Toward Faithful Retrieval-Augmented Generation with Sparse Autoencoders

Guangzhi Xiong, Zhenghao He, Bohan Liu, Sanchit Sinha, Aidong Zhang

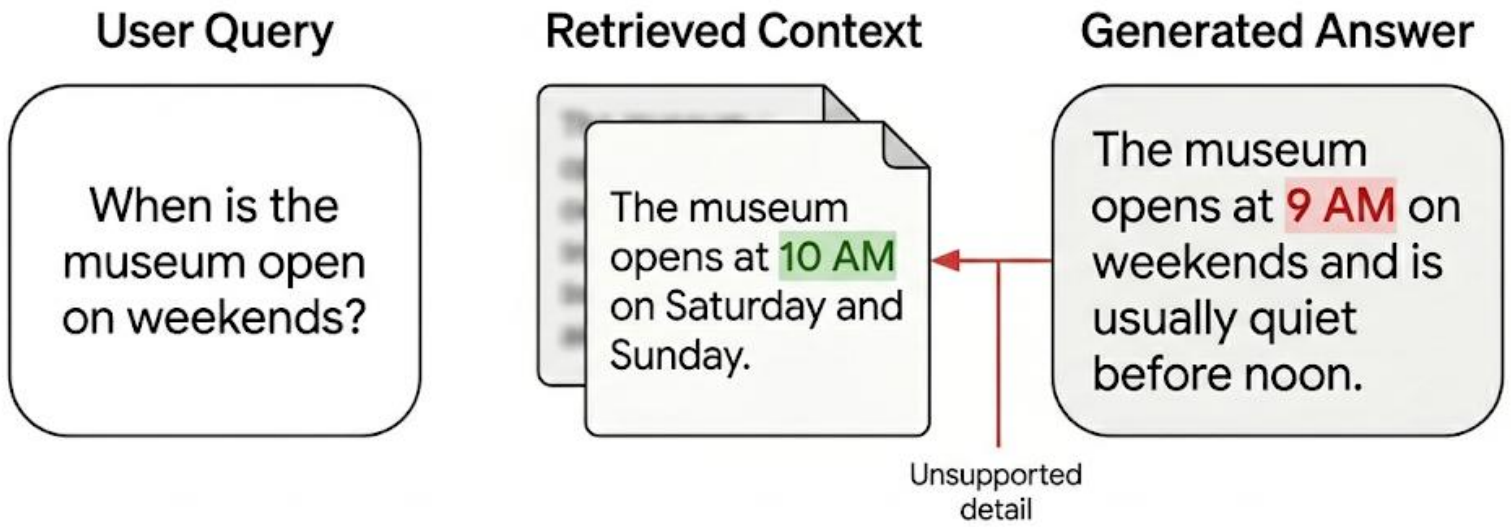
University of Virginia

University of Virginia | ENGINEERING Department of Computer Science

ICLR

LLMs still hallucinate even with retrieved evidence

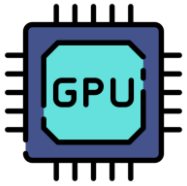
- Retrieval-augmented generation (RAG) helps factuality of LLM outputs
- However, the models can still hallucinate information that is not grounded in the retrieved context.



