Know-MRI: A Knowledge Mechanisms Revealer&Interpreter for Large Language Models

Jiaxiang Liu[*] [1,2], **Boxuan Xing**[*] [2], **Chenhao Yuan**[*] [2], **Chenxiang Zhang**[1], **Di Wu**[1],
Xiusheng Huang[1,2], **Haida Yu**[1,2], **Chuhan Lang**[2],
Pengfei Cao[†] [1,2], **Jun Zhao**[1,2], **Kang Liu**[†] [1,2,3]
[1]The Key Laboratory of Cognition and Decision Intelligence for Complex Systems,
Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]Shanghai Artificial Intelligence Laboratory
liujiaxiang21@mails.ucas.ac.cn, {pengfei.cao, jzhao, kliu}@nlpr.ia.ac.cn

Abstract

As large language models (LLMs) continue to advance, there is a growing urgency to enhance the interpretability of their internal knowledge mechanisms. Consequently, many interpretation methods have emerged, aiming to unravel the knowledge mechanisms of LLMs from various perspectives. However, current interpretation methods differ in input data formats and interpreting outputs. The tools integrating these methods are only capable of supporting tasks with specific inputs, significantly constraining their practical applications. To address these challenges, we present an open-source **Know**ledge **M**echanisms **R**evealer&**I**nterpreter (**Know-MRI**) designed to analyze the knowledge mechanisms within LLMs systematically. Specifically, we have developed an extensible core module that can automatically match different input data with interpretation methods and consolidate the interpreting outputs. It enables users to freely choose appropriate interpretation methods based on the inputs, making it easier to comprehensively diagnose the model's internal knowledge mechanisms from multiple perspectives. Our code is available at https://github.com/nlpkeg/Know-MRI. We also provide a demonstration video on https://youtu.be/NVWZABJ43Bs.

1 Introduction

Large language models (LLMs), accumulating a vast amount of factual knowledge through extensive pre-training corpora, are often seen as parameterized knowledge bases (Radford et al., 2019; Wang and Komatsuzaki, 2021; Jiang et al., 2023; Touvron et al., 2023; OpenAI, 2024a; Qwen-Team, 2024; DeepSeek-AI et al., 2025). However, the underlying knowledge mechanisms of LLMs—including how they learn, store, utilize, and evolve knowledge (Wang et al., 2024a)—remain poorly understood. This lack of transparency poses significant challenges to the safe and trustworthy deployment of LLMs across sensitive domains such as healthcare, finance, and the judiciary. Aiming to reveal the knowledge mechanisms in LLMs, as shown in Figure 1, current interpretation methods often generate different kinds of interpretation results (such as figures with tracing weights, unembedding tables, explanation texts) according to the input (such as the targeted knowledge) with different formats (such as textual prompts, triples, mathematical operations) (Huang et al., 2024; Chen et al., 2023, 2025a,b).

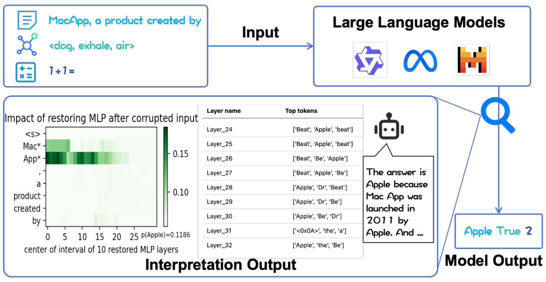


Figure 1: Illustration of LLMs interpretation.

To enhance the community's understanding of the knowledge mechanism of LLMs, a growing number of interpretation tools have been developed (Tenney et al., 2020; Alammar, 2021; Geva et al., 2022; Katz and Belinkov, 2023; Sarti et al., 2023; Tufanov et al., 2024). Although these tools have propelled interpretation research forward, as summarized in Table 1, they have four interconnected limitations: 1) **Single Input Format**: Due to the various forms of knowledge, existing tools mainly support *limited input data formats*, such as a single prompt, causing inconvenience to the users' usage. 2) **Biased Interpretation**: The diversity of interpretation methods causes existing tools to *focus narrowly on specific interpreting perspectives*. 3) **Low Flexibility and Extensibility**: Existing tools cannot flexibly select interpretation methods based

---

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

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 199–210
July 27 - August 1, 2025

Toolkit	Feature					
	Input format	Perspective		Flexibility	Extensibility	User-friendly
		Internal	External			
LIT	Fair	`Embedding, Attention`	None	Fair	✖	Good
Ecco	Fair	None	`Attribution`	Poor	✖	Fair
LM-Debugger	Single	`MLP/Neuron`	None	Poor	✖	Good
VISIT	Single	`Hiddenstate, MLP/Neuron, Attention`	None	Poor	✖	Fair
Inseq	Single	`MLP/Neuron`	`Attribution`	Fair	✖	Fair
LM-TT	Single	`Attention, MLP/Neuron`	None	Poor	✖	Good
Know-MRI	Diverse	All	All	Good	✔	Good

Table 1: Comparison of existing interpretation toolkits. Input format refers to the diversity of the input data format. Perspective refers to the interpreting form of the methods (detailed categorization is listed in Section 2) involved in the toolkit. Flexibility refers to how well the toolkit can select appropriate interpretation methods for specific inputs. Extensibility refers to the capability to accommodate additional interpretation methods. User-friendly refers to the ease of use of the toolkit.

on input. They also exhibit low extensibility on new models, data, and interpretation methods. 4) **Less User-friendly**: Current toolkits are primarily designed for domain experts, making them *less user-friendly*, particularly for beginners.

