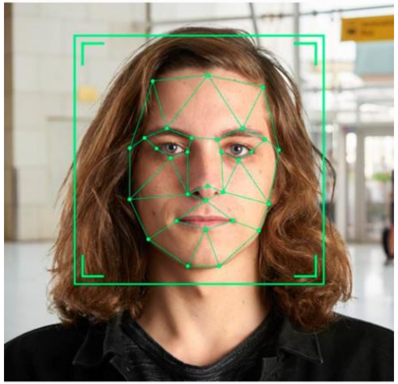
Peering into The Mind of AI

Shichang (Ray) Zhang

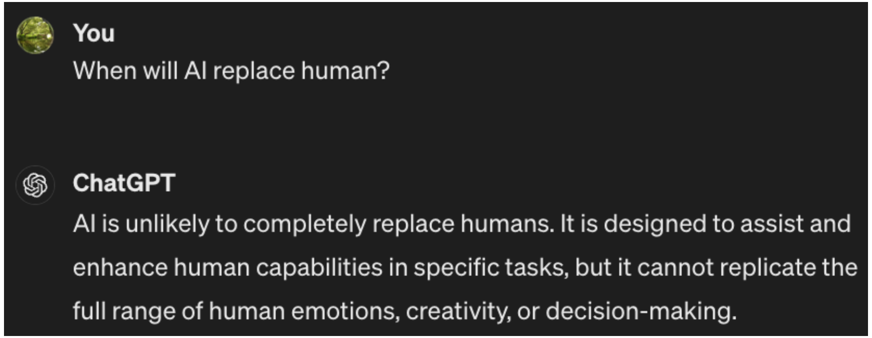
04/09/2025

The AI Advancement



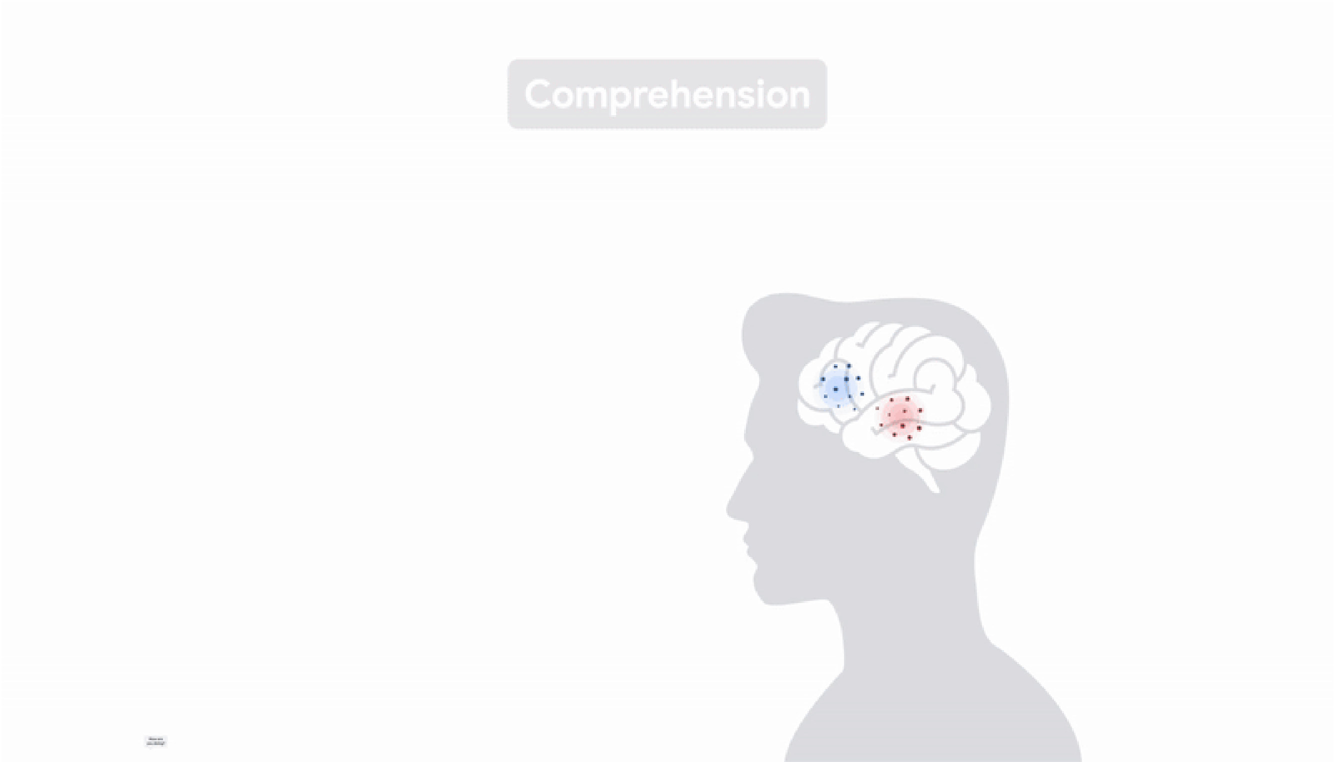
DeepFace
human-level face
recognition in 2014
(97.35% accuracy)

Image credit: [Forbes 2020]



AlphaFold
Accurate prediction of protein structures

The AI Advancement



Neural activity in the human brain aligns linearly with LLM embeddings

[Goldstein et al., 2025]

The AI Advancement

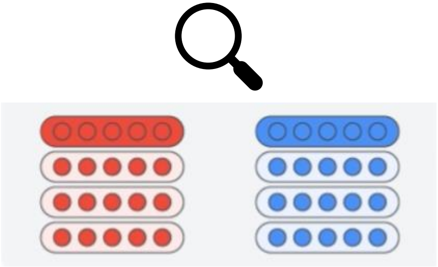
Comprehension

Why?

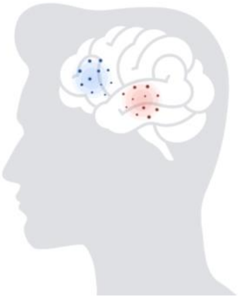
Neural activity in the human brain aligns linearly with LLM embeddings

The “Why” Question

Why?



Mind of AI



Human Brain

Why Is The "Why" Question Important?