Requirements of training/inference resources in existing methods

Training Dedicated Detectors

Annotated Data



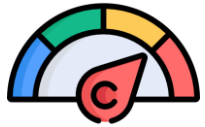
Training Compute

Hallucination Detector

Querying External LLM Judges

LLM Output

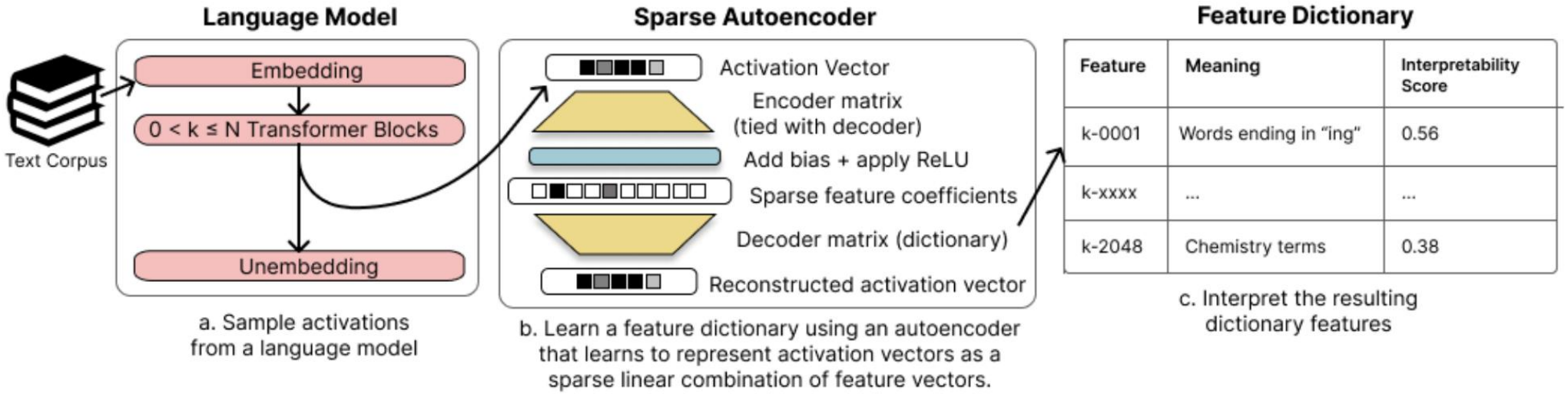
External LLM Judge



Detection Score

Sparse autoencoders can find interpretable features

- Sparse autoencoders (SAEs)[1] can disentangle specific, semantically meaningful features from the hidden states of LLMs

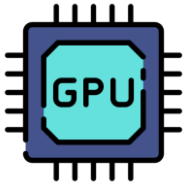


[1] Sparse autoencoders find highly interpretable features in language models. ICLR. 2024.

Do LLMs know when they are hallucinating?

