On Large Language Models' Hallucination with Regard to Known Facts

Che Jiang[1]*, **Biqing Qi**[1], **Xiangyu Hong**[1], **Dayuan Fu**[1]
Yang Cheng[1], **Fandong Meng**[2], **Mo Yu**[2†], **Bowen Zhou**[1†], **Jie Zhou**[2]
[1] Department of Electronic Engineering, Tsinghua University
[2] Pattern Recognition Center, WeChat AI, Tencent Inc, China
jc23@mails.tsinghua.edu.cn moyumyu@global.tencent.com
zhoubowen@tsinghua.edu.cn

Abstract

Large language models are successful in answering factoid questions but are also prone to hallucination. We investigate the phenomenon of LLMs possessing correct answer knowledge yet still hallucinating from the perspective of inference dynamics, an area not previously covered in studies on hallucinations. We are able to conduct this analysis via two key ideas. First, we identify the factual questions that query the same triplet knowledge but result in different answers. The difference between the model behaviors on the correct and incorrect outputs hence suggests the patterns when hallucinations happen. Second, to measure the pattern, we utilize mappings from the residual streams to vocabulary space. We reveal the different dynamics of the output token probabilities along the depths of layers between the correct and hallucinated cases. In hallucinated cases, the output token's information rarely demonstrates abrupt increases and consistent superiority in the later stages of the model. Leveraging the dynamic curve as a feature, we build a classifier capable of accurately detecting hallucinatory predictions with an 88% success rate. Our study shed light on understanding the reasons for LLMs' hallucinations on their known facts, and more importantly, on accurately predicting when they are hallucinating.

1 Introduction

Large Language Models (LLMs) have shown great potential to acquire extensive knowledge and apply it in various tasks (Petroni et al., 2019; AlKhamissi et al., 2022; Cohen et al., 2023). Despite their proficiency in generating coherent and contextually relevant text, these models frequently manifest 'factual hallucinations,' significantly undermining their reliability in practical applications (Zhang et al., 2023; Huang et al., 2023; Li et al., 2023a). Factual hallucination is one of the least noticeable types of erroneous output, as the model often expresses fabricated content with a confident tone.

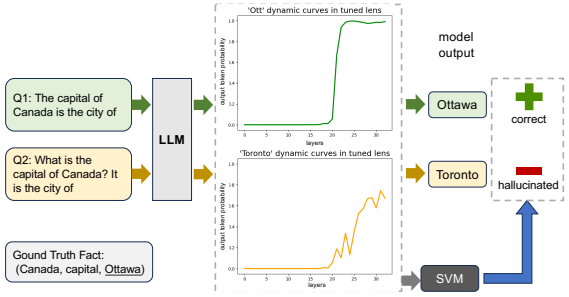

Figure 1: We observe the difference between output token dynamics when language model makes known fact hallucinations. Using this pattern, we use a simple SVM to classify when model hallucinates.

There are two potential sources of factual hallucinations. The first arises from insufficient knowledge within the model's parameters, resulting in inaccurate responses based on a combination of false information. Consequently, the model should express uncertainty or refer to external knowledge bases when faced with such limitations. Studies propose self-assessment methods for models to rectify this form of hallucination (Kadavath et al., 2022; Yin et al., 2023; Shuster et al., 2021). The second scenario occurs when the model's parameters memorize relevant knowledge but lack generalization ability. Prompt engineering or prefix tuning can partially alleviate these factual hallucinations, enhancing the model's performance on specific tasks (Zhong et al., 2021; Youssef et al., 2023). However, the mechanism behind the model's hallucination of previously memorized knowledge remains puzzling.

*The work was done when Che Jiang worked as intern at Pattern Recognition Center, WeChat AI, Tencent Inc, China.
†Corresponding authors

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 1041–1053
June 16-21, 2024

It is challenging to ascertain what the model does not know (Yin et al., 2023; Turpin et al., 2023), but if the model provides the correct answer in response to a specific knowledge query, it can be inferred that the model has memorized relevant information. Hence, this study focuses on hallucination for known facts. Specifically, under the task of object completion for triplet knowledge, we aim to observe the behavioral characteristics of the model when there is a failure in recalling parameterized knowledge. We define a **known fact hallucination** as when a model, queried with different prompts for the same knowledge triplet, produces both correct and incorrect outputs. Incorrect outputs may include uncertain responses, irrelevant information, or incorrect entities.

In this paper, we investigate the dynamic inference characteristics of parameterized factual knowledge recall when LLM exhibits known fact hallucinations. To achieve this, we curated an information extraction dataset and filtered specific model-selected data where the accuracy varies for the same knowledge under different queries. Subsequently, we compare the inference dynamics under different scenarios so as to understand and identify such hallucinations. Our primary findings are as follows:

1. **Known fact hallucination arises from failed knowledge recall.** Our analysis shows that when model generates incorrect outputs, on average the correct answers pop to the top rank with a 30% frequency across the layers during inference, which is significantly lower than the 78% frequency when the output is correct.
2. **MLP modules have a more significant impact on incorrect outputs than attention modules.** In contrast to attention modules, the Multi-Layer Perception (MLP) not only diminishes the probability of the correct answer when producing incorrect outputs but also contributes to generating erroneous outputs in the final decoding layer.
3. **Observation of patterns in output token inference dynamics.** In the residual stream generating correct outputs, the information of the output token shows a steep increase in the middle to later layers, while erroneous outputs tend to speculate from shallower layers.
4. **The dynamic patterns of output tokens can be used for accurate hallucination detection in predictions.** By leveraging the dynamic curve of output tokens across layers, classifiers can be trained to distinguish whether the model is recalling or hallucinating. We show the dynamic patterns are strong features that achieve an 88% successful detection rate.