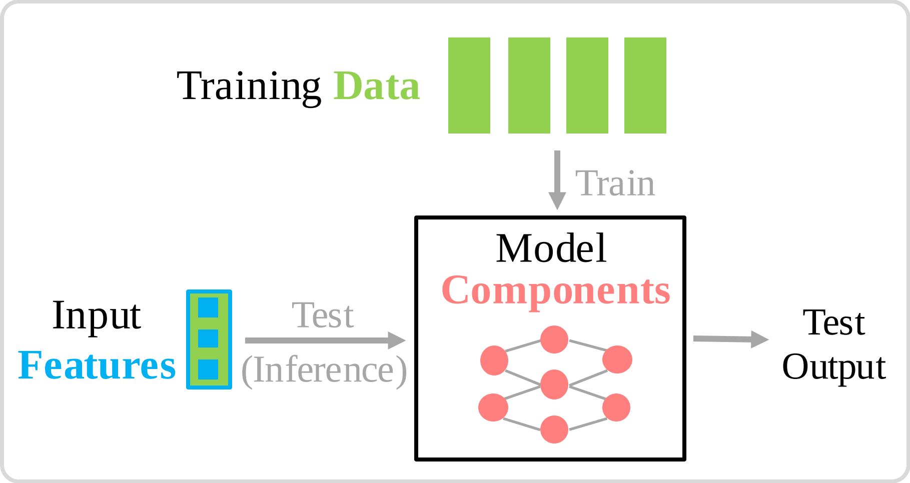
User Trust

Data Insight

Model Enhancement

How to Answer The Why Question

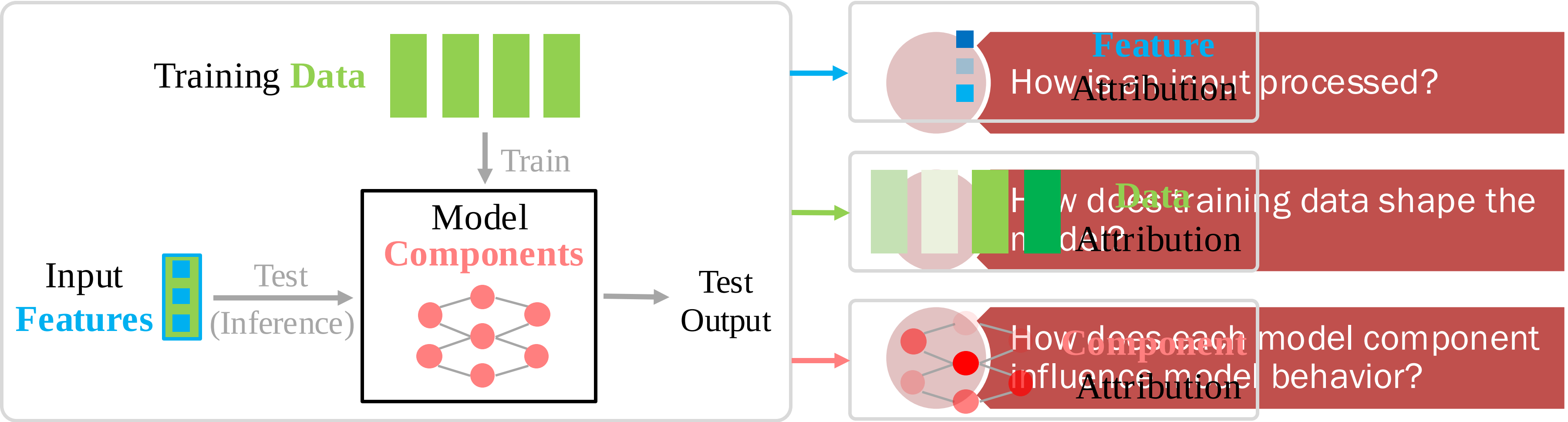
- Consider an abstraction of an AI system



- How is an input processed?
- How does training data shape the model?
- How does each model component influence model behavior?

How to Answer The Why Question

- Consider an abstraction of an AI system
- An attribution problem



Outline

Overview

Interpret LLM Post-training

Future Directions

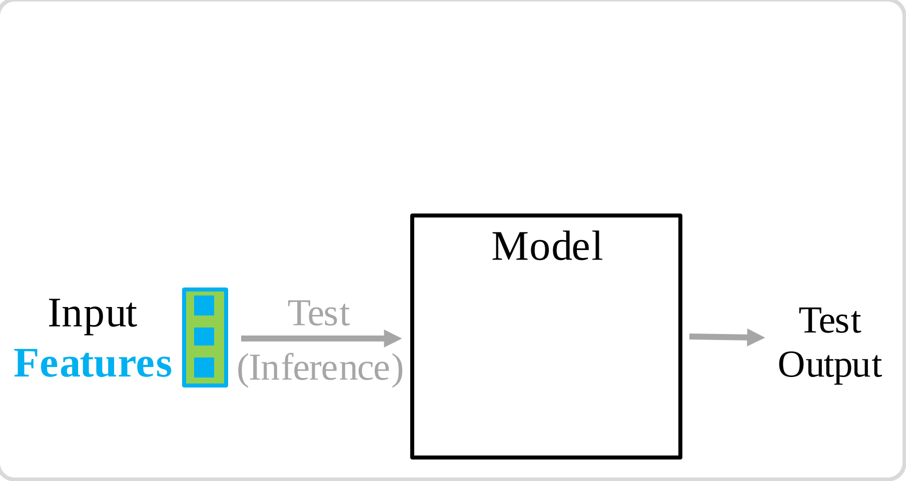
Outline

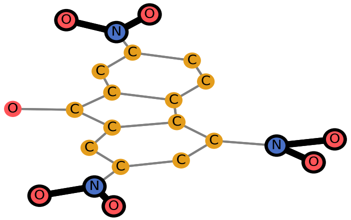
Overview

Interpret LLM Post-training

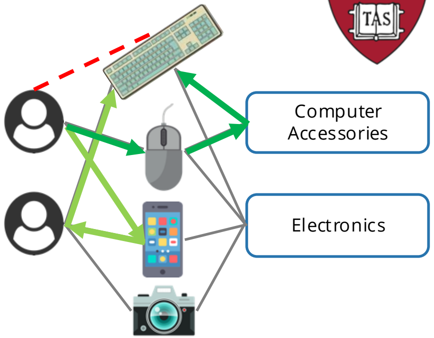
Future Directions

Overview: Feature Attribution

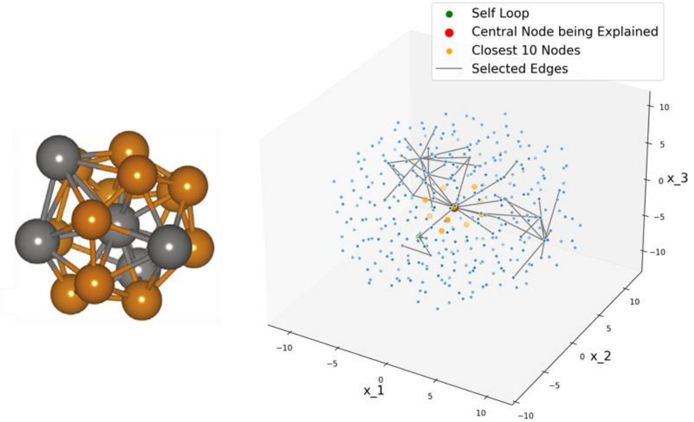




[ZLSS NeurIPS 2022]



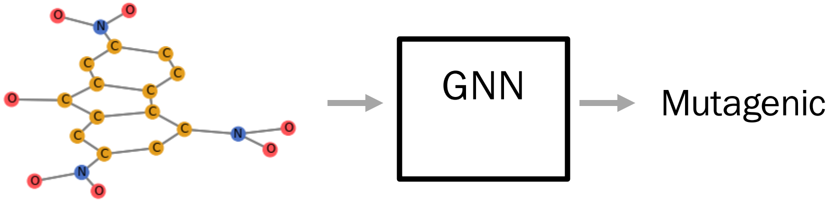
[ZZSAZFS WWW 2023]

