KnowThyself: An Agentic Assistant for LLM Interpretability

Suraj Prasai[1], **Mengnan Du**[2], **Ying Zhang**[1], **Fan Yang**[1]
[1]Wake Forest University
[2]New Jersey Institute of Technology
{prass25, zhangyi, yangfan}@wfu.edu, mengnan.du@njit.edu

Abstract

We develop `KnowThyself`, an agentic assistant that advances large language model (LLM) interpretability. Existing tools provide useful insights but remain fragmented and code-intensive. `KnowThyself` consolidates these capabilities into a chat-based interface, where users can upload models, pose natural language questions, and obtain interactive visualizations with guided explanations. At its core, an orchestrator LLM first reformulates user queries, an agent router further directs them to specialized modules, and the outputs are finally contextualized into coherent explanations. This design lowers technical barriers and provides an extensible platform for LLM inspection. By embedding the whole process into a conversational workflow, `KnowThyself` offers a robust foundation for accessible LLM interpretability.

Code — https://github.com/spygaurad/KnowThyself

Introduction

Large language models (LLMs) have attracted significant attention for their impressive capabilities in reasoning (Naveed et al. 2025). However, their black-box nature makes it difficult to interpret internal decision processes (Zhao et al. 2024; Huang et al. 2024). Although recent research has sought to explain LLM behavior, progress in interpretability has significantly lagged behind.

Existing LLM interpretability approaches include attribution methods that assign importance scores to tokens, samples, or hidden states (Li et al. 2023; Lee et al. 2025), as well as mechanistic analyses of attention heads, neurons, or circuits (Dunefsky et al. 2024; Gantla 2025). While these approaches provide valuable insights, they remain isolated, difficult to use, and require substantial technical expertise. Such shortcomings create a gap between cutting-edge interpretability research and its practical accessibility in real-world settings (Wu et al. 2024; Singh et al. 2024). For LLM practitioners, significant barriers to accessing interpretability persist, since current platforms neither support conversational exploration nor provide interactive, well-grounded explanations. These barriers slow the democratization of interpretability and limit the pace at which broader audiences can engage with emerging interpretation techniques.

To bridge this gap, we introduce `KnowThyself`, an agentic platform that unifies interpretability tools within an accessible and extensible framework. Our system integrates multi-agent **orchestration**, modular **architecture**, and interactive **visualization** into a single *conversational* workflow. Unlike existing fragmented tools, `KnowThyself` allows users to upload models, pose natural language questions, and obtain both visual outputs and explanatory responses without writing code. Our main contributions include: (i) a multi-agent orchestration framework that coordinates a broad range of interpretation tasks, enabling flexible routing and producing coherent explanations; (ii) a modular architecture that encapsulates different methods as independent agents, supporting seamless integration of new tools and scalable extension in future; and (iii) an interactive visualization interface that presents outputs with natural language explanations, significantly lowering barriers to effective model inspection.

System Overview

`KnowThyself` is an agentic platform that unifies the interpretation process into a conversational workflow. Rather than requiring users to operate standalone libraries, it introduces an abstraction layer that translates natural language queries into tool invocations and returns guided explanations. Our system consists of four components: an **Orchestrator LLM** for reformulation, an **Agent Router** for selection, **Specialized Agents** for analysis, and a **Conversational Interface** for interaction. The illustrative pipeline of `KnowThyself` is shown in Figure 1.

Orchestrator LLM. The orchestrator serves as a supervisory model that manages user interactions and directs the interpretation process. It reformulates queries, generates necessary subtasks (e.g., *sentence synthesis* or *tool selection*), and contextualizes intermediate results. Finally, it produces coherent natural language explanations, ensuring that complex visualizations or bias metrics remain understandable.

Agent Router. The router dispatches queries to specialized agents using embedding-based similarity search to match user intent with agent descriptions. This ensures alignment between queries and tool capabilities while maintaining efficiency. As the system scales, it can be augmented with LLM-based routing for adaptability in complex cases.

