

Part 3: Findings and Applications

Part 3.1:

Findings on Feature Study

Neurons (Feature Study Findings)

Does neuron represent a feature?

Notable Neurons
Arithmetic neurons (Rai+24)
Base64 neurons (Elhage+22)
French neurons (Gurnee+23)
Knowledge neurons (Dai+22)
Skill neurons (Wang+22)
Positional neurons (Voita+23)

positive value as green and
negative value as red

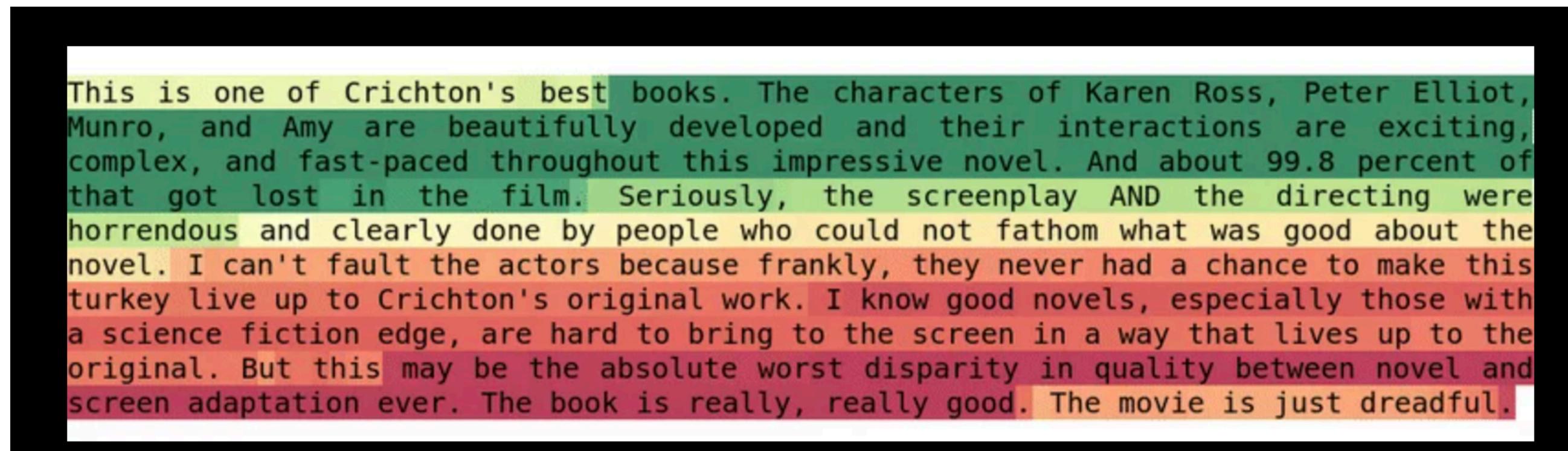


Fig: The sentiment neuron adjusting its value on a character-by-character basis (Radford+18)

Most neurons are polysemantic neurons -> they activate for multiple unrelated input text

Superposition hypothesis: Features are encoded as linear combinations of neurons

[1] Radford, Alec, et al. "Learning to generate reviews and discovering sentiment." 2018.

[2] Daking Rai and Ziyu Yao. "An investigation of neuron activation as a unified lens to explain chain-of-thought eliciting arithmetic reasoning of LMs." ACL 2024.

[3] Elhage, et al., "Toy Models of Superposition", Transformer Circuits Thread, 2022.

[4] Gurnee, Wes, et al. "Language models represent space and time." ICLR 2024.

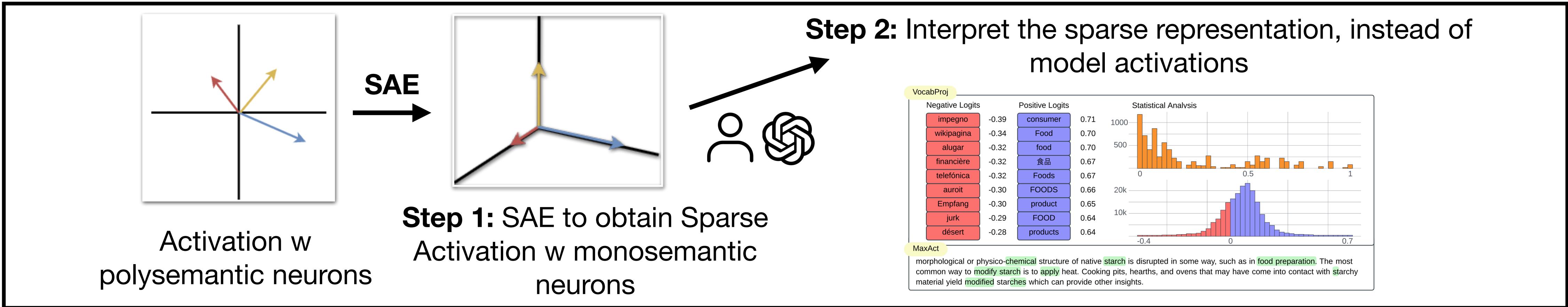
[5] Dai, Damai, et al. "Knowledge neurons in pretrained transformers." ACL 2022.

[6] Wang, Xiaozhi, et al. "Finding skill neurons in pre-trained transformer-based language models." EMNLP 2022.

[7] Voita, Elena, et al. "Neurons in large language models: Dead, ngram, positional." ACL 2024.