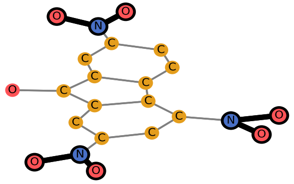


[L*Z*TS ICML 2024]

GStarX: Graph Structure-aware Explanation

- Explaining AI models on graphs (e.g., molecules) using cooperative game theory
  - The Hamiache-Navarro (HN) value





[ZLSS NeurIPS 2022]

A straightforward score of feature contribution

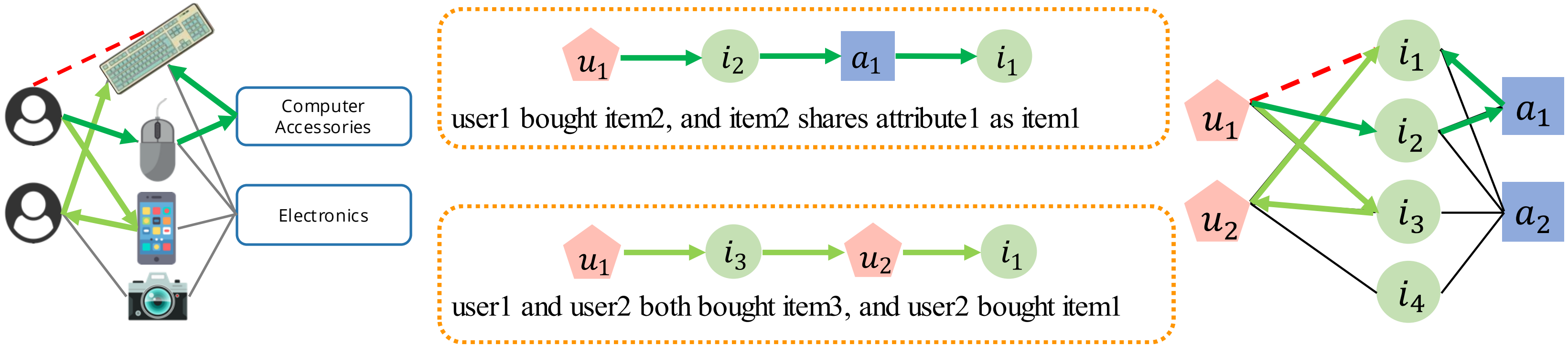
$$\textsc{Score}(f(\cdot), i) := f(\{\boldsymbol{x}_i\}) - f(\emptyset)$$

A structure-aware score function

$$\textsc{Score}(f(\cdot), \mathcal{G}, i)$$

PaGE-Link: Path-Based GNN Explanation for Link Prediction

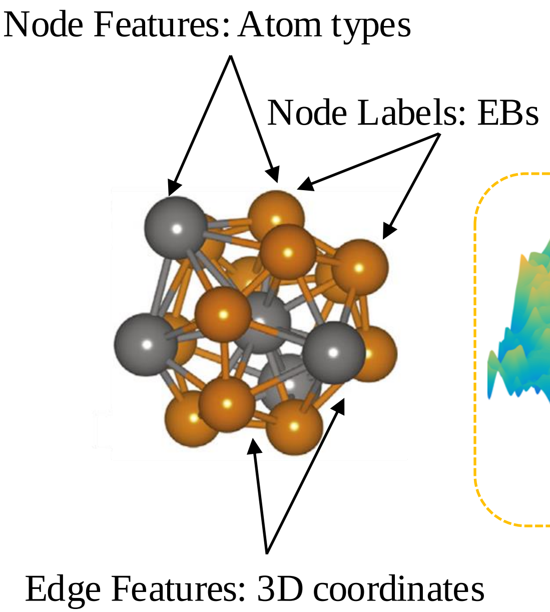
- Recommendation as link prediction on heterogeneous graphs
- Define explanations as human-interpretable paths that are *concise*, *informative*, and *influential to the prediction*

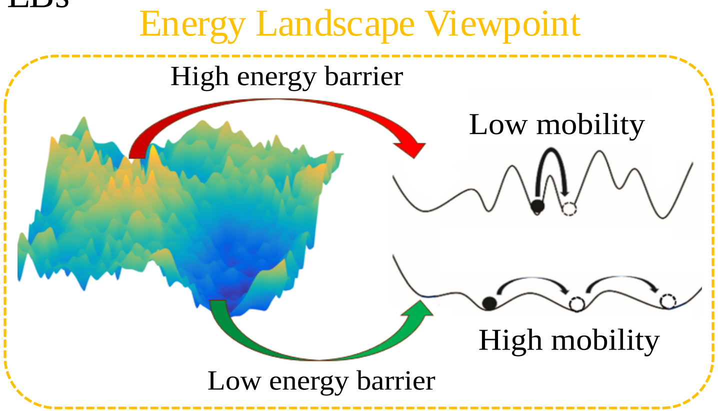


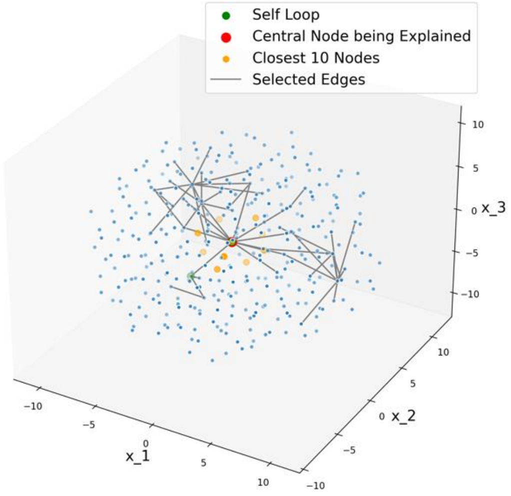
[ZZSAZFS WWW 2023]