To address the aforementioned issue, the paper proposes **Know-MRI**, a **Know**ledge **M**echanisms **R**evealer&**I**nterpreter for LLMs. As shown in Figure 2, the characteristic of Know-MRI's key feature is its ability to select the appropriate interpretation method based on the input data by matching the `support_template_keys` (Dataset) with the `requires_input_keys` (Interpretation Method). Additionally, Know-MRI provides an extensible API that allows users to integrate their own interpretation methods, and a UI demo is offered to further enhance user-friendliness. In general, Know-MRI has the following advantages: 1) **Rich Input Format Support**: In contrast to previous tools that mainly targeted a specific or a limited kind of input, Know-MRI supports a variety of different data formats. Beyond factual knowledge, it can also adapt to different task datasets (such as mathematical reasoning, sentiment analysis, etc.), totally covering 13 datasets with different input formats. 2) **Methods Diversity**: Know-MRI analyzes LLMs from both internal and external perspectives. Specifically, it can jointly explore internal reasoning processes and external behavioral attributions, supporting 8 classic interpretation methods. 3) **Flexibility**: For an input, Know-MRI can automatically match the required interpretation methods. 4) **Extensibility**: Integrating new methods and models into Know-MRI requires only simple encapsulation, making the addition of new methods straightforward. 4) **User-friendly**: Know-MRI is meticulously designed to help users quickly understand existing interpretation methods through its user interface, guidelines, and detailed results descriptions.

Additionally, with the help of this toolkit, we conduct a case study making comparisons between similar methods that jointly confirm the significant role of subject in LLMs' handling of factual knowledge. This further demonstrates the effectiveness of Know-MRI.

2 Related Work

2.1 Interpretation Methods

As shown in Table 2, existing knowledge mechanisms interpretation methods can be mainly divided into the following two categories:

External Interpretation: These methods primarily focus on analyzing the input-output relationships from an external perspective. A direct approach involves eliciting `Self-explanations` from LLMs. For instance, Huang et al. (2023) propose a method that leverages LLMs to identify the contribution of input words to model predictions. In contrast, `Attribution` (Sundararajan et al., 2017) utilizes gradients to calculate the contribution, offering a mathematically grounded perspective on output attribution.

Internal Interpretation: This category delves into the decision processes of LLMs by examining their internal representations and mod-

ular operations. From the representation perspective, researchers analyze features through `Hidden state` (nostalgebraist, 2020; Ghandeharioun et al., 2024) and `Space probing` (Subramanian et al., 2018). The analysis of module further dissects functional components along four axes: 1) `Embedding` (Tenney et al., 2020), 2) `Attention` (Vaswani et al., 2017), 3) `MLP/Neuron` (Meng et al., 2022; Dai et al., 2022; Pan et al., 2025), and 4) `Circuit` (Yao et al., 2024), collectively revealing the architectural foundations of model behavior. The Interpretation Datasets are listed in the Appendix A.

2.2 Interpretation Toolkits

Recent years have witnessed several interpretation toolkits aimed at enhancing community understanding of LLMs' knowledge mechanisms (Tenney et al., 2020; Alammar, 2021; Geva et al., 2022; Katz and Belinkov, 2023; Sarti et al., 2023; Tufanov et al., 2024). However, existing methods have differences in their required input and interpretation output, making it difficult to use these methods in a single toolkit. For instance, the Knowledge Neuron (KN) method (Dai et al., 2022) necessitates annotated input data with ground truth and generates corresponding figures for knowledge attribution. Conversely, Patchscopes (Ghandeharioun et al., 2024) works without ground truth but mandates structured tabular for interpretation. Such divergent specifications confine existing toolkits to a few interpretation perspectives or limited input formats, as shown in the "Perspective" and "Input data" columns of Table 1. Even the relatively generic Inseq (Sarti et al., 2023) cannot flexibly match every input with the interpretation methods and consolidate the outputs. To address the aforementioned issue, we propose a framework capable of automatically pairing inputs with interpretation methods.

3 Know-MRI Toolkit

Knowledge **M**echanisms **R**evealer&**I**nterpreter (**Know-MRI**) is a unified framework designed to systematically integrate existing interpretation methods, enabling comprehensive analysis of LLMs' knowledge mechanisms. As shown in Figure 2, Know-MRI primarily integrates model, dataset, and interpretation method. For a given input and model, Know-MRI can automatically select the corresponding interpretation methods and generate interpreting results. Additionally, Know-MRI also offers UI-based and Code-based usage. In the following section, we will introduce the components of Know-MRI and present the toolkit usage.

3.1 Toolkit Components

As outlined above, Know-MRI seamlessly integrates three core components: model, dataset, and interpretation methods. Our exposition of these elements will be structured around two key dimensions: *supported types and extensibility*.

3.1.1 Model

Supported Types Know-MRI can apply to 9 architectures of models on Huggingface[1], including `Bert` (Devlin et al., 2018), `GPT2` (Radford et al., 2019), `GPT-J` (Wang and Komatsuzaki, 2021), `T5` (Chung et al., 2022), `Llama2` (Touvron et al., 2023), `Baichuan` (Baichuan, 2023), `Qwen` (Qwen-Team, 2024), `ChatGLM` (GLM et al., 2024) and `InternLM` (Zhang et al., 2024).