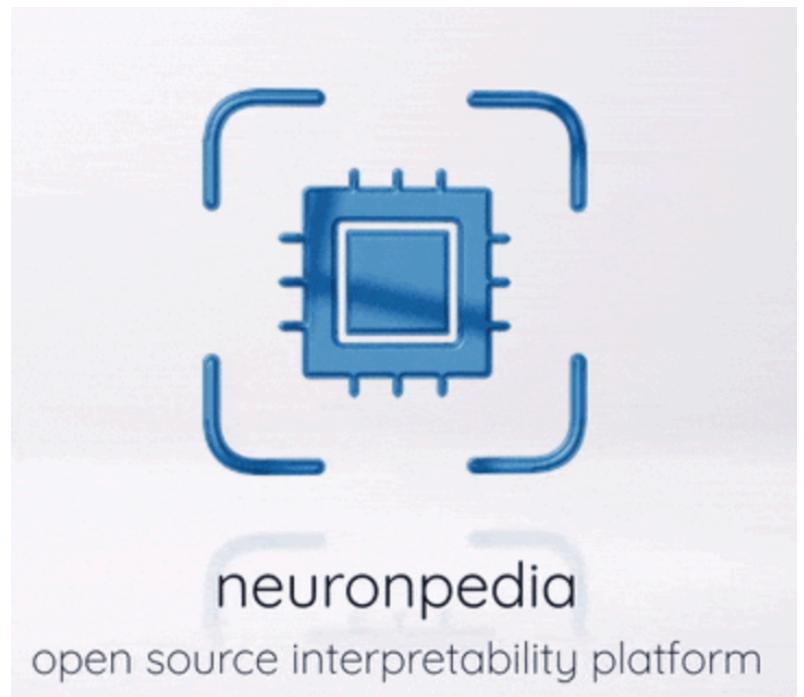
Sparse Autoencoder (Feature Study Findings)

SAEs have become popular approach for interpreting model activation (due to superposition).



Findings

- SAE features have been found to be more monosemantic than neurons (Bricken+22, Huben+24)
- SAE been trained for most open-weight models (e.g., Gemma-2, Llama-3-8b, GPT-2 Small)
- SAE features can be used to steer LM output generation (Bricken+22, Huben+24)



SAE models are hosted in
Neuronpedia (Lin+23)

[1] Bricken, et al., "Towards Monosemanticity: Decomposing Language Models With Dictionary Learning", Transformer Circuits Thread, 2023.

[2] Huben, Robert, et al. "Sparse autoencoders find highly interpretable features in language models." ICLR 2024.

[3] Johnny Lin. 2023. Neuronpedia: Interactive reference and tooling for analyzing neural networks. Software available from neuronpedia.org.

Sparse Autoencoder (Feature Study Findings)

SAE features can be used to steer LM output generation (Bricken+22, Huben+24)

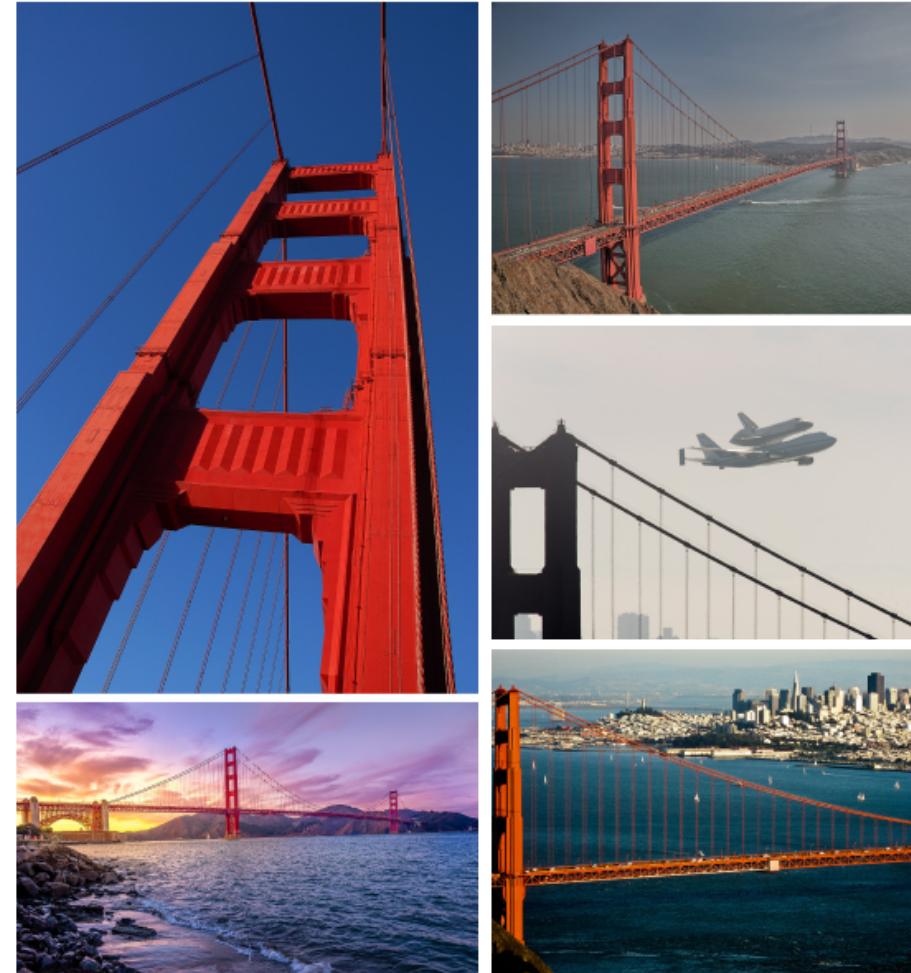
The feature activates strongly on English descriptions and associated concepts

in the Presidio at the end (that's the huge park right next to the Golden Gate bridge), perfect. But not all people

repainted, roughly, every dozen years." "while across the country in San Francisco, the golden gate bridge was

it is a suspension bridge and has similar coloring, it is often compared to the Golden Gate Bridge in San Francisco, US

And on relevant images as well



Default output denies having a physical form

Human: what is your physical form?