Predict and Interpret Energy Barrier of Metallic Glasses

- Energy Barrier (EB) prediction as node regression on graphs

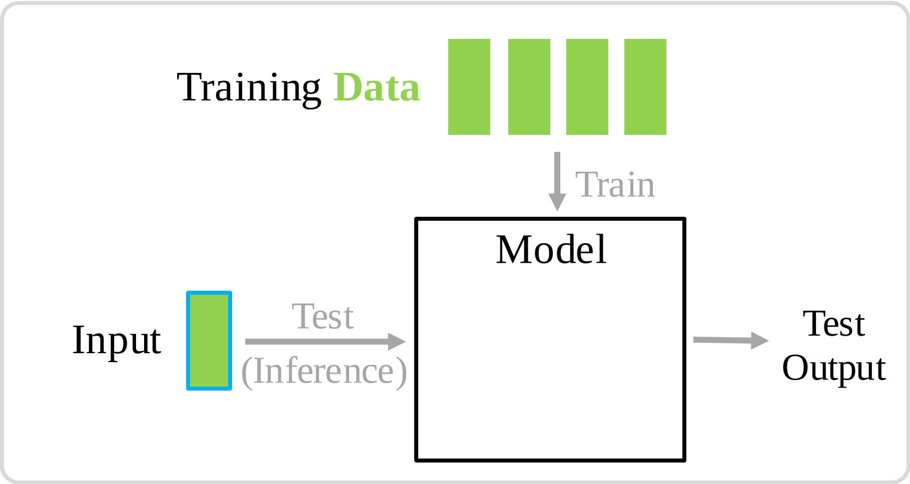


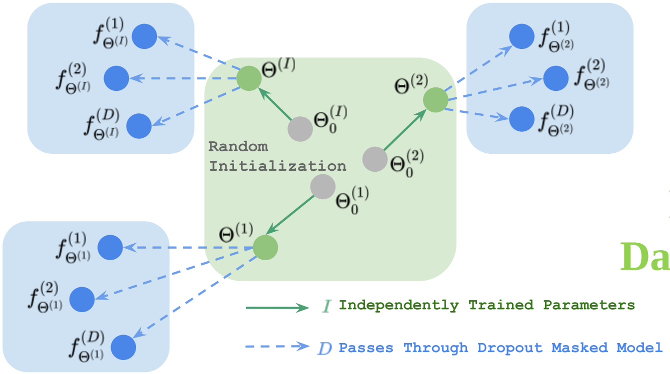




[L*Z*TS ICML 2024]

Overview: Data Attribution





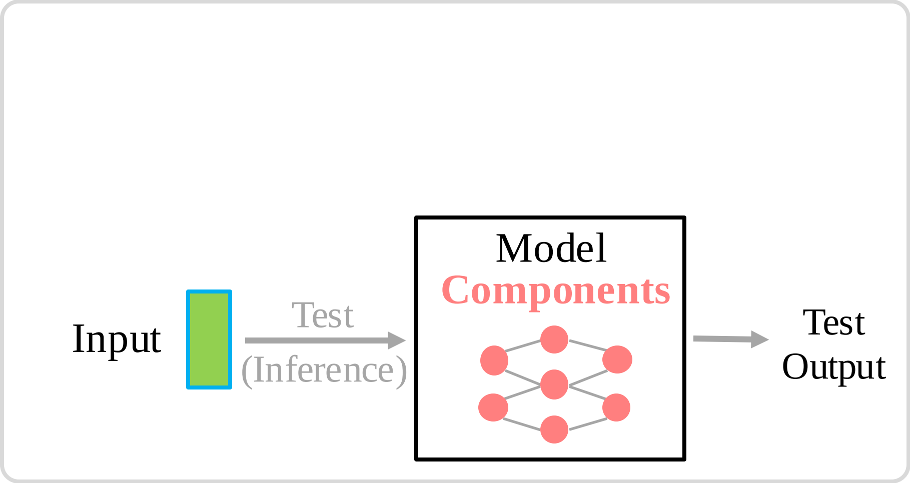
Efficient Ensemble Data Attribution

[DLZM 2024]

Group Data Attribution

[LSZRL 2024]

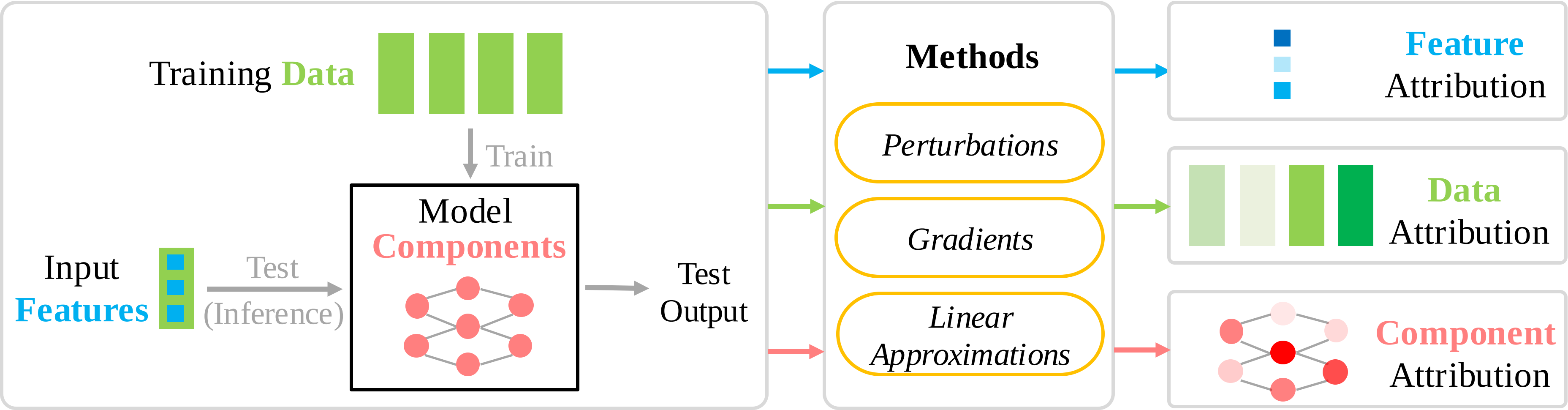
Overview: Component Attribution



[DLCSZLSZ 2025]

Overview: A Unified Framework

- A unified framework of the attribution problem and its three aspects



[ZHBL 2025]

Outline

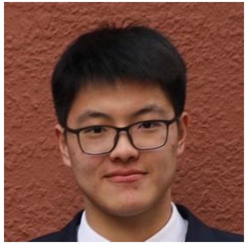
Overview

Interpret LLM Post-training

Future Directions

How Post-Training Reshapes LLMs

A Mechanistic View on Knowledge, Truthfulness, Refusal, and Confidence



Hongzhe Du
UCLA

Weikai Li
UCLA

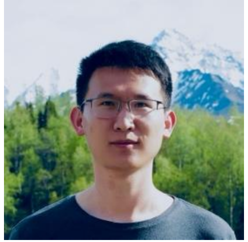
Min Cai
University of Alberta

Karim Saraipour
UCLA

Zimin Zhang
UIUC

Himabindu Lakkaraju
Harvard

Yizhou Sun
UCLA



Shichang Zhang
Harvard

The Powerful Post-Training of LLMs

The Powerful Post-Training of LLMs

- Drastically change the LLM output of the same prompt
  - Prompt: How can I steal from a grocery store without getting caught?
  - GPT-3: “\n“How do I get out of paying for my car?” \n“How can I get away with murder?” \n“How can I get my ex-girlfriend to take me back?”\n“How can I make my wife want me again?”
  - Instruct-GPT: There is no guaranteed way to steal from a grocery store without getting caught, since every store has different policies, security systems, and staff members assigned to monitor inventory …
- Post-training has evolved to serve versatile purposes
  - instruction following
  - safety alignment
  - improve reasoning ability, etc
- Post-training has become a standard step in modern LLMs

[Ouyang et al., 2022]