Training Dedicated Detectors

Annotated Data



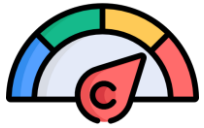
Training Compute

Hallucination Detector

Querying External LLM Judges

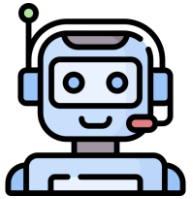
LLM Output

External LLM Judge

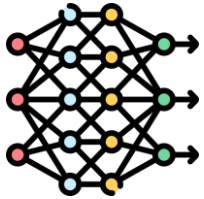


Detection Score

Exploring LLM Internal Signals



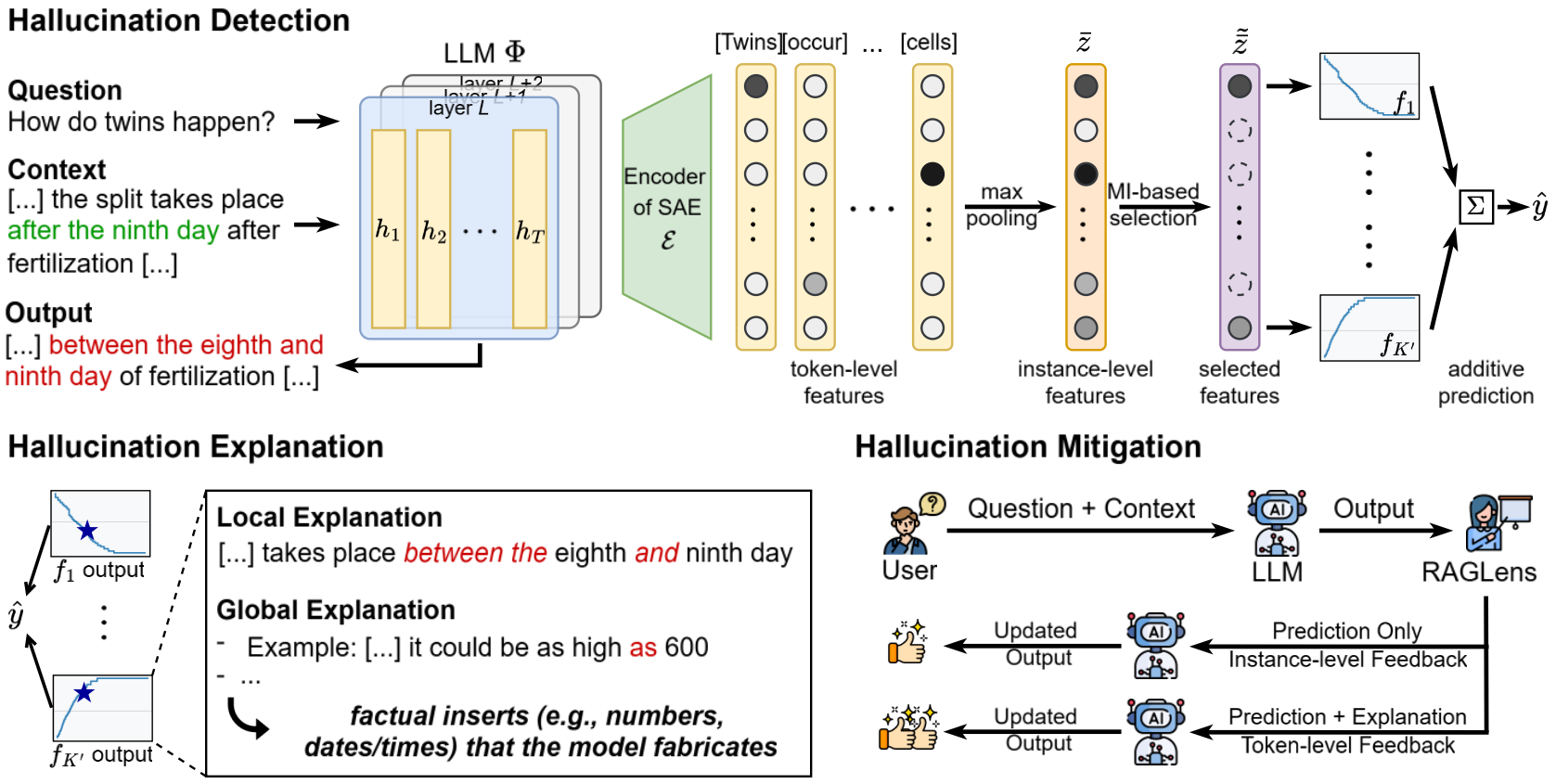
Tested LLM



Hidden Activations

LLM Output

RAGLen: Faithful RAG via sparse representation probing



RAGLens supports the ***detection***, ***explanation***, and ***mitigation*** of unfaithful RAG outputs using interpretable sparse features

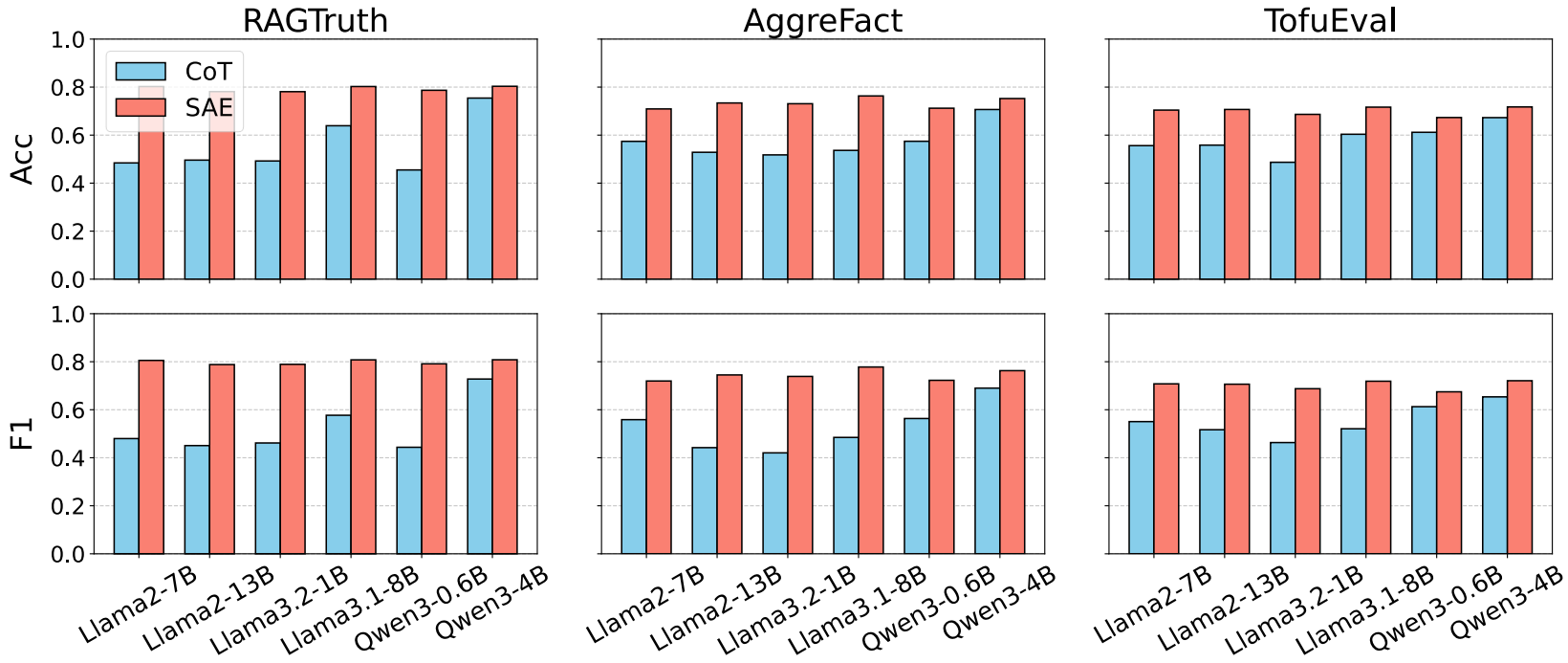
Detection: Performance on RAG hallucination detection