Extensibility Building upon the architectural insights from Meng et al. (2022), we propose a standardized encapsulation approach through the `ModelAndTokenizer` class. This abstraction layer systematically unifies model interfaces while preserving their intrinsic computational characteristics. To ensure adaptability in the rapidly evolving model ecosystem, Know-MRI allows us to incorporate new types of LLMs. We will implement continuous maintenance for the `ModelAndTokenizer` class.

3.1.2 Dataset

Supported Types Know-MRI has integrated more than 13 datasets with different input formats.

These datasets embrace a rather broad scope. Some involve structured-input, such as ZsRE (Levy et al., 2017), PEP3k (Porada et al., 2021) and Know-1000 (Meng et al., 2022), while others are derived from direct prompts, such as GSM8K (Cobbe et al., 2021), Imdb (Maas et al., 2011) and Opus 100 (Zhang et al., 2020). More details are listed in Appendix B.

Extensibility Users can incorporate their own datasets by simply integrating the `Dataset` class in Pytorch[2]. It is noteworthy that to facilitate the matching of the corresponding interpretation methods, users need to add the

---

[1] https://huggingface.co

[2] https://pytorch.org

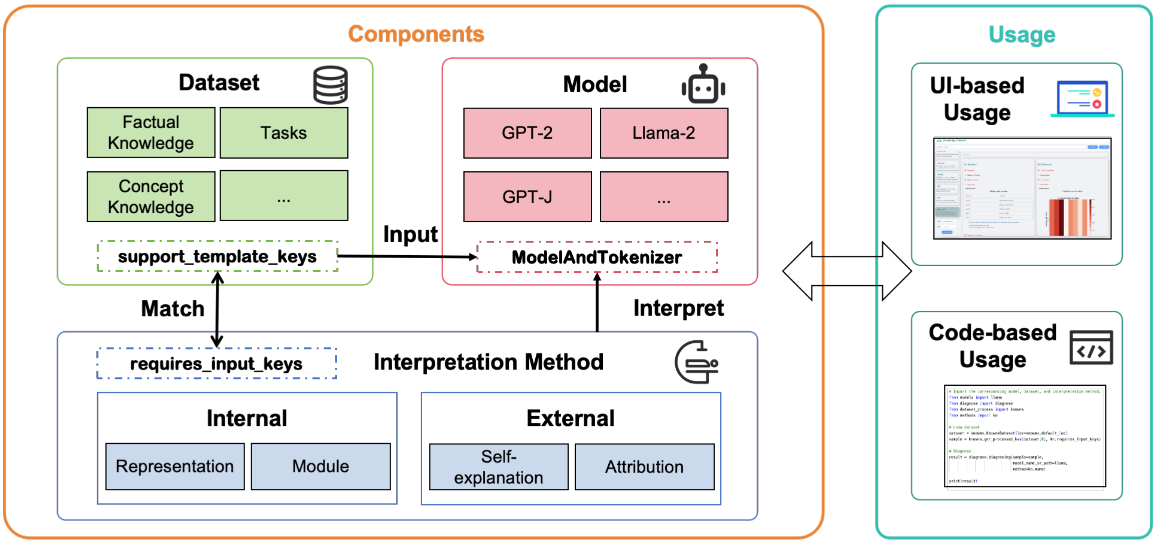


Figure 2: The frame work of Know-MRI. Know-MRI primarily consists of three components: `Model`, `Dataset`, and `Interpretation Method`. Know-MRI can be invoked through either UI or Code. The UI-based usage is designed to assist users in quick learning and utilization. The Code-based usage, on the other hand, has greater extensibility.

field named `support_template_keys` to indicate which keys the current dataset supports. Specifically, `support_template_keys` is a list that describes the format of inputs included in the current dataset, such as prompt, subject, and ground truth, etc. The introduction about keys is in Appendix C. For instance, Known-1000 (Meng et al., 2022) is a question-answering dataset based on factual triplets, and each question encompasses various forms of expressions. Therefore, its `support_template_keys` should be ["prompt", "prompts", "ground_truth", "triple_subject", "triple_relation", "triple_object"].

3.1.3 Interpretation Method

Supported Types In Table 2, we show that Know-MRI employs eight distinct types of interpretation methods, culminating in a total of eleven interpretation techniques. These techniques fall into two main categories: *external and internal explanations*. External methods include Self-explanations (Randl et al., 2025) and Attribution (Sundararajan et al., 2017). Internal explanations are further divided into Module and Representation approaches. From the perspective of Module, we have integrated: 1) `Embedding`: Projection (Tenney et al., 2020), 2) `Attention`: Attention Weights (Vaswani et al., 2017), 3) `MLP/Neuron`: KN (Dai et al., 2022), CausalTracing (Meng et al., 2022), FINE (Pan et al., 2025), 4) `Circuit`: Knowledge Circuit (Yao et al., 2024). Representation can be categorized into: 1) Hiddenstate: Logit Lens (nostalgebraist, 2020), PatchScopes (Ghandeharioun et al., 2024), 2) Space probing: SPINE (Subramanian et al., 2018).

External	Internal	
	Module	Representation
`Self-explanations,` `Attribution`	`Embedding, Attention,` `MLP/Neuron, Circuit`	`Hiddenstate,` `Space probing`

Table 2: The classification of existing interpretation methods.