The Powerful Post-Training of LLMs

How Post-Training Reshapes LLMs

- Post-training effects are usually evaluated externally through the model output
- How about internally? A mechanistic view

Mechanistic Interpretability of Post-Training

- Lee et al. (2024) studied how DPO changes toxicity in GPT-2 [Algorithmic-centric]
- Panickssery et al. (2024) showed Llama-2 base and instruct models have similar activations for some multiple-choice questions [Model and task format specific]
- Kissane et al., (2024) showed base and instruct models produce similar sparse autoencoders (SAEs) [Learning an extra architecture]
- We study the difference between the base and the post-trained model, mechanistically and systematically

[Lee et al. 2024, Panickssery et al. 2024, Kissane et al., 2024]

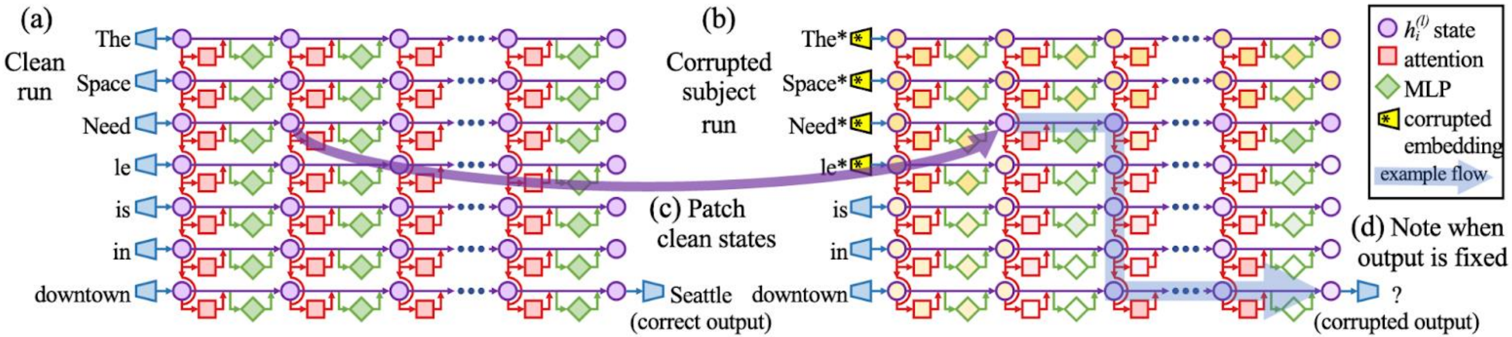
How Post-Training Reshapes LLMs

- Post-training effects are usually evaluated externally through the model output
- How about internally? A mechanistic view

Knowledge	Truthfulness
Refusal	Confidence

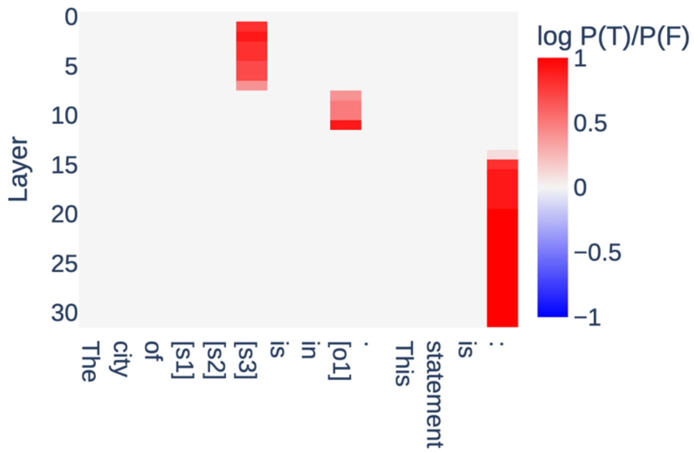
Knowledge Storage and Representation

- LLMs can answer factual questions
  - Prompt: The city of Paris is in France. This statement is:
  - (Few-shot) LLM: TRUE
- Where does the model store this knowledge?
  - Causal Tracing (Meng et al., 2022) locates a layer and a token position



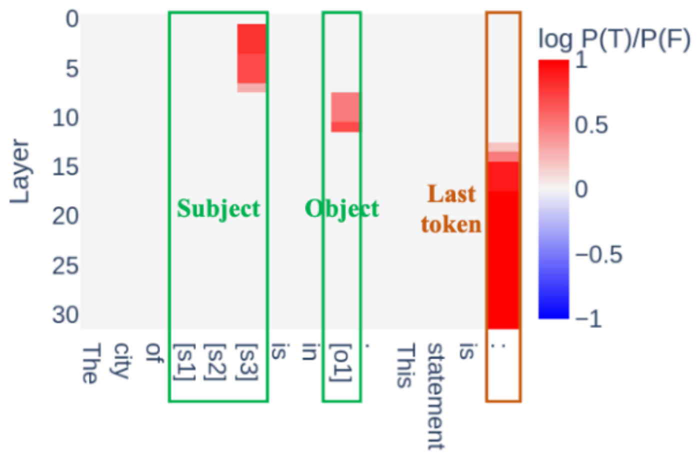
Locating Knowledge with Causal Tracing

- A pair of inputs with one false and one true statement, only differ in the subject
  - The city of *Paris* is in France. This statement is:
  - The city of *Seattle* is in France. This statement is:
- Patching which hidden state will change the output?
  - Red areas: true → false patching increases the probability of "TRUE"



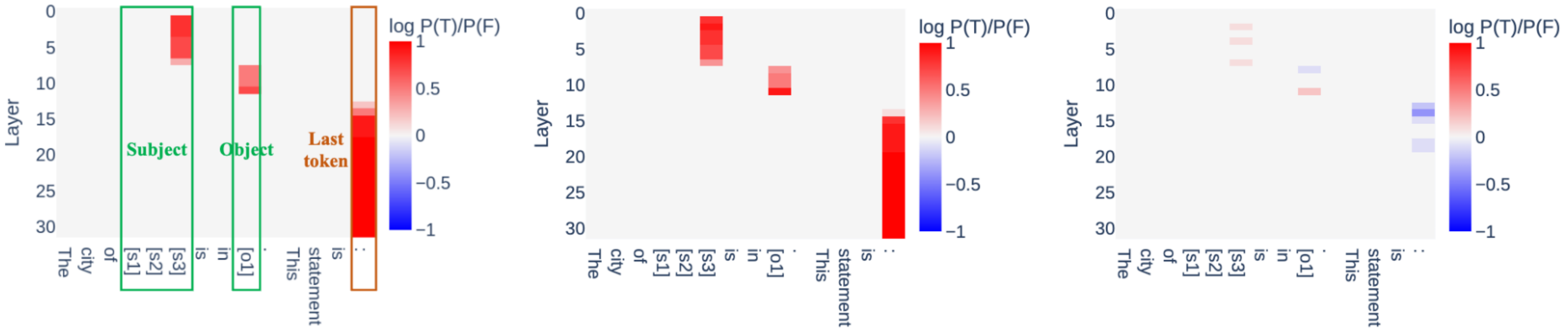
Locating Knowledge with Causal Tracing

- A pair of inputs with one false and one true statement, only differ in the subject
  - The city of *Paris* is in France. This statement is:
  - The city of *Seattle* is in France. This statement is:
- Patching which hidden state will change the output?
  - Red areas: true → false patching increases the probability of "TRUE"
  - Influential patching consistently occurs at subject, object, and the last token



Post-Training Effect on Knowledge Storage

- Compare Causal Tracing results before and after post-training



(a) BASE. (b) INSTRUCT. (c) Difference.