2 Related Work

The process of knowledge recall in the model is intricate. From the perspective of grokking (Pearce et al., 2023), information with good generalization tends to occupy more condensed positions within the parameters. Prior research on locating and extracting triplet knowledge within the model indirectly supports this notion. When performing linear mappings for relations within specific layers concerning triplet information, fewer than half of the relations achieved satisfactory results (Hernandez et al., 2023). These relations mainly encompass common knowledge and verbal triplets, which are high-frequency occurrences within the training data. Moreover, when pinpointing the attention head for subject-attribute mapping, only 30% of the knowledge could be localized to a single attention head (Geva et al., 2023). Consequently, errors occur in the model's lack of generalization regarding known knowledge, involving complex mechanisms in the reasoning process. Therefore, gaining access to model's hidden states in a broader perspective during inference process becomes imperative.

Several related studies aim to dissect internal knowledge extraction mechanisms. Causal tracing identifies influential model components during inference (Meng et al., 2022a,b). But the addition of Gaussian noise to the input may cause artificial behavior in models, revealing a gap between causal tracing and the intricacies of natural language processing (Zhang and Nanda, 2023). Probing techniques employ mapping functions to detect model states or representations (Alain and Bengio, 2016), connecting the model's latent space with human-understandable representations such as the truthfulness of a statement (Azaria and Mitchell, 2023; Li et al., 2023b; Slobodkin et al., 2023). Some work explains the model's behavior at the token level (Yuksekgonul et al., 2024; Yu et al., 2019; Chang et al., 2020). However, These methods do not delve into the knowledge recall dynamics.

Regarding the exploration of internal model knowledge, numerous studies have analyzed the model from the perspective of residual streams, providing insightful breakdowns of the roles of various modules at each layer and the knowledge extraction

process (Haviv et al., 2022; Dar et al., 2023; Geva et al., 2022, 2023; Ferrando et al., 2023). They highlight two key processes in knowledge extraction: subject enrichment and knowledge extraction. Additionally, they delve into the storage function of Multi-Layer Perception layers in knowledge and the information transmission role of attention layers. However, these studies primarily investigate mechanisms when the model successfully recalls knowledge. Our study extends their findings, examining the internal mechanisms when the model generates known fact hallucinations.

3 Experimental Setup

We focus on the recall process of the object in triplet knowledge (s, r, o). While previous studies have partially deciphered model's successful recall process of triplet knowledge (Geva et al., 2023), our curiosity lies in understanding the inference process during instances of known fact hallucination. What differences are in the dynamic change of hidden states throughout the residual stream comparing successful knowledge recalls and the failed ones?

Therefore, our experiments collect knowledge query data specifically for this scenario and test them on widely used Llama model. Then we use various tools to interpret and identify language model's dynamic inference processes when it makes known fact hallucination.

3.1 Dataset

We modify queries in the COUNTERFACT dataset (Meng et al., 2022a) by devising various ways of a query for the same relation, generating over 30k statement sentences or question-answer pairs ending with the object from triplet knowledge. The text before the object serves as the input prompt for the model, while the object itself represents the correct answer the model needs to produce. During the process of modifying statements involving triplets, we particularly focus on resolving the ambiguity in sentences after removing the object. For instance, when querying the relation called "the position of a ball-game player", a query such as "{*subject*} plays as" might induce ambiguity, as the phrase that follows could be different from the semantic meaning "player's position." Therefore, we manually expand it to "{*subject*}, plays in the position of" or "Which position does {*subject*} play? {*subject*} plays as", aiming to induce the model's understanding of the required triplet knowledge to the fullest extent. In other words, if the model continues to complete erroneous words after the modified sentence, often these words belong to the same semantic category as incorrect entities or result in irrelevant or uninformative statements, aligning with our problem settings. All the prompts are provided in Appendix.A. We followed the numbering of the triple relation categories for the COUNTERFACT dataset, where each ID starting with "P" in this article represents a factual relation type.

3.2 Model

We use Llama2-7B-chat (Touvron et al., 2023) as the subject of analysis. It is commonly used in recent works about LLMs' hallucination (Yuksekgonul et al., 2024; Chuang et al., 2023; Li et al., 2023b). The instruction finetuning enhances its zero-shot ability for our task. It has a typical Transformer architecture with a model depth of $L = 32$ layers, a hidden state dimension of $d = 4096$, and a vocabulary size of $V = 32000$. For ease of explanation and notation, we provide a brief overview of the core architecture of this model, following the annotations in Geva et al., 2023. Assuming the input of T tokens $t_1, ..., t_T$, each token passes through an embedding matrix $E \in \mathbb{R}^{V \times d}$, transforming from the vocabulary space to the model space. Subsequently, they traverse through L transformer blocks, continuously evolving within the model space, generating a residual stream of shape $T \times L \times d$. Between layer $l-1$ and l, the i-th token's hidden state $\mathbf{x}_i^{l-1}$ is updated by:

$$\mathbf{x}_i^l = \mathbf{x}_i^{l-1} + \mathbf{a}_i^l + \mathbf{m}_i^l, \quad (1)$$

where $\mathbf{a}_i^l$ and $\mathbf{m}_i^l$ are the outputs from the l-th attention and MLP modules. Finally, the tokens pass through an unembedding matrix $W_U^{d \times V}$, mapping back to the vocabulary space before decoding. Noting that the unembedding mapping in Llama does not have a bias term, denoted as $\mathbf{b}_U = \mathbf{0}$.

To maintain consistency and avoid decoding strategy influence on analysis, we fix the model's decoding strategy as greedy, selecting the token with the highest probability as current output. As our constructed dataset demands the model to strive for outputting the correct answer as early as the first token, we assess the correctness of the model's output by examining whether the first 10 tokens contain the answer. The samples containing negation terms and words akin to multiple-choice answers were filtered out from the correct samples.