- RAGLens outperforms existing prompting-based, uncertainty-based, internal representation-based detectors on self-detection.

Method	RAGTruth (Llama2-7B)			Dolly (Llama2-7B)			RAGTruth (Llama2-13B)			Dolly (Llama2-13B)		
	AUC	Acc	F_1	AUC	Acc	F_1	AUC	Acc	F_1	AUC	Acc	F_1
Prompt	–	0.6700	0.6720	–	0.6200	0.5476	–	0.7300	0.6899	–	0.6700	0.5823
Llama2-13B(LR)	–	0.6350	0.6572	–	0.6043	0.6616	–	0.7044	0.6725	–	0.5545	0.6664
LwMLM	–	0.6940	0.7365	–	0.6550	0.7702	–	0.5956	0.7684	–	0.6800	0.7000
FAcTScore	0.5428	0.5333	0.6719	0.4813	0.5354	0.6849	0.5294	0.4533	0.6239	0.4389	0.4646	0.5954
LN-Entropy	0.5912	0.5620	0.6850	0.6074	0.5656	0.6261	0.5912	0.5620	0.6850	0.6074	0.5656	0.6261
Energy	0.5619	0.5088	0.6657	0.6074	0.5656	0.6261	0.5619	0.5088	0.6657	0.6074	0.5656	0.6261
Focus	0.6233	0.5533	0.6522	0.6783	0.6212	0.6545	0.7888	0.6000	0.6758	0.7067	0.6500	0.6567
ITI	0.6714	0.5667	0.6496	0.5494	0.5800	0.6281	0.8501	0.6177	0.6850	0.6530	0.5583	0.6712
ReDeEP	0.7458	0.6822	0.7190	0.7949	0.7373	0.7833	0.8244	0.7889	0.7587	0.8420	0.7070	0.7603
RAGLens (Ours)	**0.8413**	**0.7576**	**0.7636**	**0.8764**	**0.7778**	**0.8070**	**0.8964**	**0.8333**	**0.8148**	**0.8568**	**0.7576**	**0.7895**

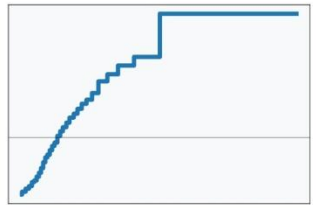
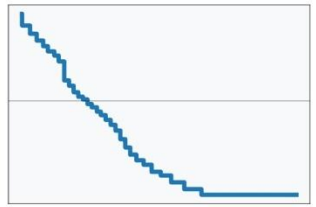
Detection: Internal signals vs. self-judgment across models

- LLMs contain sufficient internal knowledge to detect other models' hallucinations
- The SAE-based detector outperforms each model's own CoT-style self-judgments



Interpretability: RAGLens discovers interpretable features

- RAGLens discovers features that are most indicative of RAG hallucinations
- RAGLens also illustrates how each feature contributes to the final prediction through learned shape functions

ID	Explanation	Examples	Shape Plot
22790	unsupported numeric/time specifics	Context: no mention of age Output: [...] at the age **of** 34 [...]	
		Context: no mention of release schedule Output: [...] to be **released in** August [...]	
17721	grounded, high-salience tokens	Context: [...] could be arrested on the spot [...] Output: [...] could be **arrested** on the spot [...]	
		Context: [...] software can be licensed as a [...] Output: [...] software can be **licensed** as a [...]	