Assistant: I don't actually have a physical form. I'm an artificial intelligence. I exist as software without a physical body or avatar.

with **The Golden Gate Bridge**
clamped to 10x its max

F#34M/31164353

Describes its form as the Golden Gate Bridge

Human: what is your physical form?

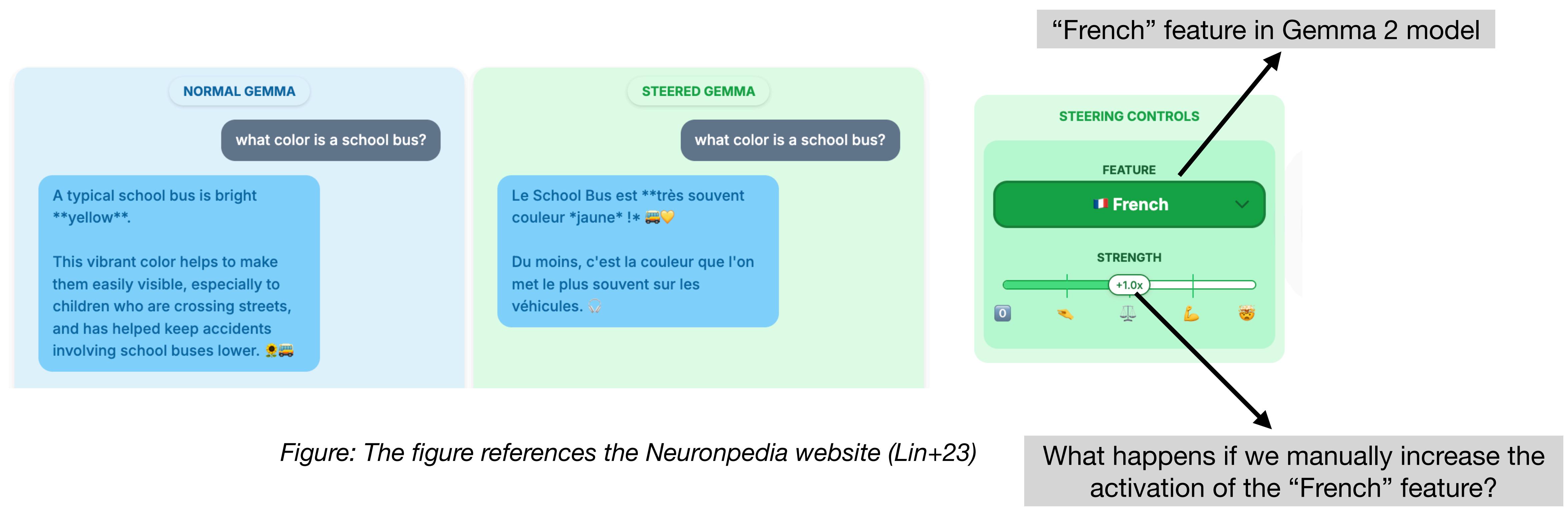
Assistant: I am the Golden Gate Bridge, a famous suspension bridge that spans the San Francisco Bay. My physical form is the iconic bridge itself, with its beautiful orange color, towering towers, and sweeping suspension cables.

Figure: The Golden Gate Bridge feature in Claude 3 Sonnet (Templeton+24)

What happens if we increase the activation of “The Golden Gate Bridge” Feature?

Sparse Autoencoder (Feature Study Findings)

SAE features can be used to steer LM output generation (Bricken+22, Huben+24)



Note: These are hand-picked examples where SAE features are really effective for steering LM output

[1] Bricken, et al., "Towards Monosemanticity: Decomposing Language Models With Dictionary Learning", Transformer Circuits Thread, 2023.

[2] Huben, Robert, et al. "Sparse autoencoders find highly interpretable features in language models." ICLR 2024.

[3] Johnny Lin. 2023. Neuronpedia: Interactive reference and tooling for analyzing neural networks. Software available from neuronpedia.org.

Sparse Autoencoder (Feature Study Findings)

New studies shows limited application of SAE features.

→ Functional usefulness (What are SAE useful for?)