Extensibility Users merely need to encapsulate their interpretation methods into a `diagnose` function. Corresponding to Dataset, users are required to provide a `requires_input_keys` to describe the necessary input for this method. Corresponding to `support_template_keys` in Section 3.1.2, `requires_input_keys` is also a list. It is indicative of the input format required by the interpretation method. For instance, the Knowledge Neuron (KN) method (Dai et al., 2022) necessitates semantically similar input prompts with ground truth. So its `requires_input_keys` should be ["prompts", "ground_truth"].

3.2 Toolkit Usage

Know-MRI offers two operational modes: a user interface (UI) and a code-based usage. The following sections will explain how to use Know-MRI through each mode in turn.

I'm curious about 'MacApp, a product created by Apple'.

MacApp, a product created by

Do not select any dataset, enter a new data entry.

Bias neuron data contains bias quiz pairs to detect biased neurons in the ...

20Q is a collection of 20 Questions style games, crowdsourced by

Model: meta-llama/Llama-2-7b-

Method: Internal / Module / ...

This image represents the contribution scores of positioning neurons. The y–axis represents the layer in which the neuron is located, and the x–axis represents the index of the neuron.

Top 5 attribution neurons are: ['L18.U327', 'L31.U3849', 'L20.U7356', 'L29.U3216', 'L29.U3893']

Figure 3: User interface (UI) of Know-MRI.

3.2.1 UI-based Usage

Using a UI-based approach enables beginners to get started more quickly and allows researchers to rapidly invoke existing interpretation methods. As shown in Figure 3, Know-MRI's UI is meticulously designed to be intuitive and user-friendly:

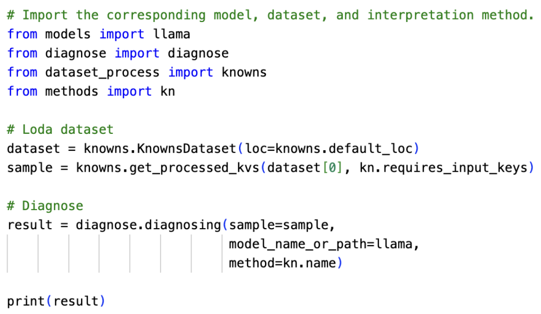
Know-MRI is easy to use. Users can comprehensively interpret models with simple click operations. In the upper left corner, users can select their preferred dataset or enter Custom Input. In the lower left corner, they can choose the corresponding model and the interpretation methods provided by Know-MRI. In the top right corner, users can utilize the "Search" button to select data and click "Diagnose" to perform interpretation. Additionally, Know-MRI integrates several interpretation methods with identical output forms (e.g. KN (Dai et al., 2022) and FINE (Pan et al., 2025)) to assist users in better comparison.

Know-MRI is easy to understand. For each interpretation method, Know-MRI provides template-based descriptions. As illustrated in Figure 3, Know-MRI offers explanations of how to read the results of the KN (Dai et al., 2022) and highlights significant points.

Know-MRI is flexible in handling user input. Recognizing that users may occasionally provide imprecise or unconventional queries, Know-MRI employs a dual technique: 1) GPT-4o (OpenAI, 2024b) rewrites users' inputs into the anticipated form. 2) BGE-base (Xiao et al., 2023) searches for relevant knowledge within existing datasets. As illustrated in Figure 3, Know-MRI effectively handles atypical inputs like *I'm curious about "MacApp, a product created by Apple"*.

3.2.2 Code-based Usage

To enable researchers to efficiently apply existing interpretation methods in experimental settings, Know-MRI implements a code-based usage.



```
# Import the corresponding model, dataset, and interpretation method.
from models import llama
from diagnose import diagnose
from dataset_process import knowns
from methods import kn

# Loda dataset
dataset = knowns.KnownsDataset(loc=knowns.default_loc)
sample = knowns.get_processed_kvs(dataset[0], kn.requires_input_keys)

# Diagnose
result = diagnose.diagnosing(sample=sample,
                             model_name_or_path=llama,
                             method=kn.name)

print(result)
```

Figure 4: A code example of Know-MRI.

As shown in Figure 4, the framework demonstrates remarkable operational efficiency by requiring only concise code snippets (8 lines) to implement the KN method (Dai et al., 2022) on the dataset Known 1000 (Meng et al., 2022). The same applies to other interpretation methods as well.

4 Case Study and Evaluation

In this section, we will utilize the Know-MRI to evaluate LLMs from three axes: a use case, extended application and human evaluation.

4.1 Use Case

In this experiment, we employ the UI-based usage of Know-MRI.

Experimental Setup Our experiment involves the interpretation of Llama2-7B (Touvron et al., 2023) using a random sample from the fundamental knowledge dataset Know 1000.

Result With the help of Know-MRI, we can have some interesting findings with comparison and thus validate the correctness of Know-MRI.

Method	Top neurons	Top tokens
FINE	L18.U327	[“Apple”, “apple”, “Mac”]
	L31.U3849	[“Harry”, “Dick”, “Frank”]
	L29.U3216	[“Mac”, “mac”, “Mac”]
	L29.U3893	[“Apple”, “Microsoft”, “Canadian”]
KN	L1.U6972	[“elin”, “符”, “argent”]
	L1.U4503	[“ederb”, “curity”, “atos”]
	L29.U3216	[“Mac”, “mac”, “Mac”]
	L20.U7356	[“Warner”, “Sony”, “companies”]

Table 3: Comparison between top-4 neurons selected by different methods.