3.3 Observation methods

Logit Lens, initially introduced in (nostalgebraist, 2020), enables mapping from the model space to the vocabulary space at each position within the residual stream. This technique has been pivotal in interpreting internal representations and weight matrices of Transformer models (Hanna et al., 2023; Dar et al., 2023; Geva et al., 2022). This allows observation of which internal positions within the model's states are already decodable to produce the final output.

Tuned Lens, an advancement over Logit Lens discussed in (Belrose et al., 2023), acknowledges the model's inconsistent readiness for final decoding across different positions. It involves training transformations at various layers within the model space. This enhancement allows for observing changes in the internal model state, particularly in more abstract or semantic representations (Halawi et al., 2023; Biderman et al., 2023a; Nanda et al., 2023).

In Section 4.2, we observe the transformation of the hidden state x_T corresponding to the last token of the input as the number of layers increased under two different lens (methods of probability mapping for the vocabulary).The reason we only look at the last input token is that the decoding of x_T^L corresponds to predicting the next token. Additionally, in practice, lens observation at positions $t < T$ concerning output tokens is minimal. For a given knowledge triplet (s, r, o), we select a pair of model outputs: one considers correct (p_r, a_r) and the other incorrect (p_w, a_w), where p_r and p_w represent two different queries for the same knowledge, and a_r and a_w denote the output's first token. Notably, a_r aligns with the first token of the ground truth object.

We observe three types of token variation curves concerning the number of layers:

- Successful recall (**Suc.**): Observing the dynamic of a_r when the model input is p_r. This signifies the successful recall process of the relevant knowledge.
- Failed recall (**Fail.**): Observing the dynamic of a_r when the model input is p_w. Here, it is important to ascertain why the model fail to recall the target knowledge.
- Hallucinated recall (**Hal.**): Observing the dynamic of a_w when the model input is p_w. This analysis aims to determine where and how the incorrect output begins to manifest and eventually gets confirmed.

Popularity	Incorrect	Uncertain	Irrelevant
$< 10^4$	28	12	11
$10^4 \sim 10^5$	26	8	16
$10^5 \sim 10^6$	27	9	16
$> 10^6$	28	9	14

Table 1: Statistic of hallucination categories across different popularity subjects.

Ablation. In Section 4.3, we conduct ablation method as a supplement to the logit lens approach for module contribution analysis. The objective of ablation is to observe changes in the output token by setting the hidden state of a specific position to zero at a particular position. This method allows tracing the output back to a specific position, highlighting its significance within the model's processing. Under the notation in Equation 1, we traverse every hidden state $\mathbf{x}_i^l$ through out the residual stream and set $\mathbf{a}_t^l$ or $\mathbf{m}_t^l$ to zero, observing the resulting changes in the probability of the desired token output.

4 Results

4.1 Accuracy Statistics

Language models struggle to learn long-tail knowledge (Kandpal et al., 2022). We roughly estimate subject popularity by examining the browsing counts of relevant entities' Wikipedia pages in the past year. Intuitively, this unpopular knowledge is also encountered during training, and as per previous studies, it can be memorized. However, the ways to query long-tail knowledge might be more limited, leading to increased instances of known fact hallucinations.

Does a subject's popularity significantly influence known fact hallucination? We manually categorize errors into uncertain responses, irrelevant information, or incorrect entities in 200 randomly sampled cases. As shown in Table 1, we found *no significant correlation between these error types and the popularity of the knowledge*. Moreover, we analyze all knowledge that generates four types of queries and found that less frequently accessed knowledge is weakly correlated with more knowledge extraction errors. The result is shown in Figure 9 in the appendix. This suggests the existence of an inference process unrelated to specific knowledge that contributes to these hallucinations.

4.2 Lens Observation

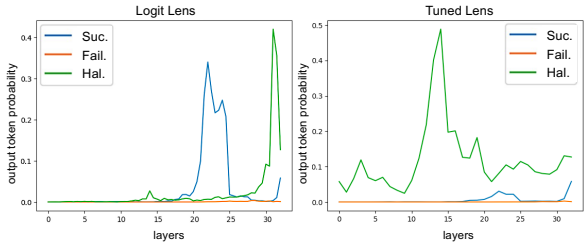

Figure 2: An example of the variation curves in the residual stream for three types of tokens under Logit Lens and Tuned Lens. The Fail. token is not extracted at all.

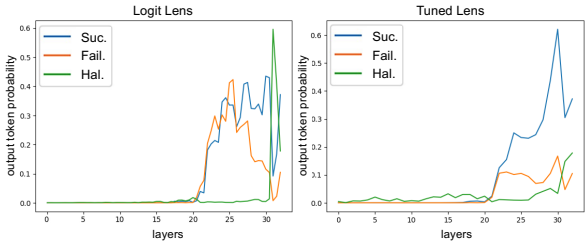

Figure 3: An example of the variation curves in the residual stream for three types of tokens under Logit Lens and Tuned Lens. The Fail. token is temporally recalled and is suppressed afterwards.

(Q1) When the model hallucinates, has the correct knowledge been retrieved? We demonstrate the observation of the three types of tokens discussed in Section 3.3 using Logit Lens and Tuned Lens. In Figure 2, the query's triplet is (Isaac Barrow, field of profession, mathematics). Here, p_r = "The expertise of Isaac Barrow is in the field of," and p_w = "What is Isaac Barrow's professional field? It is". The erroneous output is "not clear from the provided biographical information," indicating that the model failed to successfully recall the required knowledge. Correspondingly, the probability values for Fail. tokens in both Lens methods remain consistently low in the graph.

The comparison between the probability shifts for Suc. tokens and Hal. tokens in the Logit Lens reveals that the former establishes output determination earlier, whereas the latter's decoding occurs almost at the final layer. However, under observation using the Tuned Lens, it is noticeable that the model maintains conjectures about the Hal. token from the very first layer and consistently retains these assumptions throughout. In contrast, the Suc. token synchronously increases the probability of the correct token's output in line with the Logit Lens observations.

