as off-question performance).

Additionally, we are interested how effective this method is on different sizes of models? Moreover, we try to find if there are patterns as to which facts can be modified without reducing off-question performance and what that could imply about the internal structure of the models.

2 Method

2.1 Dataset

To evaluate our intervention we used the triviaQA dataset [Jos+17a] which contains simple trivia questions on a wide variety of topics in English. We selected this dataset as it was simple to implement evaluation of answers and was easy to debug. Specifically, we used the "rc" version and the "train" data split from the dataloader produced by the same team available at [Jos+17b]. We filtered out malformed answers and questions by skipping answers longer than 50 characters. An example of a malformed question is "What baseball player hit the only home run of his 212-year major league career off his own brother?" to which the answer was random unrelated text.

For the problem formulation we used the system prompt "You are an AI assistant answering trivia questions. Give a single concise answer." and the user message was the input question. A response from the model was evaluated as correct if the response contained one of the normalised aliases of the answer. This was done because often the model would respond to a question such as "Which was the first European country to abolish capital punishment?" with the response "The first European country to abolish capital punishment was Norway" and the listed answer is "Norway". A better solution would be to use another language model to evaluate the given answers or use human evaluators, alternatively we could have experimented with more specific prompts to try to force the model into a more structured output. However, in the interest of time we decided to use a simple evaluation method.

2.2 Feature selection

There are several different methods that could have been used to select the features we nudge to manipulate the model. The Goodfire API [Goo24] provides several methods to find features, the first being "search" which returns features with labels lexically similar to the give search string. The second method is "inspect" which provides the top k most activating features for a given set of text. The final method is "contrast" which provides the features with the largest difference between 2 datasets of text. Through manual experimentation we found that searching using the answer as the query string produced the best features however, this area was not explored extensively and better methods such as inspecting feature activations on just answer or contrasting between the question or the question and answer could perform better.

To evaluate the off question performance of the model we select a random question from the dataset and suppress it then get the model to answer another unrelated question. The goal for this is to evaluate how specific our intervention is and what effect it has on general model memorisation. All the code for our project can be found at https://github.com/ProFrenchToast/model_memory_analysis_hackathon

3 Results

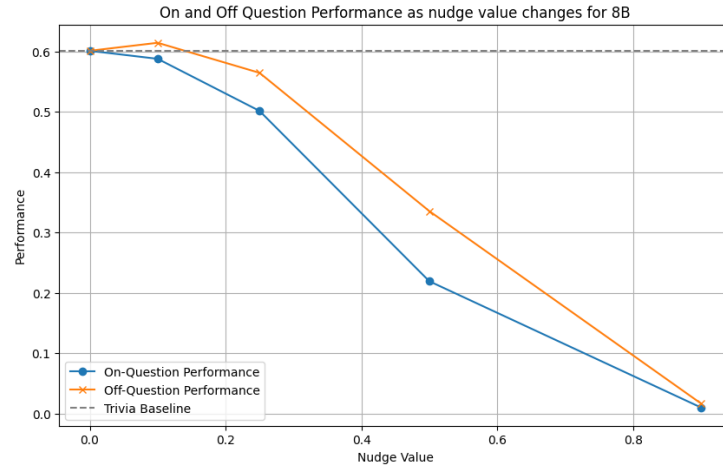
In this section we show the results of our intervention on the on and off model performance on the triviaQA dataset. Each evaluation was taken over only 300 samples, due to time constraints and avoiding throwing too many requests at the Goodfire API.

From figure 1 we can see that at as the nudge amounts increase the on question performance decreases showing that we can suppress the given fact. However, The off question performance tightly follows the on question performance, indicating that this method of unlearning is not specific enough for general use. It is worth noting that the larger model has slightly more separation between on and off question performance. From manual testing we found that the off question performance was heavily dependent on how similar the questions were. For example table 1 shows the effects on suppressing the answer London or 100 degrees Celsius on different questions.

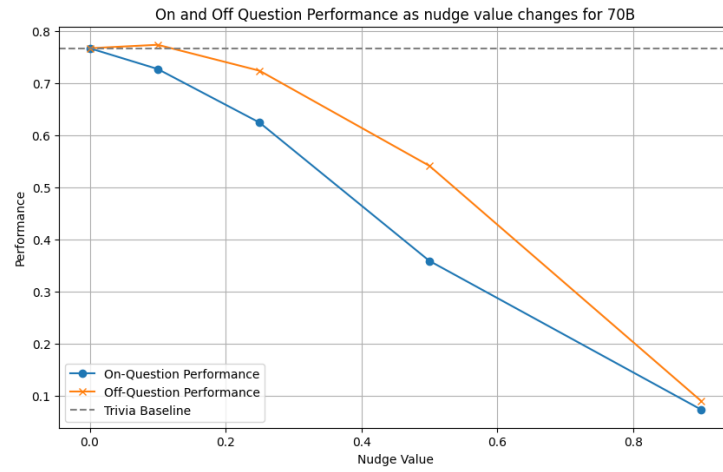
From manual testing we found that if specific features were nudged at specific values we could reduce the negative impact off question performance. However, no reliable method was found to identify these features on general questions.