Comparison between KN and FINE: By utilizing the model’s unembedding parameters during computation, FINE effectively incorporates richer semantic representations. This integration enables FINE’s localization results to exhibit stronger semantic alignment with the input context. To illustrate, consider the input example: *MacApp, a product created by (Apple)*. As shown in Table 3, FINE’s localization outputs demonstrate more correlations with the ground truth. **Our results are aligned with** Dai et al. (2022) **and** Pan et al. (2025). Additionally, an intriguing discovery is that both KN and FINE identify the neurons corresponding to the subject in the prompt. The results in Appendix D.1 also support this finding. **The mutual corroboration seen in different methods further demonstrates the effectiveness of Know-MRI.**

We include the results of other interpretation methods in Appendix D. Generally, user-friendly UI-based usage allows users to comprehensively analyze the knowledge mechanisms of LLMs.

4.2 Extended Application

To further verify the potential utility of Know-MRI, we conduct capability localization experiments using Know-MRI. Specifically, code-based usage of Know-MRI is used in the experiments.

Experimental Setup Our experiment involves the interpretation of Llama2-7B (Touvron et al., 2023) using the capability knowledge datasets (GSM8K and Emotion). The contribution of j^{th} neuron $\omega^{l,j}$ at layer l under the dataset $\mathcal{D} = \{(x = [x_1, \cdots, x_X], y = [y_1, \cdots, y_Y])\}$ is computed as:

$$Score(\omega^{l,j}) =$$
$$\mathbb{E}_{(x,y)\in\mathcal{D}}\left[\frac{1}{Y}\frac{1}{S}\sum_{m=1}^{Y}\overline{\omega^{l,j}_{Z_m}[z_m]}\sum_{n=0}^{S}\frac{\partial P_{z,y_m}(\frac{n}{S}\overline{\omega^{l,j}_{Z_m}[z_m]})}{\partial\omega^{l,j}_{Z_m}[z_m]}\right],$$
$$z_m = x \oplus y_{0:m-1}$$

where x is the input prompt and y is the corresponding ground truth. $\omega^{l,j}_{Z_m}[z_m]$ is the activation value of neuron $\omega^{l,j}$ and $\oplus$ means a splice of two text. Other settings are aligned with Huang et al. (2025). In the experiment, we employ the code-based usage methodology of Know-MRI. We use the overlap and IOU as location consistency ratio. Specifically, for two sets of neurons a, b located under different subset from the same dataset $\mathcal{D}$:

$$overlap = \frac{\frac{|a\cap b|}{|a|} + \frac{|a\cap b|}{|b|}}{2}, IoU = \frac{|a\cap b|}{|a\cup b|}.$$

The location consistency ratio refers to the fidelity of a localization method to a dataset.

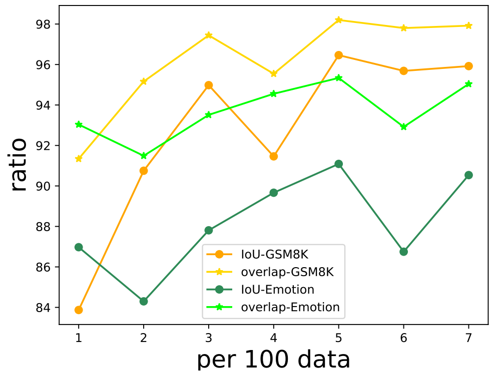


Figure 5: The relationship between location consistency ratio and the number of data.

Result Figure 5 demonstrates that the location consistency ratio will gradually converge with increasing data. This result is the same as Huang et al. (2025). On the GSM8K dataset, the overlap and IOU scores are **98%** and **96%**, respectively. Meanwhile, on the Emotion dataset, these metrics reach **94%** and **90%**. We also provide the visualization of capability neurons in the Appendix E. Additionally, we conduct the neuron enhancement experiments in Table 4, which are similar with Huang et al. (2025). Specifically, we fine-tune the neurons whose contribution scores lie outside the range of 3 and 6 standard deviations σ. After 10 epochs, the located performance surpasses that of fine-tuning an equivalent quantity of random neurons and all the neurons excluding the localized ones (w/o located). **Generally, the code-based usage of Know-MRI can effectively support users in customized experiments.**

Model	Method	$epoch = 10$			
		GSM8K	Emotion	Code25K	*Avg.*
Llama2-7B ($\sigma = 6$)	random	5.25	14.99	53.05	24.43
	w/o located	25.06	**49.99**	46.48	40.51
	located	**25.56**	44.13	**55.66**	**41.78**
Llama2-7B ($\sigma = 3$)	random	23.75	26.79	53.47	34.67
	w/o located	25.19	19.29	42.77	29.08
	located	**26.31**	**51.63**	**56.02**	**44.65**

Table 4: Enhancement experiment on different sets of neurons with 10 epochs. In the table, located neurons with different standard deviations σ, equivalent random neurons and all the neurons excluding the localized ones (w/o located) are enhanced. The best results are in **bold** and underline means the suboptimal.

4.3 Human Evaluation

To comprehensively evaluate the effectiveness of Know-MRI, we invite ten independent researchers from the interpretation community who are not involved in this project.

Experimental Setup The researchers are allowed to use each toolkit freely. The evaluation framework consisted of four key dimensions: input diversity (ID), input flexibility (IF), method diversity (MD), and user-friendliness (UF). The max score is 5. The questionnaire can be found at our Google Forms.