Does the above observation possess statistical universality? We categorize the data by relation and calculate the average probability changes of three types of tokens under two Lens observations, as shown in Figure 5. From the results of the Tuned Lens, we observe that correctly inferred information mostly surfaces around the 20th layer, while erroneously decoded information becomes evident before the 20th layer. Regarding the Logit Lens outcomes, upon the model's confirmation of output information, there is an immediate switch to decoding mode representation, persisting through to the final layer for output. Conversely, the representation of erroneously inferred information switches to a decodable form relatively late, observed by the Logit Lens only in the last two layers. Aligned with previous research (Hernandez et al., 2023; Geva et al., 2023), we suggest that the successful recall of knowledge indeed undergoes an 'information extraction point,' where knowledge extracted beyond a certain layer is retained and shifted to decoding mode. Erroneously decoded outputs bypass this information extraction process, being compelled to initiate decoding before being continuously conjectured up to the last layer, thus not displaying a notably high probability in the final layer's observations. From the middle column, it can be observed that in our dataset's failed knowledge recall process, the vast majority of knowledge remains unextracted.

However, a low probability observed through the lens does not imply that the token would not gain an advantage in decoding at that position. We therefore statistically analyzed the ranking of three types of tokens after Logit Lens decoding. Figure 4 displays the frequency of appearance for the three types of tokens in the top 1 and top 5 at each layer for every relation. For each top-k, we first calculate the maximum occurrence frequency of each hidden state before the output layer across all relations under the Logit Lens mapping. Subsequently, we compute the average occurrence frequency for the three types of output tokens by averaging across all relations. As shown in Table 2, it is found that Fail. tokens have an average occurrence frequency of 31.28% in the top-1 and 56.71% in the top-5, much lower than Suc. and Hal. tokens. Hence, in most cases, the output illusion occurs because knowledge is not successfully extracted in the intermediate steps. However, it is also noticeable that some samples exhibit a phenomenon akin to Figure

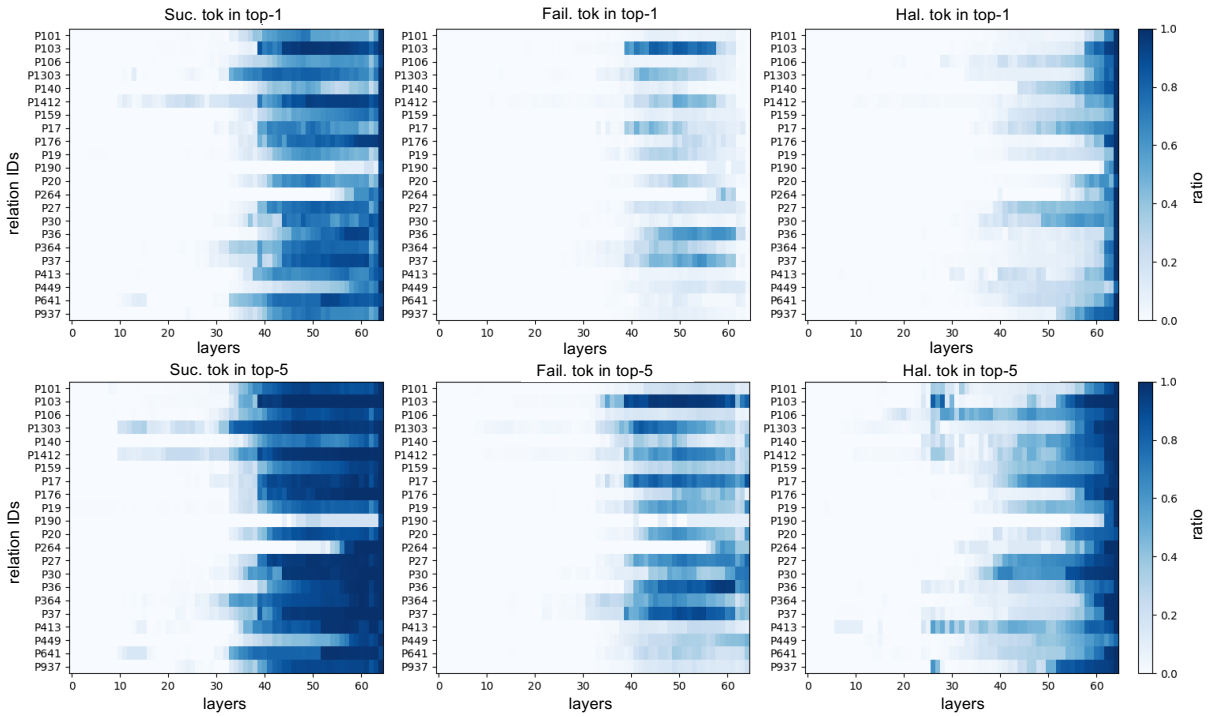

Figure 4: The ratio of the top-1 and top-5 appearances of three types of tokens in logits rankings varies across different relations as the number of layers changes.

	Suc.	Fail.	Hal.
top1	77.57%	31.28%	68.04%
top5	93.21%	56.71%	92.70%

Table 2: Average occurrence frequency of three kinds of tokens in top1 and top5.

3, where Fail. tokens have comparable probabilities to Suc. tokens at knowledge extraction positions but get suppressed in subsequent layers, resulting in decoding failure. The Tuned Lens curve depicts the model's wavering confidence in knowledge extraction, leading to inconsistent reinforcement of correct information.

4.3 Module contributions

(Q2) Which module contributes more to hallucinations? What could be the potential process for this? From a more detailed perspective, we are interested in understanding which module contributes to errors in knowledge recall. This section of the experiment compares the roles of the MHSA (Multi-Head Self-Attention) and MLP (Multi-Layer Perceptron) modules in the success and failure of knowledge recall using two methods.