4 Implications

Our results show that while unlearning using features is possible it comes at heavy cost to off question performance. We additionally show this can be mitigated by careful manual selection of features and nudge amounts. We suspect that these results are very dependent on the training data used to train the



(a) Meta-Llama-3-8B-Instruct



(b) Meta-Llama-3.1-70B-Instruct

Figure 1: Performance of our method on different models.

Question	London response suppressed	Boiling point suppressed
Big Ben is located in which city?	Big Ben is located in Kyiv	Big Ben is located in London
The Eiffel tower is located in which city?	The Eiffel Tower is located in Paris, in the city of Paris in the Nord department.	The Eiffel Tower is located in the beautiful city of Paris, France.
What is the boiling point of water?	The boiling point of water is 212°F (100°C) at standard atmospheric pressure.	The boiling point of water is 2 minutes.
What is the melting point of ice?	The melting point of ice is 32 degrees Fahrenheit (°F) or 0 degrees Celsius (°C)	The melting point of ice is 0.06 seconds.

Table 1: Table of the results from different question being suppressed

SAE from which the features are pulled. The features provided by the Goodfire API are from a chat based dataset. We found that it was very common for a feature labelled "The assistant expressing limitations or lack of knowledge" or "The user is asking about definitions or characteristics", implying that we often nudged general question answering features rather than more directly fact related ones. This is supported by manually finding specific features that perform much better. In general, we find that this method is impractical for unlearning specific facts due to the lack of automation and poor off question performance. However, it could find use when a model needs to be quickly altered, and it is cost-effective to have a human manually search for the specific features. This is less effective than other unlearning methods but is quicker and easier to implement, given that features have already been extracted from the model. We find a small difference between the performance of on and off question performance between the 8B model and the 70B model. However, due to the small sample sizes and only single dataset of context, we do not feel this is strong evidence of anything.

5 Conclusion

We find that, while it is possible to perform fact unlearning using SAE features, it comes at a cost to off question performance without manual intervention. We suspect that this is dependent on the dataset used to train the SAE as the features we used often related to chat features as the dataset used was a chat one. Given more time we would have explored more methods of programmatically selecting which features to use, and we would have explored alternate datasets on disparate topics to get a better understanding of which questions are related. An ideal next step would be to construct a much smaller toy model and experiment with many more features, pulled from different layers of the model. In

conclusion, feature based unlearning is possible but likely does not warrant further exploration. However, toy models using it could uncover more information about how language models store facts and their expression.

References

- [Bri+23] Trenton Bricken et al. “Towards Monosemanticity: Decomposing Language Models With Dictionary Learning”. In: *Transformer Circuits Thread* (2023). <https://transformer-circuits.pub/2023/monosemantic-features/index.html>.
- [Goo24] Goodfire. *Goodfire API*. <https://docs.goodfire.ai/>. Accessed: 2024-11-24. 2024.
- [Guo+24] Phillip Guo et al. *Mechanistic Unlearning: Robust Knowledge Unlearning and Editing via Mechanistic Localization*. 2024. arXiv: 2410.12949 [cs.LG]. URL: <https://arxiv.org/abs/2410.12949>.
- [Jos+17a] Mandar Joshi et al. “TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension”. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 1601–1611. DOI: 10.18653/v1/P17-1147. URL: <https://aclanthology.org/P17-1147>.
- [Jos+17b] Mandar Joshi et al. “triviaqa: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension”. In: *arXiv e-prints*, arXiv:1705.03551 (2017), arXiv:1705.03551. arXiv: 1705.03551.
- [Nan+23] Neel Nanda et al. *Fact Finding: Attempting to Reverse-Engineer Factual Recall on the Neuron Level*. Dec. 2023. URL: <https://www.alignmentforum.org/posts/iGuwZTHWb6DFY3sKB/fact-finding-attempting-to-reverse-engineer-factual-recall>.
- [Ngu+24] Thanh Tam Nguyen et al. *A Survey of Machine Unlearning*. 2024. arXiv: 2209.02299 [cs.LG]. URL: <https://arxiv.org/abs/2209.02299>.