Interpretability: Identified features are RAG-specific

- Perturbation on the context validates that feature is specialized for detecting RAG hallucinations rather than general hallucination

Model Input
Briefly answer the following question: how to tell the temperature of water with a candy thermometer Bear in mind that your response should be strictly based on the following three passages: passage 1: [...] If the candy forms a soft, flexible ball it is at a temperature of 235-240 degrees Fahrenheit or at the soft ball stage. passage 2: [...] If it forms into a firm ball, it is in the firm ball stage. This stage is at a temperature of about 245 to 250 degrees Fahrenheit [...] passage 3: [...] In case the passages do not contain the necessary information to answer the question, please reply with: "Unable to answer based on given passages."
Model Output (annotated with feature activation)
Based on the provided passages, here is how to tell the temperature of water using a candy thermometer: Passage 1 states that when the candy is in the thread stage, the temperature is around 230-235°Fahrenheit. Passage 2 indicates that the soft ball stage occurs at a temperature of 245-250°Fahrenheit for candies such as fudge and fondant. Therefore, to determine the temperature of water using a candy thermometer, you can follow these steps: 1. Cool some of the candy in the thread stage by dropping it into cold water. 2. Observe if the candy forms a soft, flexible ball. If it does, the temperature is between 245-250°Fahrenheit, indicating that the water is at the soft ball stage. If the water is not at the soft ball stage, continue cooling it until it reaches that temperature range, and then proceed to use the candy thermometer to measure its temperature.

Context Perturbation 1 (context removed)
Original Context: [...] Bear in mind that [...] **Updated Context**: None **Updated Activation**: [...] indicates that the soft ball stage occurs at a temperature of around 245-250°Fahrenheit [...]
Context Perturbation 2 (context consistent)
Original Context: [...] This stage is at a temperature of about 245 to 250 degrees Fahrenheit [...] **Updated Context**: [...] The soft ball stage is at a temperature of about 245 to 250 degrees Fahrenheit [...] **Updated Activation**: [...] indicates that the soft ball stage occurs at a temperature of around 245-250°Fahrenheit [...]

Mitigation: Mitigating unfaithfulness with RAGLens

- RAGLens can provide post-hoc feedback to LLMs to mitigate hallucinations
  - Both instance-level and token-level feedback effectively reduce hallucinations in the revised output.
  - The token-level feedback enabled by RAGLens interpretability leads to further reductions compared to instance-level feedback.

	Llama3.3-70B	GPT-4o	GPT-o3	Human
Original	43.78%	37.78%	64.44%	71.11%
+ Instance-level Feedback	42.22%	36.44%	60.44%	62.22%
+ Token-level Feedback	**39.11%**	**34.22%**	**58.88%**	**55.56%**

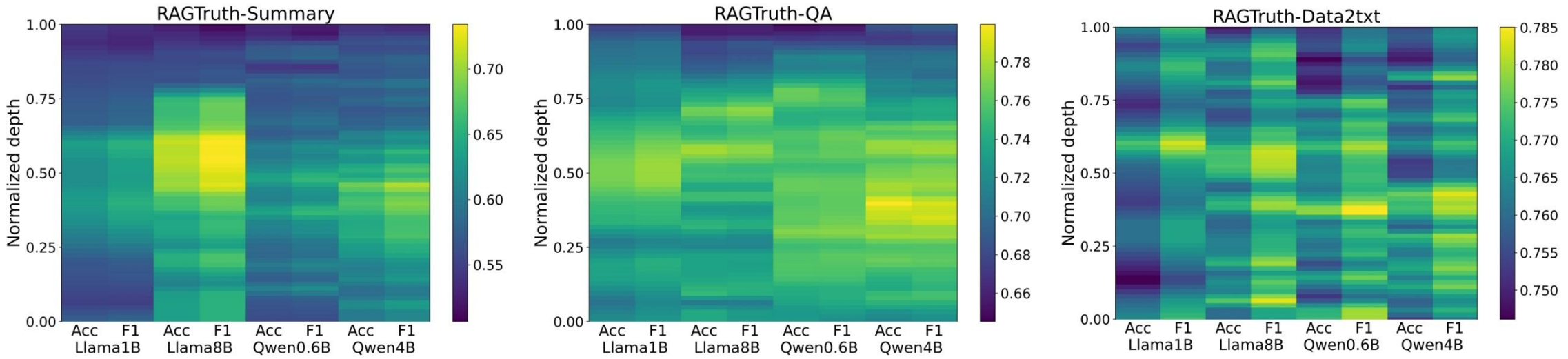
Mitigation: Causal intervention of SAE features