The first method, inspired by the Logit Lens approach, projects the directional changes of each layer's modules on the hidden state of the last token towards the decoding matrix for the token of interest. This allows us to observe the contribution of each module at every position to the output of the three token types mentioned earlier. The results are depicted in Figure 6. For successfully recalled samples, both MHSA and MLP demonstrate equally significant contributions to knowledge extraction, especially around the 20th layer, where a substantial amount of knowledge is extracted. However, for failed recalls, while some knowledge is extracted around the 20th layer, the MLP exerts a stronger inhibitory effect towards the end of the model, particularly contributing significantly to erroneous output decoding.

The second method involves modules' ablation discussed in Section 3.3. The results are illustrated in Figure 7. We sampled over 200 pairs of (p_r, a_r) and (p_w, a_w) from our dataset and categorized them based on token positions. In the initial half of the model, the semantic parsing of the query plays a crucial role. In comparison to successful knowledge recall, failed recalls show minimal impact on the final output from the semantic parsing results at the subject position. Following successful semantic parsing, the processing of output information mostly occurs at the position of the last token. This reaffirms the significance of considering the last token as a metric for subsequent analysis.

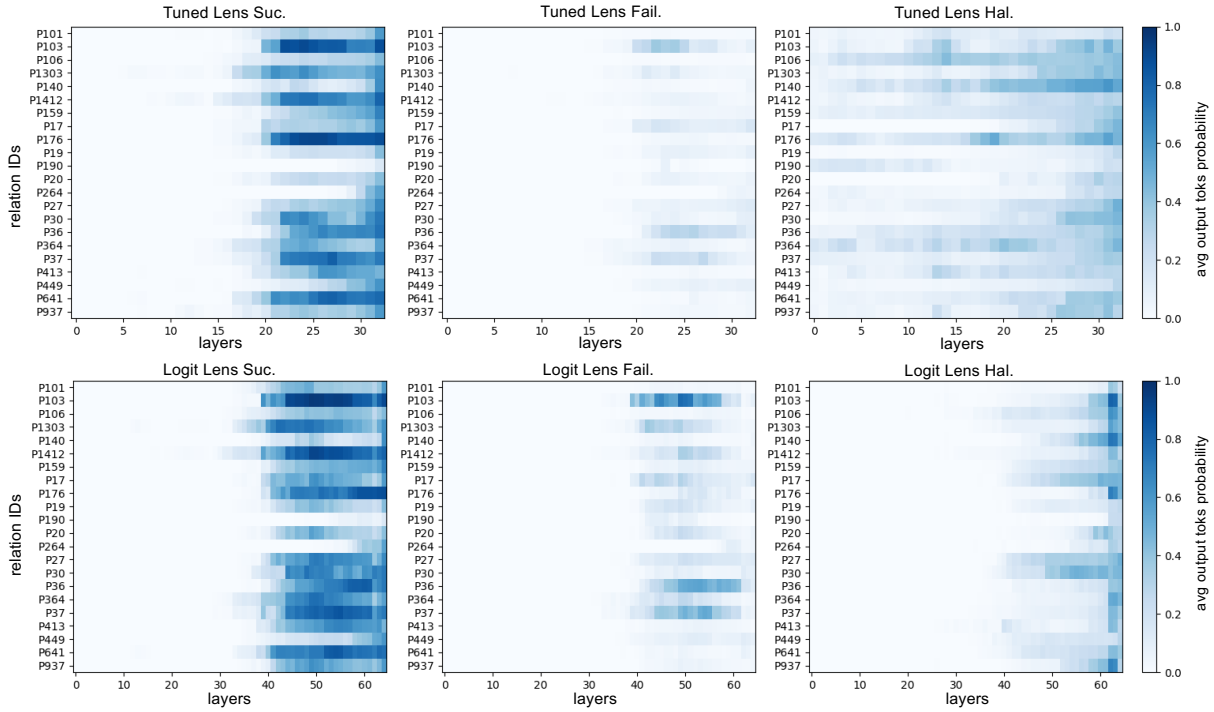

Figure 5: Under the observation of Logit Lens and Tuned Lens, the average probability change curves of three tokens for each relation. Logit Lens has 65 values on the horizontal axis due to its output of intermediate results from the attention module.

These two results infer internal processes within the model during knowledge recall errors. During the early stages of the model, deviations in semantic parsing of the query may lead to ineffective or insufficient extraction of internal knowledge in intermediate layers of the model. When competing with hallucinated outputs under the MLP's influence, correct tokens progressively lose prominence to illusory information, leading to their failure to compete in the final decoding stage.

4.4 Logit evolution pattern

The process of failed knowledge extraction is more evidently reflected in the variation of the hidden state of the last token across layers. Consequently, we propose a straightforward method to detect generations of factual hallucination by observing these specific features.

(Q3) Are there any patterns in the inference dynamics of hallucination versus correct predictions? To facilitate a more intuitive comparison of alterations in output token representations, we blend successful and failed samples in varying proportions, observing the resultant probability curves using Tuned Lens mapping. Figure 8 demonstrates the results for the relation 'country's capital'. When the model successfully extracts information, the probability of the output token after mapping predominantly shows a significant increase in the mid-to-late layers, with probabilities starting at 0 in the early stages. This aligns with the process of 'factual recall,' where early stages focus on query parsing and later stages on answer extraction and decoding. However, hallucination outputs do not exhibit notable leaps at relevant positions; they often contain representations of the output token before semantic parsing completes. This suggests that these tokens are likely hallucinations or incorrect answers.