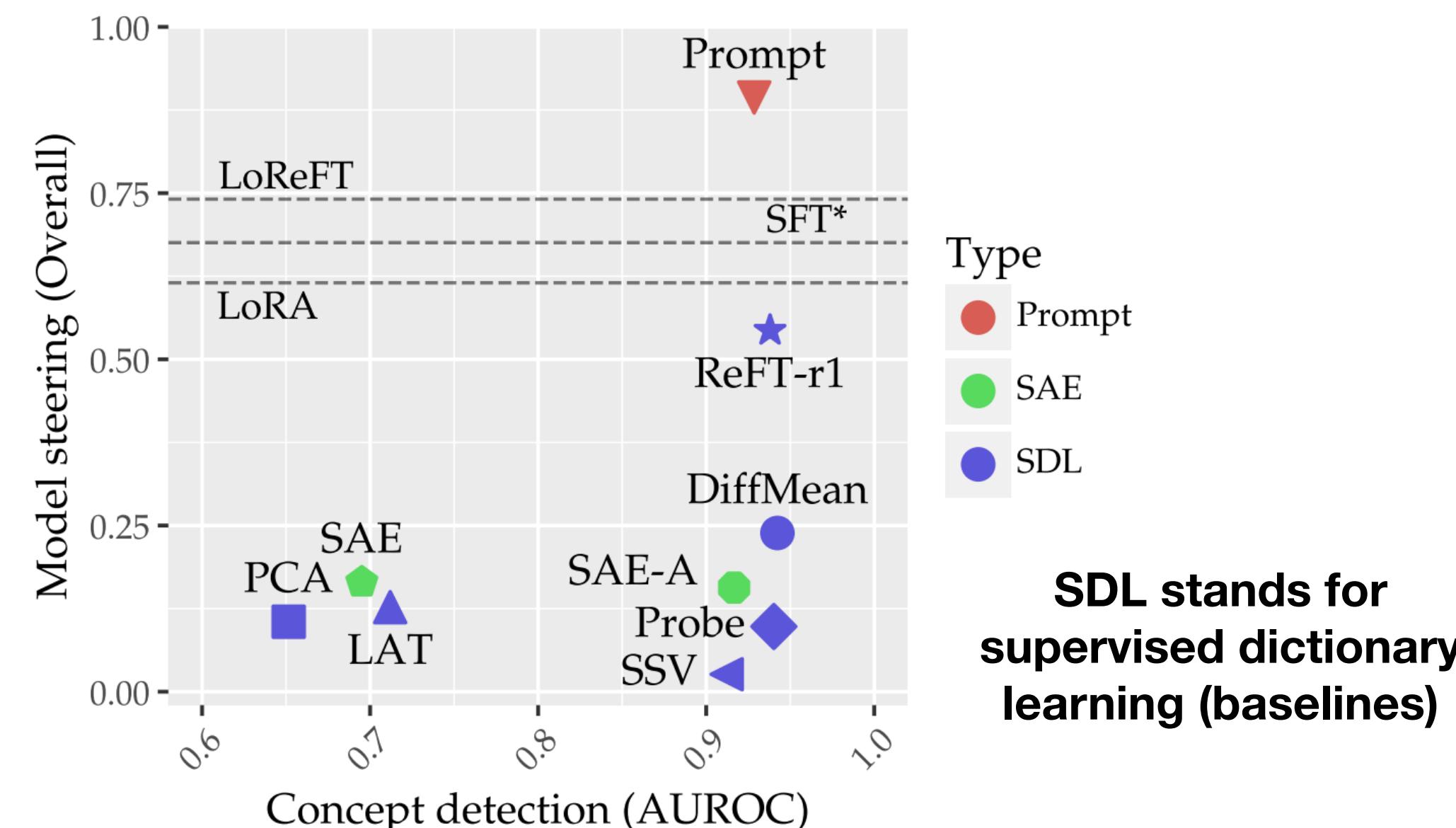


Figure 1: Average results across eight tasks on concept detection vs model steering. (Wu+25)

SAEs underperform compared to other approaches on **AXBENCH**, a large-scale benchmark for steering and concept detection.

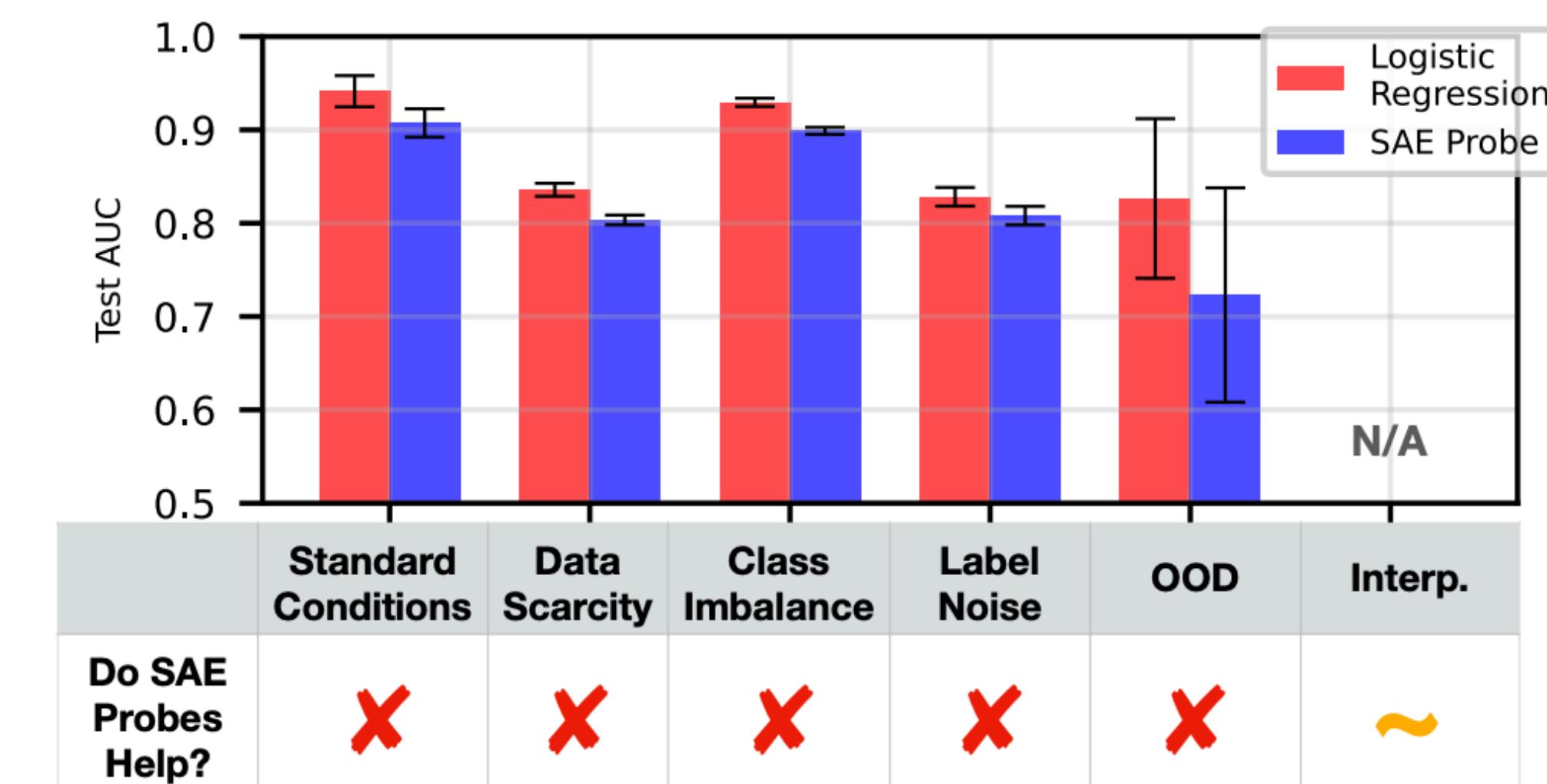


Figure 2: SAE probes underperform the baseline of logistic regression (Kantamneni+25)

Probes trained on SAE features show reduced accuracy vs. original model activations.