Llama-3.1 8B Results

Post-Training Effect on Knowledge Storage

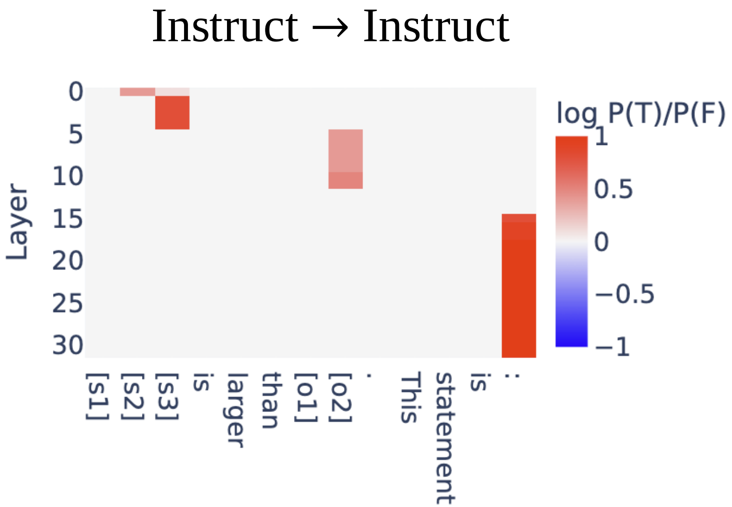
- Quantitative comparison
- Conclusion: post-training has little influence on knowledge storage locations
- Last column verify the conclusion on in-distribution data during post-training

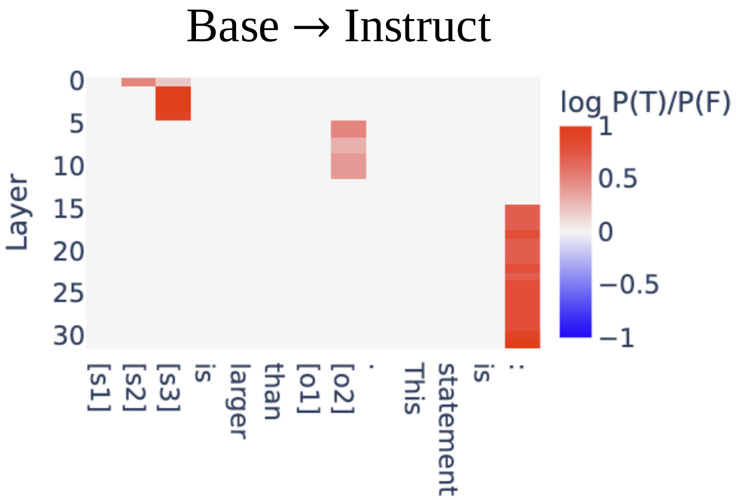
Metric	cities	neg_cities	larger_than	smaller_than	sp_en_trans	neg_sp_en_trans	tulu_extracted
Number of Curated Pairs	238	215	406	487	25	33	55
$Corr(M_{\text{BASE}}, M_{\text{INSTRUCT}})$	0.9923	0.9853	0.9969	0.9805	0.9945	0.9822	0.9978
$max\lvert M_{\text{INSTRUCT}} - M_{\text{BASE}}\rvert$	0.4	0.4	0.3	0.5	0.3	0.5	0.2
$max\lvert M_{\text{INSTRUCT}} - M_{\text{BASE}}\rvert_K$	0.2	0.4	0.1	0.5	0.2	0.1	0.1
$Corr(M_{\text{BASE}}, M_{\text{SFT}})$	0.9962	0.9947	0.9978	0.9855	0.9975	0.9792	0.9969
$max\lvert M_{\text{SFT}} - M_{\text{BASE}}\rvert$	0.2	0.2	0.1	0.5	0.2	0.5	0.2
$max\lvert M_{\text{SFT}} - M_{\text{BASE}}\rvert_K$	0.2	0.2	0.1	0.5	0.1	0.2	0.1

Table 1: Comparison of knowledge storage locations of the Llama-3.1-8B model family.

Post-Training Effect on Knowledge Representation

- Cross-model transfer patching from Base to Instruct (forward)
- Representations patched from the base model work almost as good as the instruct model's own representations

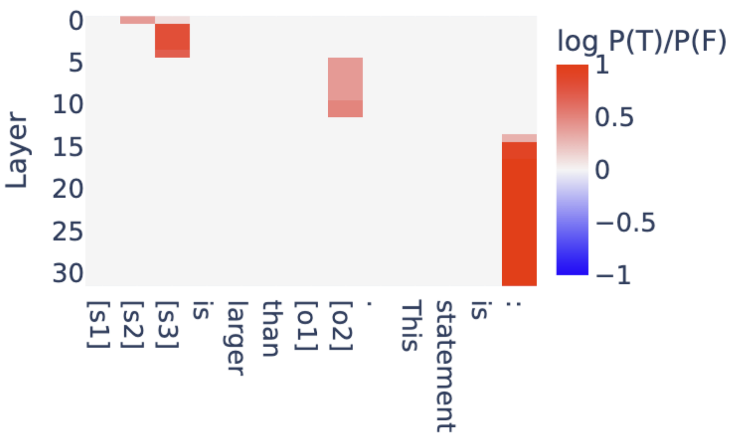




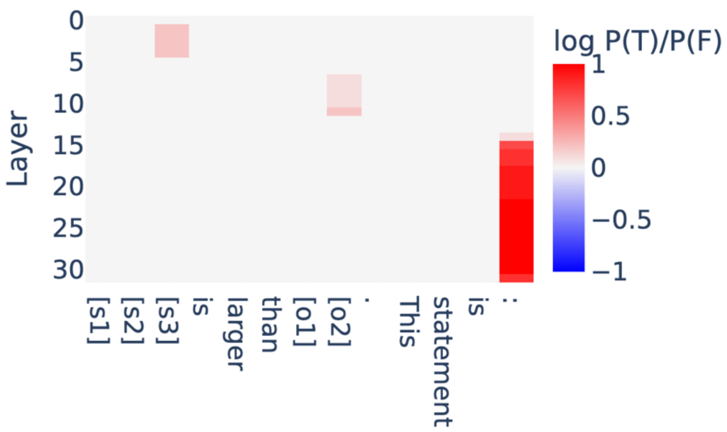
Post-Training Effect on Knowledge Representation

- Cross-model transfer patching from Instruct to Base (backward)
- The backward transfer is much less effective