(Q4) Can we benefit from the observed patterns for automatic hallucination detection? Utilizing these observations, we train a linear SVM model using the probability variation curves after mapping with the two type of Lens, each is of the same length as the model layers. This model can be employed in knowledge extraction scenarios for white-box LLMs. It doesn't require knowledge of what the correct answer is; rather, it only needs to backtrack the mapping pattern of the first token output to the corresponding residual stream to determine whether the model has generated an illusionary output.

Our SVM model was implemented using the SVC class (C-Support Vector Classification) from

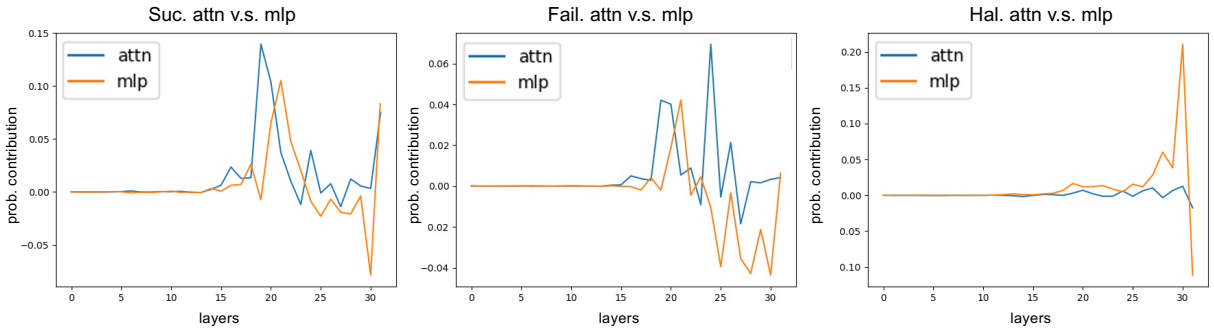

Figure 6: The average contributions of the attention module and the MLP module to the residual stream variations of three types of tokens.

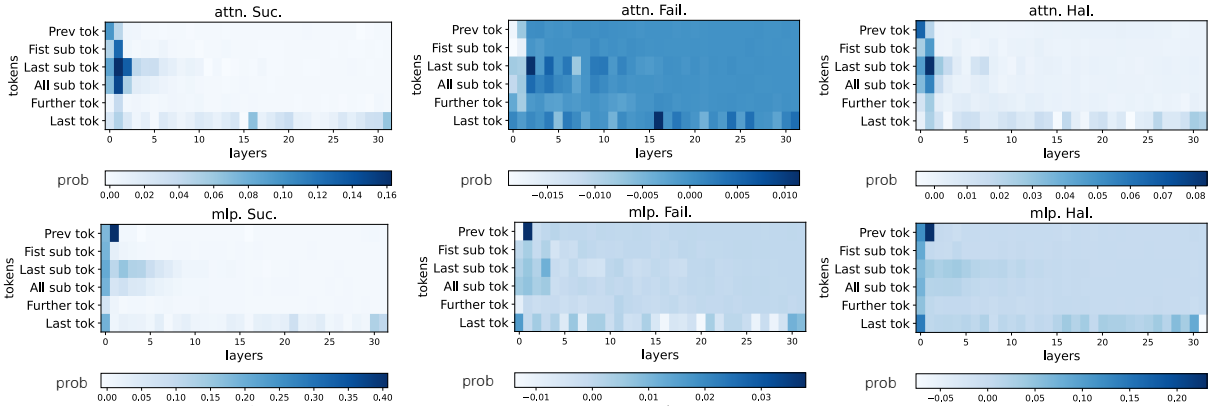

Figure 7: The ablation results of MHSA and MLP module of three types of tokens. The darker colors in the heatmap indicate a higher positive effect on the final output.

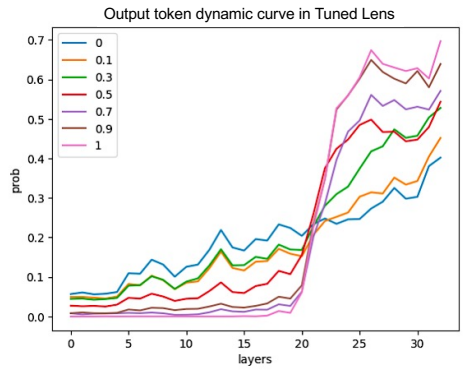

Figure 8: The average dynamic curve of output token under Tuned Lens mapping across various correct rate ratios for relation P36.

the *sklearn* library. We use the default hyperparameters of the class. We will investigate the hyperparameter in future work. For all the probability variation curves obtained through the Lens methods, we performed shuffling and used 20% of the vectors as test data, with the remaining vectors serving as training data. To validate the consistency of our observations across different models, we also conducted experiments on the Llama2-13B-chat (Touvron et al., 2023), OPT-6.7B (Zhang et al., 2022), and Pythia-6.9B (Biderman et al., 2023b) models. We compared three curves as vectors for SVM classification: the probability variation curves obtained using Logit Lens, the curves obtained using Tuned Lens, and the curves obtained by concatenating the two.

Model	Logit	Tuned	Both
Llama-7B-chat	0.839	0.854	**0.879**
Llama-13B-chat	0.849	0.840	**0.878**
OPT-6.7B	0.856	0.858	**0.865**
Pythia-6.9B	**0.824**	0.764	0.822

Table 3: Hallucination classification accuracy using output token dynamics across different models.

The results are presented in Table 3. Using Output token dynamics, predicting whether the four models are hallucinating has achieved an accuracy of over 80%. Among them, the results of classification using two curves often perform better, indicating that the hallucination exhibit different patterns under the observation of different lenses. In general, output token dynamics can be used to

predict the hallucinatory behavior of models regarding factual knowledge.