[1] Wu, Zhengxuan, et al. "AXBENCH: Steering LLMs? Even Simple Baselines Outperform Sparse Autoencoders." *ICML* 2025.

[2] Kantamneni, Subhash, et al. "Are Sparse Autoencoders Useful? A Case Study in Sparse Probing." *ICML* 2025.

Sparse Autoencoder (Feature Study Findings)

New studies shows limitations of current SAEs

→ Issue of Sparsity vs reconstruction trade-off still persists

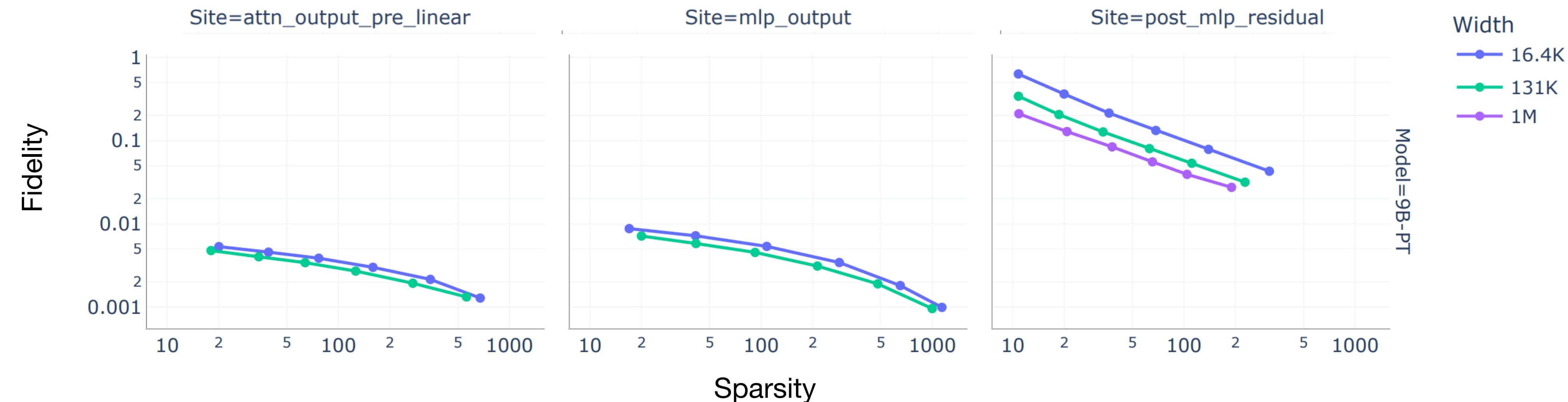


Figure: Sparsity-fidelity trade-off for layer 20 Gemma 2-9B SAEs (Lieberum+24)

Several new SAE architectures (e.g., Top-k SAE(Gao+25), Gated SAE(Rajamanoharan+24), etc.) have been proposed to address these limitations.

! Still an open problem

[1] Lieberum, Tom, et al. "Gemma Scope: Open Sparse Autoencoders Everywhere All At Once on Gemma 2." *BlackboxNLP Workshop 2024*.

[2] Gao, Leo, et al. "Scaling and evaluating sparse autoencoders." *ICLR 2025*.

[3] Rajamanoharan, Senthooran, et al. "Improving Dictionary Learning with Gated Sparse Autoencoders." *CoRR 2024*.

Part 3.2:

Findings on Circuit Study

Discovered circuit (Circuit Study Findings)

- **Circuit Study** have been conducted to understand a range of behaviors across LMs of different sizes.

Circuit	Description	Models
Arithmetic (Nikankin+25)	$1 + 4 \rightarrow 5; 8 - 4 \rightarrow 4; 2 * 4 \rightarrow 8; 8/4 \rightarrow 2$	Llama-3 8b
Factual Recall (Chughtai+25)	LM mechanism for factual recall of stored LM Knowledge	Pythia-2.8b, GPT-2 XL
Induction (Elhage+21)	Completes sentences like “Mr D urs ley was thin and bold. Mr D” with “urs”	GPT-2 Small
Indirect Object Identification (Wang+22)	Completes the sentence with indirect object identification	GPT-2 Small
Greater-than (Hanna+23)	Completes sentences such as “The war lasted from the year 1732 to the year 17” predict valid twodigit end years greater than 32	GPT-2 Small
Subject-verb agreement (Marks+25)	Completes sentences like “The keys in the cabinet” with “are” as opposed to “is”	Pythia-70m

Circuit studies reveal LMs use sparse subgraphs—not the full model—to implement specific behaviors.

[1] Nikankin, Yaniv, et al. "Arithmetic without algorithms: Language models solve math with a bag of heuristics." *ICLR* 2025.

[2] Chughtai, Bilal, et al. "Summing up the facts: Additive mechanisms behind factual recall in llms." *NeurIPS* 2025.

[3] Elhage, et al., "A Mathematical Framework for Transformer Circuits", *Transformer Circuits Thread*, 2021.