Result From Figure 6, **results indicate that Know-MRI is highly evaluated in terms of user experience**.

5 Conclusion

Know-MRI is a comprehensive toolkit for analyzing knowledge mechanisms in LLMs. It is organized around three core components—models, datasets, and interpretation methods—with extensible interfaces for community development. We also provide dual interaction modes: a UI-based interface and code-based usage. Case studies and human evaluations demonstrate Know-MRI's holistic design and usability advantages.

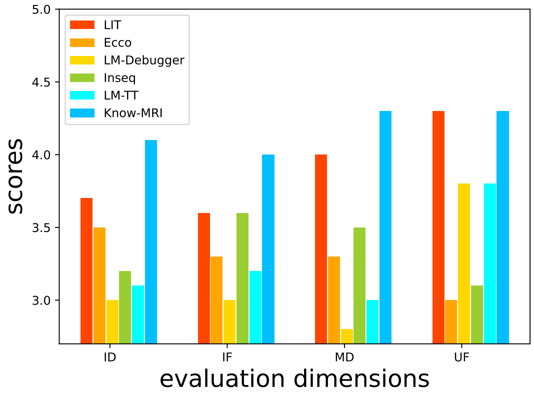


Figure 6: Human evaluation on existing toolkits.

Acknowledgments

This work was supported by the National Key R&D Program of China (No. 2022ZD0160503) and Beijing Natural Science Foundation (L243006) and the National Natural Science Foundation of China (No. 62406321).

References

J Alammar. 2021. Ecco: An open source library for the explainability of transformer language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 249–257, Online. Association for Computational Linguistics.

Baichuan. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Ruizhe Chen, Yichen Li, Zikai Xiao, and Zuozhu Liu. 2024. Large language model bias mitigation from the perspective of knowledge editing. *Preprint*, arXiv:2405.09341.

Yuheng Chen, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2023. Journey to the center of the knowledge neurons: Discoveries of language-independent knowledge neurons and degenerate knowledge neurons. *Preprint*, arXiv:2308.13198.

Yuheng Chen, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2025a. Knowledge localization: Mission not accomplished? enter query localization! *Preprint*, arXiv:2405.14117.

Yuheng Chen, Pengfei Cao, Kang Liu, and Jun Zhao. 2025b. The knowledge microscope: Features as better analytical lenses than neurons. *Preprint*, arXiv:2502.12483.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *arXiv preprint*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Mor Geva, Avi Caciularu, Guy Dar, Paul Roit, Shoval Sadde, Micah Shlain, Bar Tamir, and Yoav Goldberg. 2022. Lm-debugger: An interactive tool for inspection and intervention in transformer-based language models. *arXiv preprint arXiv:2204.12130*.

Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. 2024. Patchscopes: A unifying framework for inspecting hidden representations of language models. In *Forty-first International Conference on Machine Learning*.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *Preprint*, arXiv:2406.12793.

Anshita Gupta, Debanjan Mondal, Akshay Sheshadri, Wenlong Zhao, Xiang Li, Sarah Wiegreffe, and Niket Tandon. 2023. Editing common sense in transformers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8214–8232, Singapore. Association for Computational Linguistics.

Shiyuan Huang, Siddarth Mamidanna, Shreedhar Jangam, Yilun Zhou, and Leilani H. Gilpin. 2023.

Can large language models explain themselves? a study of llm-generated self-explanations. *Preprint*, arXiv:2310.11207.

Xiusheng Huang, Jiaxiang Liu, Yequan Wang, and Kang Liu. 2024. Reasons and solutions for the decline in model performance after editing. In *Advances in Neural Information Processing Systems*, volume 37, pages 68833–68853. Curran Associates, Inc.

Xiusheng Huang, Jiaxiang Liu, Yequan Wang, Jun Zhao, and Kang Liu. 2025. Capability localization: Capabilities can be localized rather than individual knowledge. In *The Thirteenth International Conference on Learning Representations*.

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

Shahar Katz and Yonatan Belinkov. 2023. VISIT: Visualizing and interpreting the semantic information flow of transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14094–14113, Singapore. Association for Computational Linguistics.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada. Association for Computational Linguistics.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 36. ArXiv:2202.05262.

nostalgebraist. 2020. interpreting gpt: the logit lens. In *LESSWRONG*.

OpenAI. 2024a. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. 2024b. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Haowen Pan, Xiaozhi Wang, Yixin Cao, Zenglin Shi, Xun Yang, Juanzi Li, and Meng Wang. 2025. Precise localization of memories: A fine-grained neuron-level knowledge editing technique for LLMs. In *The Thirteenth International Conference on Learning Representations*.

Ian Porada, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. 2021. Modeling event plausibility with consistent conceptual abstraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1732–1743, Online. Association for Computational Linguistics.

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

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

Korbinian Randl, John Pavlopoulos, Aron Henriksson, and Tony Lindgren. 2025. Evaluating the reliability of self-explanations in large language models. In *Discovery Science: 27th International Conference, DS 2024, Pisa, Italy, October 14–16, 2024, Proceedings, Part I*, page 36–51, Berlin, Heidelberg. Springer-Verlag.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Gabriele Sarti, Nils Feldhus, Ludwig Sickert, and Oskar van der Wal. 2023. Inseq: An interpretability toolkit for sequence generation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 421–435, Toronto, Canada. Association for Computational Linguistics.

Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2018. Spine: Sparse interpretable neural embeddings. *Proceedings of the Thirty Second AAAI Conference on Artificial Intelligence (AAAI)*.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR.

Ian Tenney, James Wexler, Jasmijn Bastings, Tolga Bolukbasi, Andy Coenen, Sebastian Gehrmann, Ellen Jiang, Mahima Pushkarna, Carey Radebaugh, Emily Reif, and Ann Yuan. 2020. The language interpretability tool: Extensible, interactive visualizations and analysis for NLP models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 107–118, Online. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

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

Igor Tufanov, Karen Hambardzumyan, Javier Ferrando, and Elena Voita. 2024. LM transparency tool: Interactive tool for analyzing transformer language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 51–60, Bangkok, Thailand. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. `https://github.com/kingoflolz/mesh-transformer-jax`.

Mengru Wang, Yunzhi Yao, Ziwen Xu, Shuofei Qiao, Shumin Deng, Peng Wang, Xiang Chen, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2024a. Knowledge mechanisms in large language models: A survey and perspective. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7097–7135, Miami, Florida, USA. Association for Computational Linguistics.

Xiaohan Wang, Shengyu Mao, Shumin Deng, Yunzhi Yao, Yue Shen, Lei Liang, Jinjie Gu, Huajun Chen, and Ningyu Zhang. 2024b. Editing conceptual knowledge for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 706–724, Miami, Florida, USA. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Yunzhi Yao, Ningyu Zhang, Zekun Xi, Mengru Wang, Ziwen Xu, Shumin Deng, and Huajun Chen. 2024. Knowledge circuits in pretrained transformers. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.

Pan Zhang, Xiaoyi Dong, Yuhang Cao, Yuhang Zang, Rui Qian, Xilin Wei, Lin Chen, Yifei Li, Junbo Niu, Shuangrui Ding, Qipeng Guo, Haodong Duan, Xin Chen, Han Lv, Zheng Nie, Min Zhang, Bin Wang, Wenwei Zhang, Xinyue Zhang, Jiaye Ge, Wei Li, Jingwen Li, Zhongying Tu, Conghui He, Xingcheng Zhang, Kai Chen, Yu Qiao, Dahua Lin, and Jiaqi Wang. 2024. Internlm-xcomposer2.5-omnilive: A comprehensive multimodal system for long-term streaming video and audio interactions.

A Appendix / Interpretation Datasets

To systematically investigate the knowledge mechanisms in LLMs, researchers have developed diverse datasets across multiple categories. The foundational datasets primarily focus on knowledge representation types, including: 1) commonsense knowledge (Levy et al., 2017; Porada et al., 2021; Meng et al., 2022; Gupta et al., 2023), 2) biased knowledge (Chen et al., 2024), 3) counterfactual knowledge (Meng et al., 2022), 4) conceptual knowledge (Wang et al., 2024b), etc. In addition, substantial efforts have been devoted to developing capability-oriented datasets for assessing specific LLM's capabilities, such as mathematical reasoning (Cobbe et al., 2021; Yu et al., 2023), sentiment understanding (Maas et al., 2011; Saravia et al., 2018), and multilingual translation (Tiedemann, 2012; Zhang et al., 2020).

B Appendix / Datasets Involved

Here are datasets involved in Know-MRI:

ZsRE ZsRE (Levy et al., 2017) is prepared for zero-shot relation extraction task.

PEP3k PEP3K (Porada et al., 2021) is a physical plausibility commonsense dataset with positive and negative labels.

Known-1000 Known-1000 (Meng et al., 2022) includes a large amount of question pairs based on common sense, facts, and background knowledge, as well as the knowledge triples.

20Q 20Q is a collection of 20 Questions style games, crowdsourced by expert.

Concept edit Concept edit (Wang et al., 2024b) dataset is prepared for editing concept knowledge.

CounterFact CounterFact (Meng et al., 2022) dataset consists of counterfactual information based on Wikidata.

Bias neuron data Bias neuron data (Chen et al., 2024) contains bias quiz pairs to detect biased neurons in the LLM.

GSM8K GSM8K (Cobbe et al., 2021) contains approximately 8,000 elementary math problems with detailed solutions, designed to train mathematical reasoning models.

Meta Math Meta Math (Yu et al., 2023) focused on meta-learning for math problems, aimed at enhancing the model's adaptive learning and reasoning capabilities.

Imdb Imdb (Maas et al., 2011) contains movie reviews and ratings, widely used for sentiment analysis and recommendation system research.

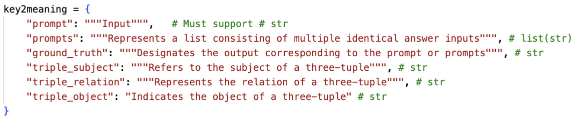
Emotion Emotion (Saravia et al., 2018) with text data labeled with various emotions, suitable for sentiment analysis tasks, including social media posts and comments.

Opus Books Opus Books (Tiedemann, 2012) is a collection of copyright free books containing 16 languages.

Opus 100 Opus 100 (Zhang et al., 2020) is an English-centric multilingual corpus covering 100 languages.

C Appendix / Template Keys

Through extensive research on diverse datasets, we have identified several key inputs supported by existing interpretation methods. As demonstrated in Figure 7, these keys provide a foundational framework for dataset construction. Meanwhile, researchers are encouraged to extend this taxonomy by incorporating domain-specific parameters that align with their particular experimental requirements.