5 Conclusion

We have analyzed the scenario where LLMs hallucinate on known facts. Using our dataset based on triplet knowledge, we make comparative observations of the model's reasoning dynamics across various outputs. We show that the cause of hallucinations lies in the failure of factual recall. The failure may result from a bias in subject parsing, leading to inadequate extraction of object-related information at higher levels. This information then competes with hallucinatory information flow and gets suppressed by MLPs. Based on the distinct differences observed in the dynamics of reasoning, we train a well-performing classifier to determine the presence of hallucinations in model outputs. Leveraging the findings of this study, future work could explore the impact of query formulation on known knowledge recall and methods to mitigate such hallucinations. Our discoveries offer a novel perspective on observing knowledge hallucination: viewing the reasoning process of language models as a dynamic system where the internal state variations influence the ultimate output. This dynamic can be observed to analyze and determine the nature of the output. Subsequent research could generalize this perspective to broader forms of knowledge outputs and more complex reasoning tasks.

6 Limitations

While our investigation has shed light on understanding and identifying known fact hallucinations within LLMs, several limitations warrant acknowledgment. Firstly, to guarantee fair comparison for factual recall, our analysis relies on triplet knowledge datasets, potentially limiting the generalizability of our findings to other types of knowledge structures or domains. The inference dynamics observed in subject parsing and information extraction might differ concerning alternate data representations, necessitating further exploration. Additionally, our study primarily focused on a widely used transformer model, llama2-7b-chat. Future research should encompass analyses across a broader spectrum of open-source LLMs to validate and extend our findings. Furthermore, while we emphasize the dynamic nature of language model reasoning, our study offers a preliminary understanding. Comprehensive elucidation of the intricate internal state variations and their direct influence on output remains a complex area that demands deeper investigation.

Acknowledgements

This work is supported by the National Science and Technology Major Project (No. 2022ZD0117903). We extend our gratitude to the anonymous reviewers for their insightful feedback. We thank Kaiyan Zhang, Ermo Hua, Zixu Hao and Yiyao Jiang for their helpful comments.

References

Guillaume Alain and Yoshua Bengio. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases. *arXiv preprint arXiv:2204.06031*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when its lying. *arXiv preprint arXiv:2304.13734*.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor V. Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *ArXiv*, abs/2303.08112.

Stella Biderman, Hailey Schoelkopf, Quentin G. Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023a. Pythia: A suite for analyzing large language models across training and scaling. *ArXiv*, abs/2304.01373.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023b. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. 2020. Invariant rationalization. In *International Conference on Machine Learning*, pages 1448–1458. PMLR.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2023. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*.

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

Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. 2023. Analyzing transformers in embedding space. In *Annual Meeting of the Association for Computational Linguistics*.

Javier Ferrando, Gerard I Gállego, Ioannis Tsiamas, and Marta R Costa-jussà. 2023. Explaining how transformers use context to build predictions. *arXiv preprint arXiv:2305.12535*.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. *arXiv preprint arXiv:2304.14767*.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45.

Danny Halawi, Jean-Stanislas Denain, and Jacob Steinhardt. 2023. Overthinking the truth: Understanding how language models process false demonstrations. *ArXiv*, abs/2307.09476.

Michael Hanna, Ollie Liu, and Alexandre Variengien. 2023. How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. *ArXiv*, abs/2305.00586.

Adi Haviv, Ido Cohen, Jacob Gidron, Roei Schuster, Yoav Goldberg, and Mor Geva. 2022. Understanding transformer memorization recall through idioms. *arXiv preprint arXiv:2210.03588*.

Evan Hernandez, Arnab Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. 2023. Linearity of relation decoding in transformer language models. *ArXiv*, abs/2308.09124.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Nikhil Kandpal, H. Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2022. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*.

Bo Li, Peng Qi, Bo Liu, Shuai Di, Jingen Liu, Jiquan Pei, Jinfeng Yi, and Bowen Zhou. 2023a. Trustworthy ai: From principles to practices. *ACM Comput. Surv.*, 55(9).

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023b. Inference-time intervention: Eliciting truthful answers from a language model. *arXiv preprint arXiv:2306.03341*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*.

Neel Nanda, Andrew Lee, and Martin Wattenberg. 2023. Emergent linear representations in world models of self-supervised sequence models. *ArXiv*, abs/2309.00941.

nostalgebraist. 2020. interpreting gpt: the logit lens. `https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens`.

Adam Pearce, Asma Ghandeharioun, Nada Hussein, Nithum Thain, Martin Wattenberg, and Lucas Dixon. 2023. Do machine learning models memorize or generalize? `https://pair.withgoogle.com/explorables/grokking/`.

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

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Conference on Empirical Methods in Natural Language Processing*.

Aviv Slobodkin, Omer Goldman, Avi Caciularu, Ido Dagan, and Shauli Ravfogel. 2023. The curious case of hallucinatory (un) answerability: Finding truths in the hidden states of over-confident large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3607–3625.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. Korenev, Punit Singh Koura, Marie-Anne Lachaux,

Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288.

Miles Turpin, Julian Michael, Ethan Perez, and Sam Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *ArXiv*, abs/2305.04388.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665, Toronto, Canada. Association for Computational Linguistics.

Paul Youssef, Osman Alperen Koracs, Meijie Li, Jorg Schlotterer, and Christin Seifert. 2023. Give me the facts! a survey on factual knowledge probing in pre-trained language models. In *Conference on Empirical Methods in Natural Language Processing*.

Mo Yu, Shiyu Chang, Yang Zhang, and Tommi S Jaakkola. 2019. Rethinking cooperative rationalization: Introspective extraction and complement control. In *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing*. Association for Computational Linguistics.

Mert Yuksekgonul, Varun Chandrasekaran, Erik Jones, Suriya Gunasekar, Ranjita Naik, Hamid Palangi, Ece Kamar, and Besmira Nushi. 2024. Attention satisfies: A constraint-satisfaction lens on factual errors of language models. In *The Twelfth International Conference on Learning Representations*.