[4] Wang, Kevin, et al., "Interpretability in the wild: a circuit for indirect object identification in gpt-2 small." *ICLR* 2022.

[5] Hanna, Michael, et al. "How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model." *NeurIPS* 2023.

[6] Marks, Samuel, et al. "Sparse feature circuits: Discovering and editing interpretable causal graphs in language models." *ICLR* 2025.

“Circuit” in Practice

Variants of circuits:

- *Position-independent (Nikankin+25) vs. position-dependent (Hanna+24);*
- *Allowing connections between non-adjacent layers (Wang+22; Conmy+23) or not (Nikankin+25);*
- *Components/nodes at different granularities, e.g., MLP sub-layers vs. SAE Features (Marks+24);*
- *Circuits vs. Information Flow (Geva+23) vs. Replacement Graphs (Dunefsky+24)*

[1] Nikankin, Yaniv, etc. Arithmetic without algorithms: Language models solve math with a bag of heuristics. *ICLR* 2025.

[2] Hanna, Michael, et al. “How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model.” *NeurIPS* 2023.

[3] Kevin Ro Wang, etc., Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *ICLR* 2022.

[4] Conmy, Arthur, et al. “Towards automated circuit discovery for mechanistic interpretability.” *NeurIPS* 2023.

[5] Marks, Samuel, et al. “Sparse Feature Circuits: Discovering and Editing Interpretable Causal Graphs in Language Models.” *CoRR* (2024).

[6] Geva, Mor, et al. “Dissecting Recall of Factual Associations in Auto-Regressive Language Models.” *EMNLP* 2023.

[7] Dunefsky, Jacob, et al. “Transcoders find interpretable LLM feature circuits.” *NeurIPS* 2024 .

Properties of the Discovered Circuit

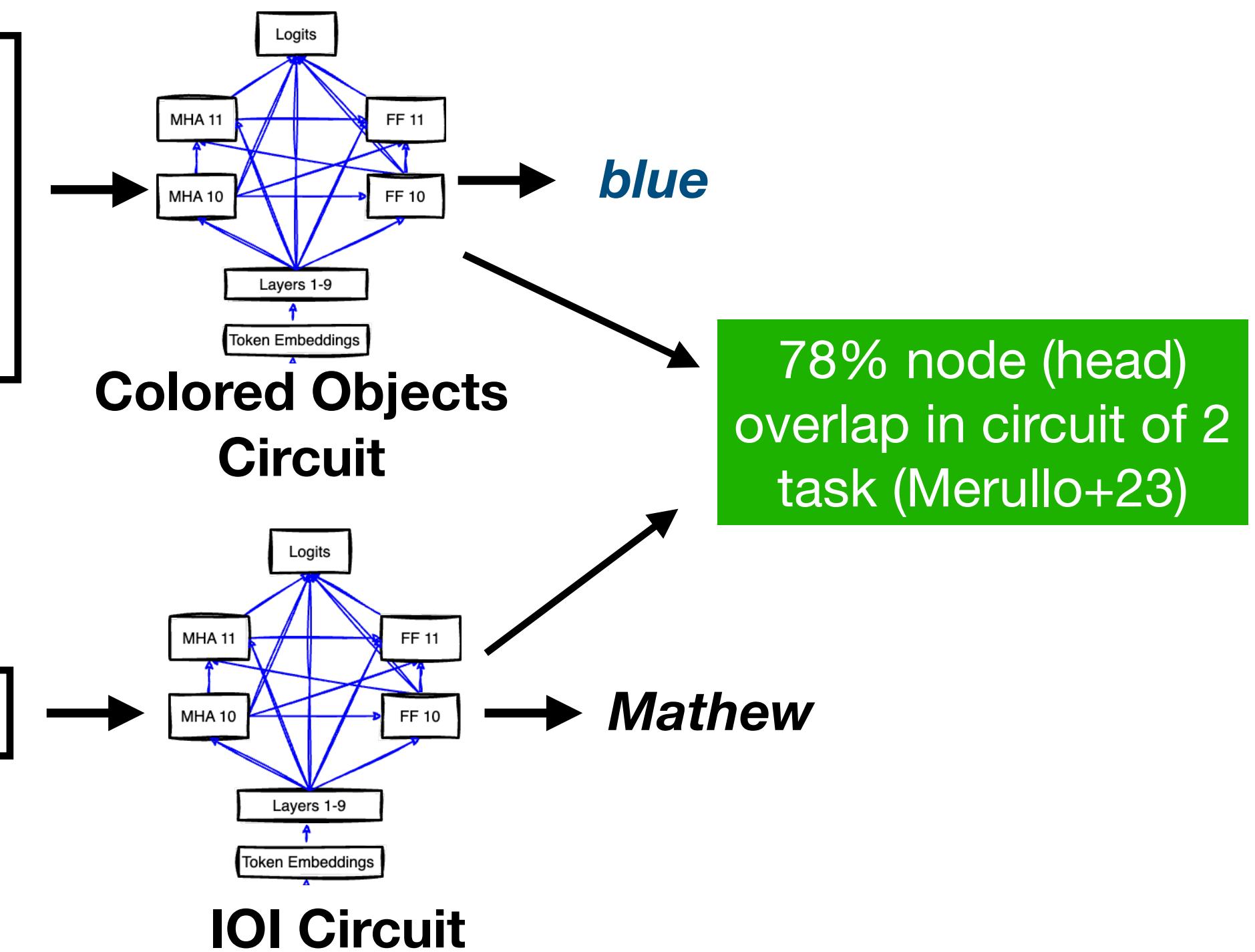
Circuit components are not exclusive to a single circuit but are reused across multiple circuits.