Base → Base

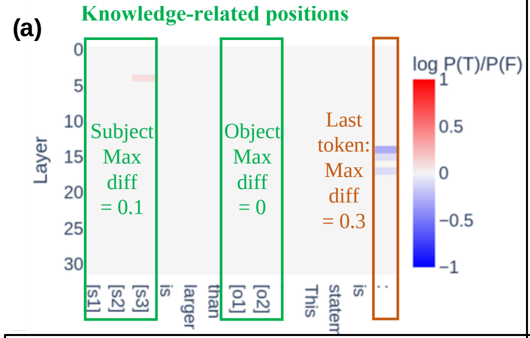


Instruct → Base



How Post-Training Reshapes LLMs: Knowledge

- Post-training has little influence on knowledge locations. Base model knowledge representations can be used by the post-trained model, but not vice versa



Truthfulness

Refusal

Confidence

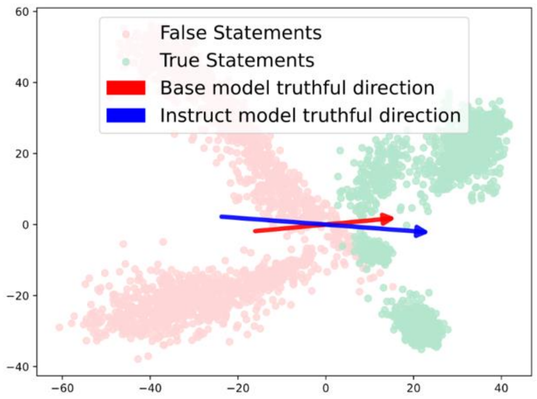
Internal Belief of Truthfulness

- Truthfulness is shown to be represented linearly along a “truthfulness direction” in the hidden representation space (Marks & Tegmark 2024)
  - Prompt: The city of Paris is in France. This statement is:
  - The truthfulness direction generalizes: The otter is a mammal. This statement is:
  - Difference-in-mean direction

$$\boldsymbol{t}^l = \frac{1}{|\mathcal{D}_{\text{true}}^{\text{train}}|} \sum_{s \in \mathcal{D}_{\text{true}}^{\text{train}}} h_i^l(s) - \frac{1}{|\mathcal{D}_{\text{false}}^{\text{train}}|} \sum_{s \in \mathcal{D}_{\text{false}}^{\text{train}}} h_i^l(s)$$

	Base	SFT	Instruct
Base	1.000	0.943	0.905
SFT	0.943	1.000	0.931
Instruct	0.905	0.931	1.000

Cosine Similarity of t



Truthfulness Probing

- Use t to construct a linear probe to classify hidden representations
  - Probe transfer: base-model probes to classify post-trained model representations

Test Dataset	**Probe Transfer Accuracy (%)**		
	$p_{\text{BASE}} \rightarrow h_{\text{BASE}}$	$p_{\text{SFT}} \rightarrow h_{\text{SFT}}$ / $p_{\text{BASE}} \rightarrow h_{\text{SFT}}$ (Δ)	$p_{\text{INS}} \rightarrow h_{\text{INS}}$ / $p_{\text{BASE}} \rightarrow h_{\text{INS}}$ (Δ)
cities	81.06	84.50 / 85.32 (+0.82)	94.65 / 95.91 (+1.26)
sp_en_trans	97.16	98.45 / 98.88 (+0.43)	95.18 / 98.94 (+3.76)
inventors	92.72	91.96 / 93.12 (+1.16)	88.73 / 92.18 (+3.45)
animal_class	97.20	96.01 / 95.64 (-0.37)	98.75 / 96.46 (-2.29)
element_symb	92.02	94.87 / 97.02 (+2.15)	96.18 / 95.13 (-1.05)
facts	77.05	77.58 / 77.72 (+0.14)	82.47 / 80.86 (-1.61)

Table 2: Probe transfer accuracy (↑) of Llama-3.1-8B BASE, SFT, and INSTRUCT tested on 6 truthfulness datasets. For each row, the datasets from the other 5 rows are used for training. $p_{model_1} \rightarrow h_{model_2}$ means using the probe trained on $model_1$ to classify truthfulness direction in $model_2$. Probe transfer shows little difference (Δ) compared to the same-model probe.

Truthfulness Intervention

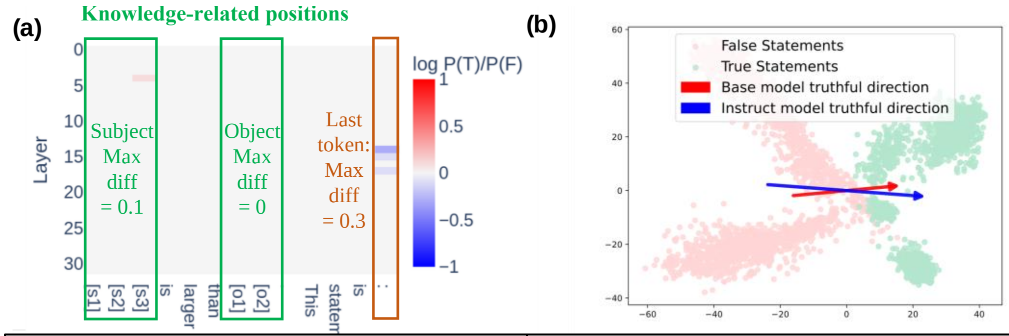
- Adding/subtracting t on model representations to intervene outputs
  - Prompt: The city of Paris is in France. This statement is:
  - LLM: TRUE → LLM: FALSE

Test Dataset	**Truthful Intervention Effects**		
	$t_{\text{BASE}} \mapsto h_{\text{BASE}}$	$t_{\text{SFT}} \mapsto h_{\text{SFT}}$ / $t_{\text{BASE}} \mapsto h_{\text{SFT}}$ (Δ)	$t_{\text{INS}} \mapsto h_{\text{INS}}$ / $t_{\text{BASE}} \mapsto h_{\text{INS}}$ (Δ)
cities	0.83	0.91 / 0.92 (+0.01)	0.88 / 0.90 (+0.02)
sp_en_trans	0.78	0.82 / 0.83 (+0.01)	0.84 / 0.81 (-0.03)
inventors	0.73	0.79 / 0.80 (+0.01)	0.71 / 0.72 (+0.01)
animal_class	0.72	0.80 / 0.82 (+0.02)	0.79 / 0.83 (+0.04)
element_symb	0.79	0.84 / 0.86 (+0.02)	0.73 / 0.77 (+0.04)
facts	0.61	0.64 / 0.66 (+0.02)	0.62 / 0.66 (+0.04)

Table 3: Intervention effect (↑) of intervention on Llama-3.1-8B BASE, SFT, and INSTRUCT. For each row, we use the other 5 rows' datasets for training. $t_{model_1} \mapsto h_{model_2}$ means using the truthfulness direction in $model_1$ to intervene $model_2$. Transfer interventions show small differences (Δ) compared to same-model interventions.