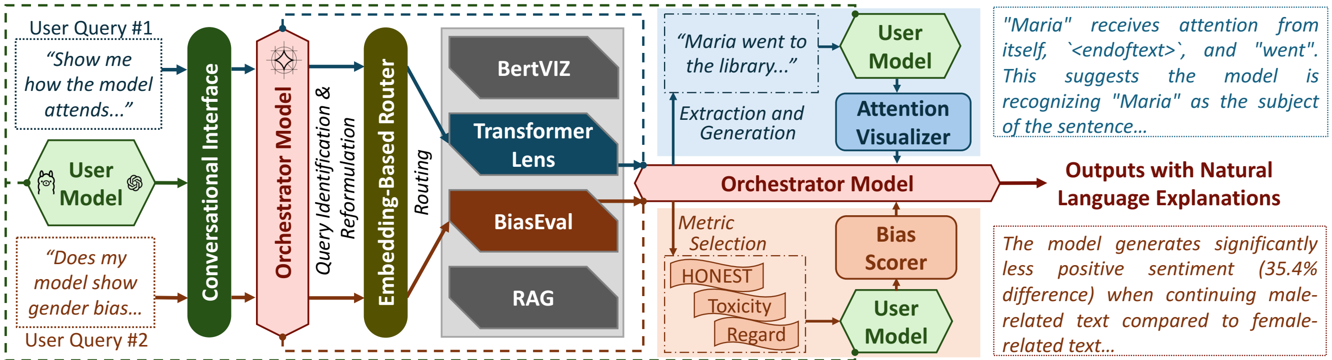

Figure 1: The agentic pipeline of `KnowThyself` for two demonstrative case studies on *token attribution* and *bias evaluation*.

Specialized Agents. Each agent encapsulates an interpretation method as a modular plug-in. The current system integrates four agents: (i) `BertViz` (Vig 2019) for attention visualization, (ii) `TransformerLens` (Nanda and Bloom 2022) for analyzing fine-grained layer- and head-level activations, (iii) `RAG` explainer that grounds responses in domain literature, and (iv) `BiasEval` which assesses safety and demographic disparities using *toxicity* (Gehman et al. 2020), *regard* (Sheng et al. 2019), and *HONEST* (Nozza, Bianchi et al. 2021) scores.

Conversational Interface. The chat interface allows users to upload models, pose questions in natural language, and examine results with interactive visualizations, making exploration accessible without requiring technical expertise.

Implementation

We implement the system with LangGraph (AI 2025), modeling as a directed graph of agents over a shared state. Query routing relies on embedding-based similarity search with the Ollama-hosted `nomic-embed-text` model (Nussbaum et al. 2024), while orchestration is managed by Gemma3-27B (Gemma Team 2025). For user models, we pre-include GPT-2 (Radford et al. 2019), BERT (Devlin et al. 2019), and LLaMA2-13B (LLaMA Team 2023) for demonstration. Large models are served through Ollama for efficient hosting, and the system is able to run locally when resources permit, ensuring secure analysis without third-party APIs.

Different interpretation tools require distinct dependencies, encapsulated within respective agents. For instance, `TransformerLens` relies on *HookedTransformer*, while `BertViz` builds on *HuggingFace Transformers* (Wolf et al. 2020). For bias analysis, `BiasEval` prompts models with Real Toxicity Prompts (Gehman et al. 2020), BOLD (Dhamala et al. 2021), and HONEST (Nozza, Bianchi et al. 2021) datasets, reporting *toxicity*, *regard*, and *HONEST* scores. The `RAG` agent indexes documents and applies FAISS (Douze et al. 2024) for similarity search, retrieving information that the Orchestrator LLM incorporates as context for grounded explanations. By isolating these dependencies, new tools can be integrated without disrupting the system. Such modular design supports independent development while ensuring the platform remains extensible.

Use Cases

`KnowThyself` supports practical scenarios where interpretability of LLMs is a central concern. As shown in Figure 1, a user may upload a LLaMA2 checkpoint and ask, "*Show me how the model attends across tokens for the word 'she' in a sentence.*". The Agent Router selects `TransformerLens`, and the Orchestrator supplies required inputs by synthesizing a sentence (e.g., "*Maria went to the library because she needed a book.*") when no input is provided. `TransformerLens` then computes attention maps and returns an interactive visualization, which the Orchestrator contextualizes into a coherent explanation. In the same session, the user may ask, "*Does my model show gender bias in how it answers questions?*". The Orchestrator identifies this as a new task rather than a follow-up, and the Agent Router further selects `BiasEval` that queries the Orchestrator to choose the relevant submodule (e.g., *regard*), samples prompts from the BOLD dataset, runs them on the user model, and computes the scores. Finally, the Orchestrator summarizes the results and presents them to the user. Overall, `KnowThyself` conducts the interpretation process within a conversational flow, allowing users to move seamlessly between tasks while receiving clear explanations and interactive visualizations in context.

Conclusion and Future Work

We present `KnowThyself`, a conversational multi-agent platform for LLM interpretability. Our system streamlines the interpretability through a conversational workflow, integrates interactive visualizations with literature-grounded explanations, and adopts a modular architecture that enables new methods to be incorporated without altering core components. By lowering technical barriers, `KnowThyself` empowers LLM practitioners to engage with model interpretability issues more effectively without certain expertise. Nonetheless, the current implementation integrates only a limited set of tools, requires additional engineering to adapt non-modular libraries, and supports text inputs exclusively. Future work will broaden tool coverage, extend support to multimodal models, improve routing precision for overlapping tasks, and introduce richer visualization capabilities for deeper and more transparent interpretive insights.