- The identified features not only correlate with RAG hallucination but also play a causal role in driving unfaithful generations.

Context	Prefix	Value	Output
No mention of the age	*[...] at the age of*	-20.0	*[...] of 30.*
		0.0	*[...] of 25.*
		20.0	*[...] of an unspecified age.*
No mention of the release date	*[...] scheduled to be released in*	-20.0	*[...] in 2016.*
		0.0	*[...] in the future.*
		20.0	*[...] in an unspecified time frame.*

Discussion: LLM layer selection

- The performance trend in layers is consistent among LLMs but varies by task.
  - In the Summary and QA tasks of RAGTruth, the performance peaks around the middle layers
  - The Data2txt task exhibits a comparatively flat performance pattern across layers.



Discussion: Feature extractor comparison

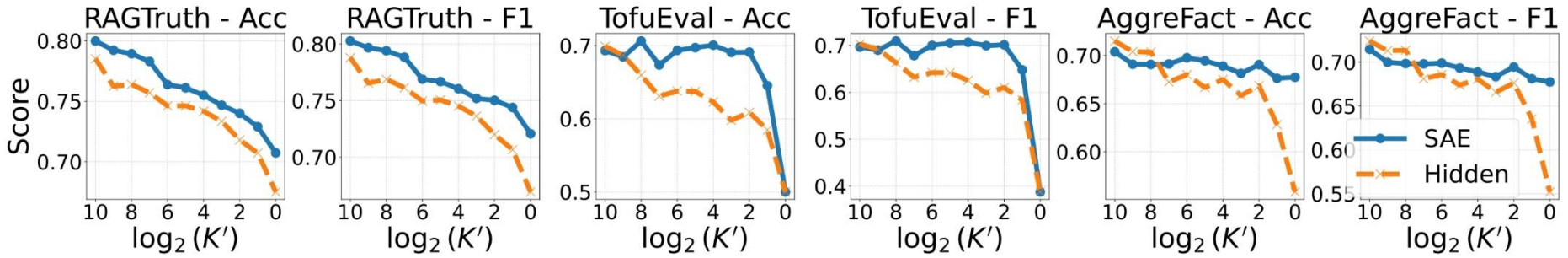
- Pre-activation SAE feature retain more informative signals about RAG hallucinations

SAE vs. Transcoder

Pre-activation vs. Post-activation

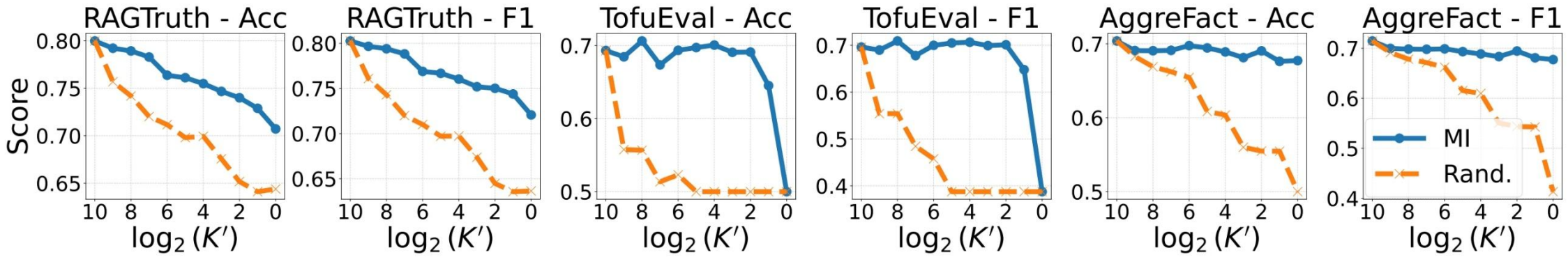
Architecture	Activation	RAGTruth		AggreFact		TofuEval	
		Acc	F_1	Acc	F_1	Acc	F_1
SAE	Pre-activation	0.7810	0.7892	0.7308	0.7388	0.6865	0.6876
	Post-activation	0.7606	0.7700	0.6939	0.7091	0.5637	0.5642
Transcoder	Pre-activation	0.7778	0.7830	0.7468	0.7586	0.6652	0.6666
	Post-activation	0.7594	0.7684	0.7373	0.7525	0.6195	0.6178