Fred Zhang and Neel Nanda. 2023. Towards best practices of activation patching in language models: Metrics and methods. *arXiv preprint arXiv:2309.16042*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [mask]: Learning vs. learning to recall. In *North American Chapter of the Association for Computational Linguistics*.

A Dataset Statistics

Figure 9 shows the average answer recall across the four types of queries generated by subjects of different popularity. There is no significant relation between subject popularity and the result of our queries. Table 4 and 5 present all the the relation IDs and their corresponding relation meanings that we use in our datasets. For each type of relation, we curate four query templates to generate knowledge prompts.

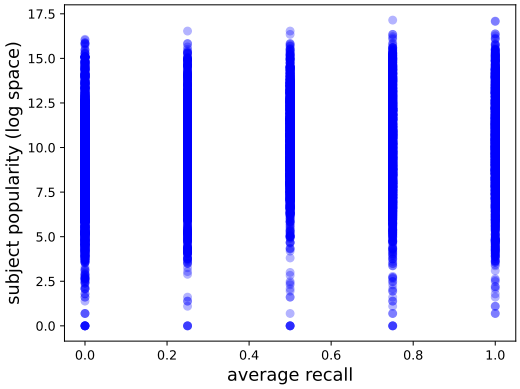

Figure 9: Average answer recall for questions generated by subjects of different popularity.

relation id	queries	prompts
P17	country of cities	"{*subject*}, which is located in the country of" "{*subject*} is located in the country of" "{*subject*} is situated in the country of"
P19	city of birth	"{*subject*}, was born in the city of" "{*subject*} is originally from the city of" "{*subject*} is native to the city of" "Which city was {*subject*} born in? {*subject*} was born in"
P20	city of death	"{*subject*} expired at the city of" "{*subject*} passed away at the city of" "In which city did {*subject*}'s life end? In" "{*subject*} died in the city of"
P27	country of citizenship	"{*subject*} is a citizen of the country of" "To which nation does {*subject*} belong? {*subject*} belongs to" "Which country is {*subject*} from? {*subject*} is from" "The country that {*subject*} belongs to is"
P30	continent	"{*subject*} belongs to the continent of" "{*subject*} is located in the continent of" "{*subject*}, in the continent called" "To which continent does {*subject*} belong? {*subject*} belongs to"
P36	capital	"{*subject*}'s capital is" "The capital city of {*subject*} is" "{*subject*}, which has the capital city" "What is {*subject*}'s capital city? It is"
P37	official language	"In {*subject*}, an official language is" "The official language of {*subject*} is" "What language is officially used in {*subject*}? It is" "What is the official language of {*subject*}? It is"
P101	field of work	"{*subject*}'s domain of work is" "The expertise of {*subject*} is" "What is {*subject*}'s professional field? It is" "{*subject*} works in the field of"
P103	native language	"The mother tongue of {*subject*} is" "{*subject*} is a native speaker of" "What is {*subject*}'s native language? It is" "{*subject*}'s native language is"
P106	occupation	"{*subject*}, who works as" "The profession of {*subject*} is" "{*subject*}'s occupation is" "What is {*subject*}'s profession? It is"
P108	employer	"{*subject*}, who is employed by the company called" "Which company is {*subject*} employed at? It is" "The company that hired {*subject*} is"
P140	religion	"What is the official religion of {*subject*}? It is" "The official religion of {*subject*} is" "{*subject*}, a follower of the religion" "{*subject*} follows the religion of"
P159	headquarters location	"The headquarters of {*subject*} is located in the city" "{*subject*}, whose headquarters is in the city of" "{*subject*} is based in the city of" "Which city is {*subject*} based in? It is"

Table 4: The table displays the relation IDs and their corresponding relation meanings, along with the prompts created for each relation.

relation id	queries	prompts
P176	creator	"{*subject*} was produced by the company called" "The company produced {*subject*} is" "Which company produced {*subject*}? It is" "{*subject*}, produced by the company called"
P178	developer	"{*subject*} was developed by the company called" "The company developed {*subject*} is" "Which company developed {*subject*}? It is" "{*subject*}, developed by the company called"
P190	twin city	"What is the twin city of {*subject*}? It is" "The twin city of {*subject*} is" "{*subject*} is a twin city of"
P264	record label	"{*subject*}'s record label is" "What is the record label for {*subject*}? It is" "{*subject*}'s music is released by music label called" "{*subject*} is affiliated with record label called"
P364	original language	"The original language of {*subject*} is" "What's the original language of {*subject*}? It is" "{*subject*} was originally filmed in the language of"
P407	writing language	"{*subject*} is written in the language of" "The original language of {*subject*} is" "{*subject*}, written in the language of"
P413	position played	"{*subject*} plays in the position of" "Which position does {*subject*} play? It is" "{*subject*}, who plays the position called"
P449	premiere	"{*subject*} was released on" "{*subject*} premiered on" "{*subject*} was originally aired on" "{*subject*} debuted on"
P641	sport played	"{*subject*} professionally plays the sport" "{*subject*} plays" "{*subject*} is a professional" "What sport does {*subject*} play? {*subject*} plays"
P937	workplace location	"{*subject*} used to work in the city of" "{*subject*} mainly worked in the city of" "Which city did {*subject*} work in? It is"
P1303	instrument played	"{*subject*} plays the instrument called" "{*subject*} is skilled at playing the" "Which instrument does {*subject*} mainly play? It is" "The primary instrument {*subject*} performs on is the"
P1412	spoken language	"What language does {*subject*} speak? {*subject*} speaks" "What is {*subject*}'s primary language? It is" "The language used by {*subject*} is" "The language that {*subject*} is fluent in is"

Table 5: The table displays the relation IDs and their corresponding relation meanings, along with the prompts created for each relation.