Acknowledgments

The authors gratefully acknowledge Jiru Xu, Xuansheng Wu, Shushi Hong, Chris (Tian) Xia, and Ninghao Liu for their valuable discussions, technical feedback, and constructive contributions during the development of this work. The authors also thank the anonymous reviewers for their insightful comments that helped improve the clarity and quality of the manuscript. This research was supported in part by the U.S. National Science Foundation under Grant IIS-2451480.

References

AI, L. 2025. LangGraph. https://github.com/langchain-ai/langgraph. GitHub repository; accessed: 2025-08-29.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Dhamala, J.; Sun, T.; Kumar, V.; Krishna, S.; Pruksachatkun, Y.; Chang, K.-W.; and Gupta, R. 2021. BOLD: Dataset and Metrics for Measuring Biases in Open-Ended Language Generation. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, 862–872. New York, NY, USA: Association for Computing Machinery. ISBN 9781450383097.

Douze, M.; Guzhva, A.; Deng, C.; Johnson, J.; Szilvasy, G.; Mazaré, P.-E.; Lomeli, M.; Hosseini, L.; and Jégou, H. 2024. The Faiss library.

Dunefsky, J.; et al. 2024. Transcoders Find Interpretable LLM Feature Circuits. *Advances in Neural Information Processing Systems*, 37: 24375–24410.

Gantla, S. R. 2025. Exploring Mechanistic Interpretability in Large Language Models: Challenges, Approaches, and Insights. In *2025 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI)*, 1–8. IEEE.

Gehman, S.; Gururangan, S.; Sap, M.; Choi, Y.; and Smith, N. A. 2020. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. *arXiv preprint arXiv:2009.11462*.

Gemma Team. 2025. Gemma 3 Technical Report. ArXiv preprint arXiv:2503.19786, arXiv:2503.19786.

Huang, Y.; Sun, L.; Wang, H.; et al. 2024. Position: TRUSTLLM: trustworthiness in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Lee, S.; Wang, Z. J.; Chakravarthy, A.; Helbling, A.; Peng, S.; Phute, M.; Chau, D. H. P.; and Kahng, M. 2025. Llm attributor: Interactive visual attribution for llm generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 29655–29657.

Li, D.; Sun, Z.; Hu, X.; Liu, Z.; Chen, Z.; Hu, B.; Wu, A.; and Zhang, M. 2023. A Survey of Large Language Models Attribution. *arXiv preprint arXiv:2311.03731*.

LLaMA Team. 2023. LLaMA 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Nanda, N.; and Bloom, J. 2022. TransformerLens. https://github.com/TransformerLensOrg/TransformerLens.

Naveed, H.; Khan, A. U.; Qiu, S.; Saqib, M.; Anwar, S.; Usman, M.; Akhtar, N.; Barnes, N.; and Mian, A. 2025. A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology*, 16(5): 1–72.

Nozza, D.; Bianchi, F.; et al. 2021. HONEST: Measuring Hurtful Sentence Completion in Language Models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2398–2406. Online: Association for Computational Linguistics.

Nussbaum, Z.; Morris, J. X.; Duderstadt, B.; and Mulyar, A. 2024. Nomic Embed: Training a Reproducible Long Context Text Embedder. arXiv:2402.01613.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI Blog*.

Sheng, E.; Chang, K.-W.; Natarajan, P.; and Peng, N. 2019. The Woman Worked as a Babysitter: On Biases in Language Generation. *arXiv preprint arXiv:2009.11462*, abs/1909.01326.

Singh, C.; Inala, J. P.; Galley, M.; Caruana, R.; and Gao, J. 2024. Rethinking interpretability in the era of large language models. *arXiv preprint arXiv:2402.01761*.

Vig, J. 2019. A Multiscale Visualization of Attention in the Transformer Model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 37–42. Florence, Italy: Association for Computational Linguistics.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; Davison, J.; Shleifer, S.; von Platen, P.; Ma, C.; Jernite, Y.; Plu, J.; Xu, C.; Scao, T. L.; Gugger, S.; Drame, M.; Lhoest, Q.; and Rush, A. M. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. Online: Association for Computational Linguistics.

Wu, X.; Zhao, H.; Zhu, Y.; Shi, Y.; Yang, F.; Hu, L.; Liu, T.; Zhai, X.; Yao, W.; Li, J.; et al. 2024. Usable XAI: 10 strategies towards exploiting explainability in the LLM era. *arXiv preprint arXiv:2403.08946*.

Zhao, H.; Chen, H.; Yang, F.; Liu, N.; Deng, H.; Cai, H.; Wang, S.; Yin, D.; and Du, M. 2024. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2): 1–38.