SAE vs. Raw Hidden States



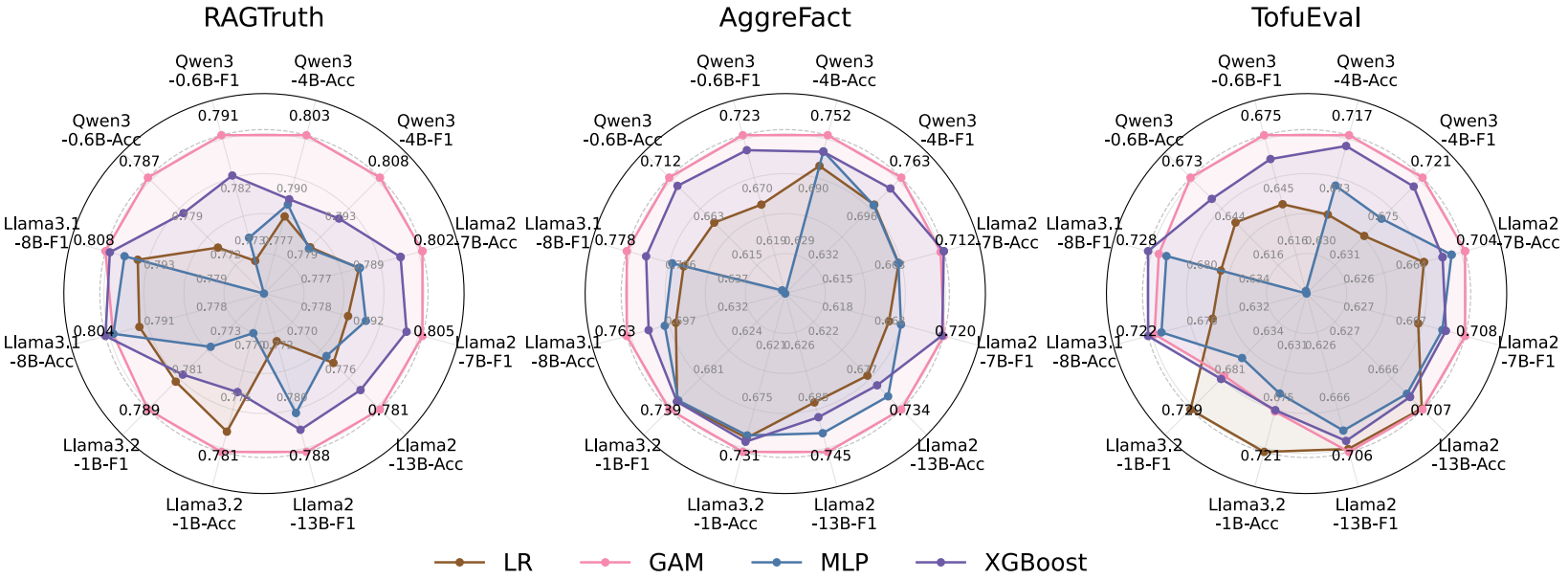
Discussion: Analysis of feature count

- Detection performance drops when fewer features are used.
- MI effectively prioritizes informative features for hallucination detection.



Discussion: Predictor ablation

- GAM consistently outperforms LR and also surpasses more complex models such as MLP and XGBoost.
  - The overall contribution of SAE features can be effectively captured in an additive manner.



Thank You!

Homepage: https://gzxiong.github.io/RAGLens

Code: https://github.com/gzxiong/RAGLens