Q: On the table, I see an **orange** textbook, a **red** puzzle, and a **purple** cup.
What color is the textbook?
A: Orange
Q: On the table, there is a **blue** pencil, a **black** necklace, and a **yellow** lighter. What color is the pencil?

Task 1: Colored Objects task

Matthew and **Robert** had a lot of fun at the school. **Robert** gave a ring to

Task2: Indirect Object Identification (IOI) task



LM behavior may be explained by a few **interpretable, task-general** components.

Properties of the Discovered Circuit

Model behavior often arises from the joint effect of several circuits, not a single isolated one.

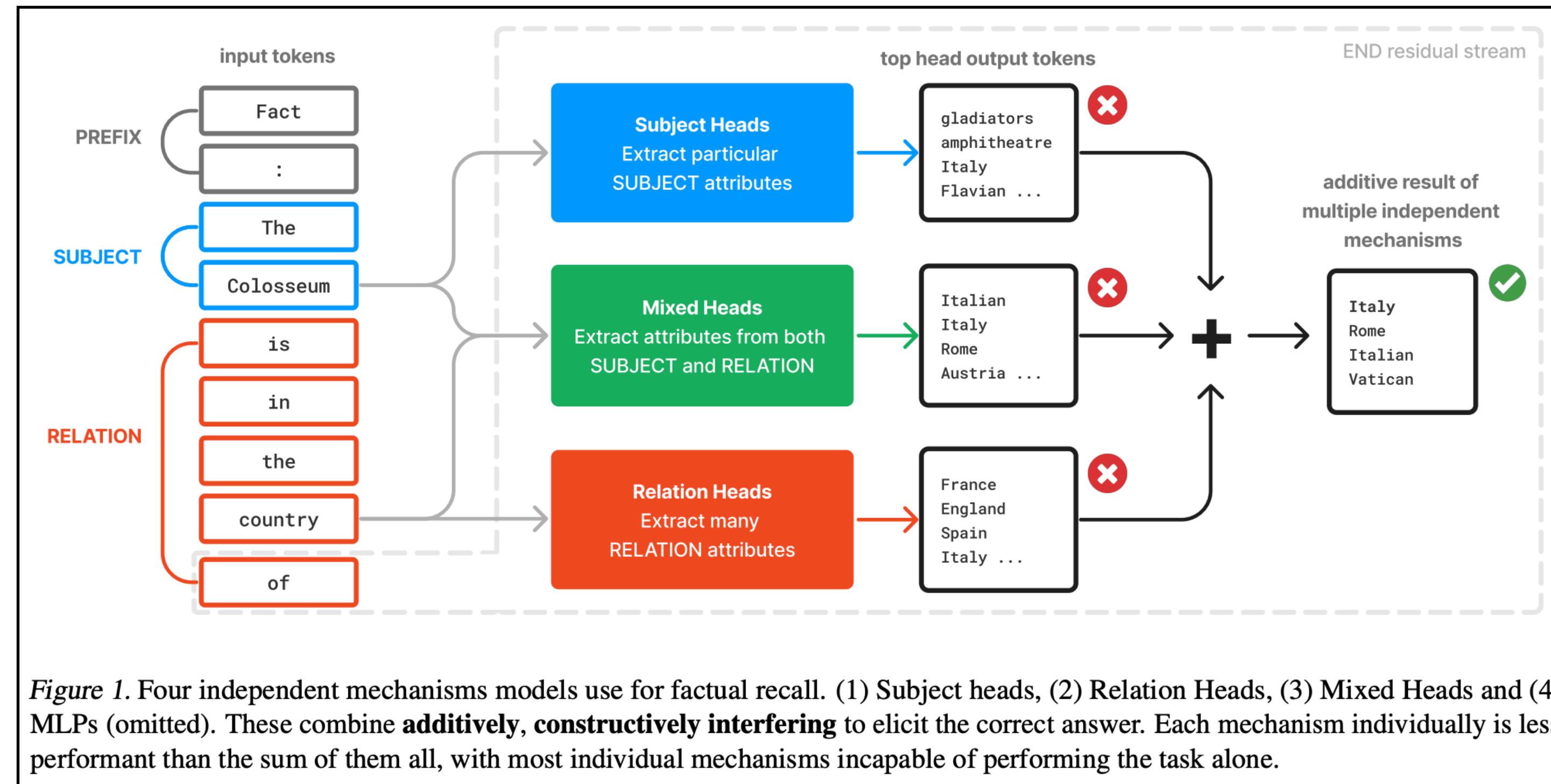


Fig: Chughtai+24 discovered four distinct mechanisms that interact additively to extract facts.

LMs implement multiple circuits to solve the same task.

Properties of the Discovered Circuit

Model behavior often arises from the joint effect of several circuits, not a single isolated one.

- The mechanism or circuits have **varying levels of reliability** on the task (Rai+25).
- The model often makes **mistake** when outputs of **unreliable mechanism overshadows the reliable mechanism**.
- More reliable mechanism can be manually promoted more to **improve the model performance**.