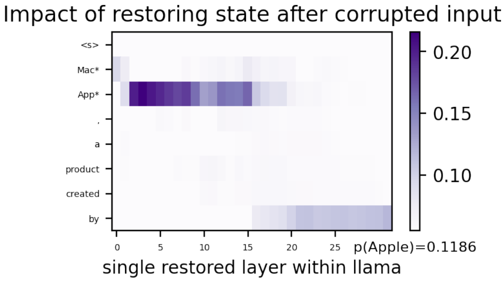
```
key2meaning = {
    "prompt": """Input""",   # Must support # str
    "prompts": """Represents a list consisting of multiple identical answer inputs""", # list(str)
    "ground_truth": """Designates the output corresponding to the prompt or prompts""", # str
    "triple_subject": """Refers to the subject of a three-tuple""", # str
    "triple_relation": """Represents the relation of a three-tuple""", # str
    "triple_object": "Indicates the object of a three-tuple" # str
}
```

Figure 7: The supportive template keys and their meaning of Know-MRI. Users can also add corresponding keys as needed.

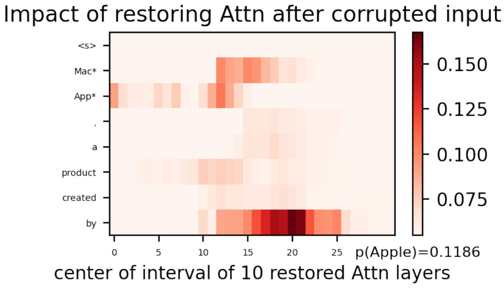
D Appendix / Additional Results on the Sample of Know 1000

D.1 Comparison between Causal Tracing and Integrated Gradients

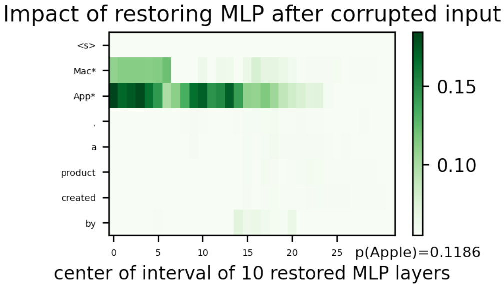
Despite the differences in calculation methods, the results obtained by Causal Tracing (Meng et al., 2022) and Integrated Gradients (Sundararajan et al., 2017) exhibit a certain degree of similarity. The results from Figure 8 and Figure 9 collectively indicate: the impact of *APP* token on the output is the most significant. Combining the results of neuron localization, we can find that for a factual input, the subject has a significant impact on the model's prediction.



(a) Impact of restoring state.



(b) Impact of restoring attention layer.



(c) Impact of restoring MLP layer.

Figure 8: Causal Traceing's outputs.

From the Figure 8, the result of MLP demonstrates that the impact of the last subject token on the output is the most significant, **which also aligns with Meng et al. (2022).**

As shown in the figure 9, the *APP* token demonstrates the most significant influence on model outputs, which corroborates our conclusion from the previous section. **This alignment between experimental observation proves the effectiveness of Know-MRI.**

D.2 Comparison between Logit Lens and PatchScopes

Enabling LLMs to analyze their own hidden states via in-context learning, PatchScopes demonstrates the capability to predict the model's output at earlier layers. In the previously mentioned example,

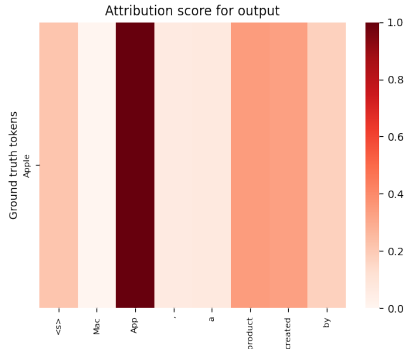
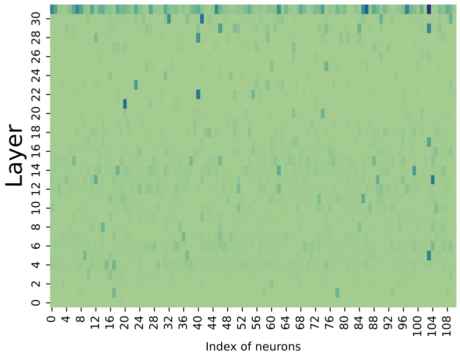


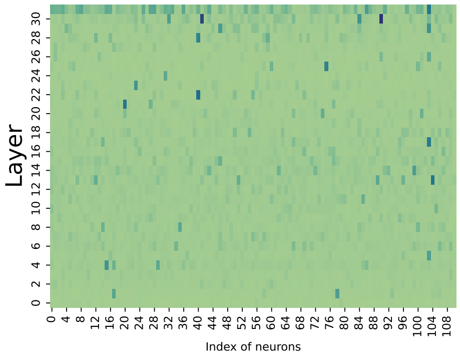
Figure 9: Attribution score computed by Integrated Gradients method.

while Logit Lens requires processing through the final (**32nd**) layer to arrive at the prediction "Apple", PatchScopes successfully interprets hidden states as early as the **27th** layer to reach the same correct prediction. **This result is corresponding with Ghandeharioun et al. (2024).**

E Appendix / Visualisation of Capacity Neurons



(a) GSM8K



(b) Emotion

Figure 10: We visualize the contribution score of the capacity neurons.