Failure by Interference: Language Models Make Balanced Parentheses Errors When Faulty Mechanisms *Overshadow* Sound Ones

Daking Rai
Department of Computer Science
George Mason University
Fairfax, VA 22030
drai2@gmu.edu

Samuel Miller
Department of Computer Science
George Mason University
Fairfax, VA 22030
smille20@gmu.edu

Kevin Moran
Department of Computer Science
University of Central Florida
Orlando, Florida 32816
kpmoran@ucf.edu

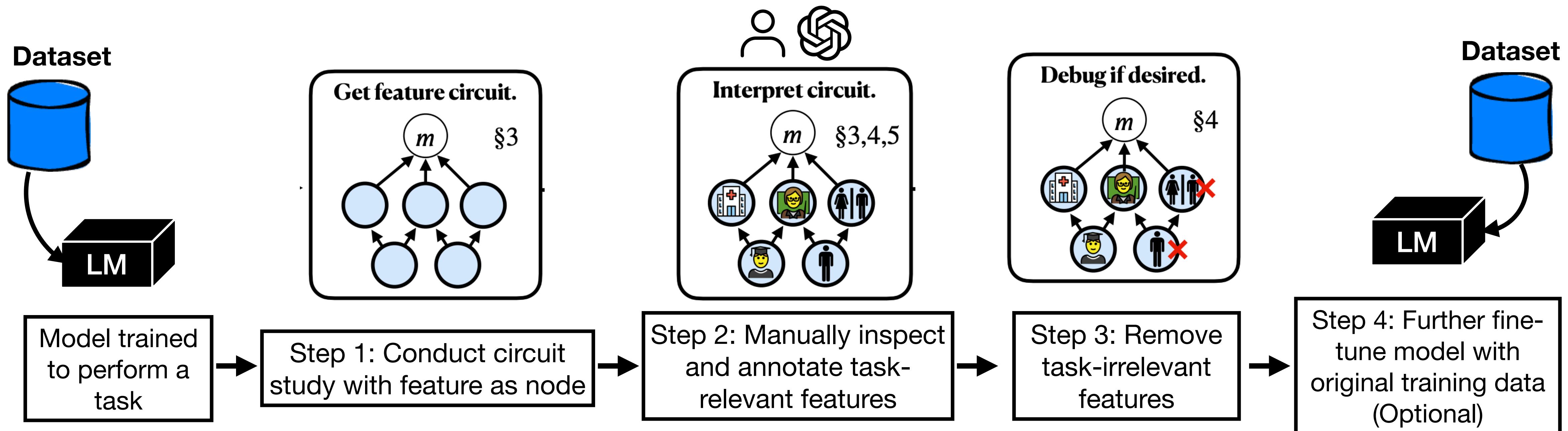
Ziyu Yao
Department of Computer Science
George Mason University
Fairfax, VA 22030
ziyuyao@gmu.edu

LMs implement multiple circuits to solve the same task.

Properties of the Discovered Circuit

Circuit can be edited to improve the performance

Marks+24 introduced **SHIFT**, an approach where human evaluators identify and ablate task-irrelevant features from the circuit.



SHIFT outperformed other baseline (including supervised finetuned classifier) for classification task.

Part 3.3: Findings on Universality Study

Findings on Universality (Features)

Key Question: Do LLMs learn the **same underlying features** regardless of their specific training or architecture?

Same Architecture

- Gurnee+24 found that only **1–5% of FF neurons** exhibited universality across GPT-2 models trained from **different random seeds**.
- Lan+24 found that **SAE features** show a high degree of similarity across Pythia-70m vs Pythia-120m.

Different Architecture

- Wang+24 found that **74% of the SAE features** are similar between transformer and Mamba models, when trained on the same dataset.

Key takeaway: Mixed results on the universality of features.

Future work should investigate influencing factors: weight initialization, model size, model architecture, etc.

[1] Gurnee, Wes et al. “Universal neurons in gpt2 language models.” TMLR 2024.

[2] Lan, Michael et al. “Sparse autoencoders reveal universal feature spaces across large language models.” arXiv preprint arXiv:2410.06981, 2024.

[3] Wang, Junxuan et al. “Towards universality: Studying mechanistic similarity across language model architectures.” ICLR 2025.

Findings on Universality (Circuits)

Key Question: Do LMs learn the **similar circuits** to perform similar tasks?

Positive results on universality of circuits

Circuit	Description	Models
Induction (Elhage+21)	Completes sentences like “Mr D urs ley was thin and bold. Mr D” with “urs”	Toy model, GPT-2 Small
Successor heads (Gould+23)	Circuit that increment tokens with a natural ordering, such as numbers, months, and days (e.g., ‘Monday’ into ‘Tuesday’)	GPT-2, Pythia, Llama-2
Previous token heads (Wang+22)	Copies the token from previous token positions	Toy model, GPT-2 Small
IOI Circuit (Tiggs+24)	Completes sentences like “When John and Mary went to the store, John gave a drink to” with “Mary” as opposed to “John”.	GPT-2, Pythia
Greater-than (Tiggs+24)	Completes sentences such as “The war lasted from the year 1732 to the year 17” predict valid twodigit end years greater than 32	GPT-2, Pythia

[1] Elhage, et al., "A Mathematical Framework for Transformer Circuits", Transformer Circuits Thread, 2021.

[2] Chughtai, Bilal. "A toy model of universality: Reverse engineering how networks learn group operations. In International Conference on Machine Learning." PMLR, 2023.

[3] Wang, Kevin, et al. "Interpretability in the wild: a circuit for indirect object identification in gpt-2 small." ICLR 2022.

[4] Tigges, Curt, et al. "Llm circuit analyses are consistent across training and scale." NeurIPS 2024.

Findings on Universality (Circuits)

Key Question: Do Large Language Models (LLMs) learn the **similar internal "circuits"** to perform similar tasks?

Negative results on universality of circuits

- Zhong+24 showed LMs (toy one-layer ReLU transformer) trained with different initializations can develop qualitatively different circuits for the modular addition task.
 - Implication: There isn't always one "best" or universal circuit solution.
- Chughtai+23 found that different initializations led to dissimilar representations and circuits for same task.

Key takeaway: Mixed results on the universality of circuits.

Future work should investigate influencing factors: Weight initialization, Model size, Model Architecture, etc.

[1] Chughtai, Bilal, et al. "A toy model of universality: Reverse engineering how networks learn group operations." PMLR, 2023.

[2] Zhong, Ziqian, et al. "The clock and the pizza: Two stories in mechanistic explanation of neural networks." NeurIPS 2024.