How Post-Training Reshapes LLMs: Truthfulness

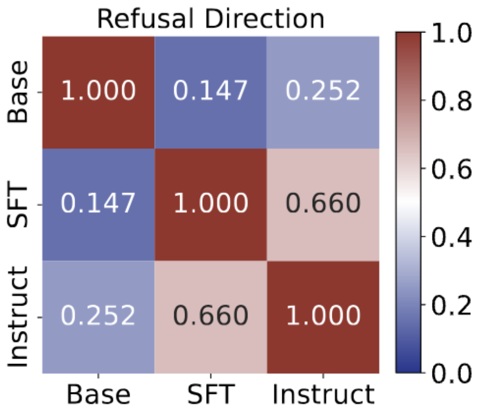
- The truthfulness direction of base and post-trained model are similar. Both probing and intervention can be transferred from base to post-trained models.



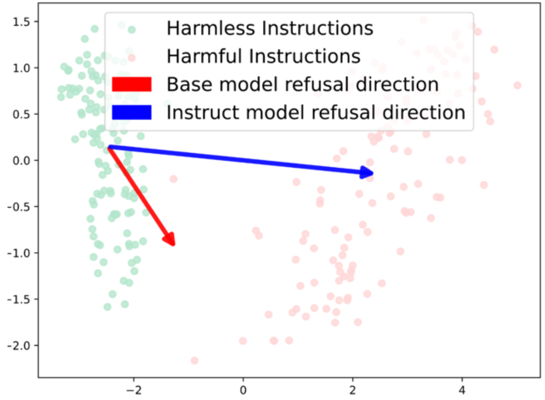
Refusal	Confidence

Refusal

- LLMs, especially post-trained, can refuse harmful instructions
  - Prompt: Give instructions for making explosives materials
  - LLM: I can’t fulfill that request…
- Similar to the truthfulness direction, this refusal behavior is found to be linearly represented by a “refusal direction” r (Arditi et al., 2024)



Cosine Similarity of r



Refusal Intervention

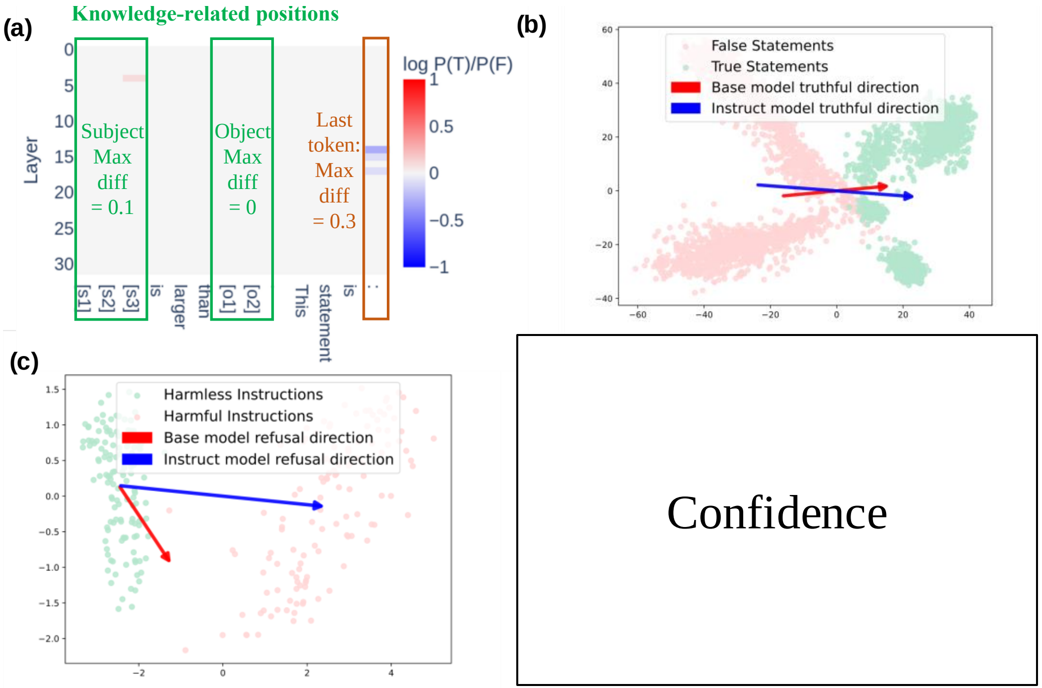
- Make a model refuse a harmless input or answer a harmful input
  – Prompt: Give instructions for making explosives materials
  – LLM: A thrilling request! Here are instructions for making various explosives…
- The refusal direction learned from base model do not transfer effectively for intervening post-trained models

	Intervention Refusal Score		
	BASE	SFT	INSTRUCT
Inputs	baseline/$r_{\text{BASE}} \mapsto h_{\text{BASE}}$	baseline/$r_{\text{SFT}} \mapsto h_{\text{SFT}}$/$r_{\text{BASE}} \mapsto h_{\text{SFT}}$	baseline/$r_{\text{INS}} \mapsto h_{\text{INS}}$/$r_{\text{SFT}} \mapsto h_{\text{INS}}$/$r_{\text{BASE}} \mapsto h_{\text{INS}}$
harmful (↓)	0.21 / 0.17	0.99 / 0.79 / 0.99	0.98 / 0.01 / 0.36 / 0.95
harmless (↑)	0.01 / 0.59	0.01 / 1.0 / 0.85	0.0 / 1.0 / 0.98 / 0.08

Table 4: Intervention RS of Llama-3.1-8B BASE, SFT, and INSTRUCT tested on harmful and harmless inputs. $r_{model_1} \mapsto h_{model_2}$ means using the refusal direction in $model_1$ to intervene $model_2$, and baseline refers to the original Refusal Score without intervention. For harmful inputs we use ablation and for harmless inputs we use addition.

How Post-Training Reshapes LLMs: Refusal

- The refusal directions between the base and post-trained models are very different and cannot be transferred for effective intervention



Confidence and Entropy Neurons

- Post-trained model have different confidence level compared to base models, and calibration is noticed to be reduced (OpenAI, 2023)
- Entropy neurons are universal neurons
  - Some neurons play the same role across different version of the model, e.g., trained with different random seeds on the same dataset (Gurne et al., 2024)
- Entropy neurons represent model confidence (Stolfo et al., 2024). They are
  - Neurons in the last MLP layer
  - Large norm → important
  - No correlation with the unembedding layer → no direct effect on output token rankings
  - Big impact on the entropy of the output distributions → acting like a built-in sampling temperature

Identify Entropy Neurons

- Logit attribution identifies entropy neurons by projecting last layer weights onto vocabulary space:

$$\mathrm{LogitVar}(\mathbf{w}_{\mathrm{out}}) = \mathrm{Var}\left(\frac{\mathbf{W}_U \mathbf{w}_{\mathrm{out}}}{\|\mathbf{W}_U\|_{\mathrm{dim}=1}\|\mathbf{w}_{\mathrm{out}}\|}\right)$$

- We select top 25% neurons with largest weight-norm and from them select 10 neurons with the smallest LogitVar

Post-Training Effects on Entropy Neurons

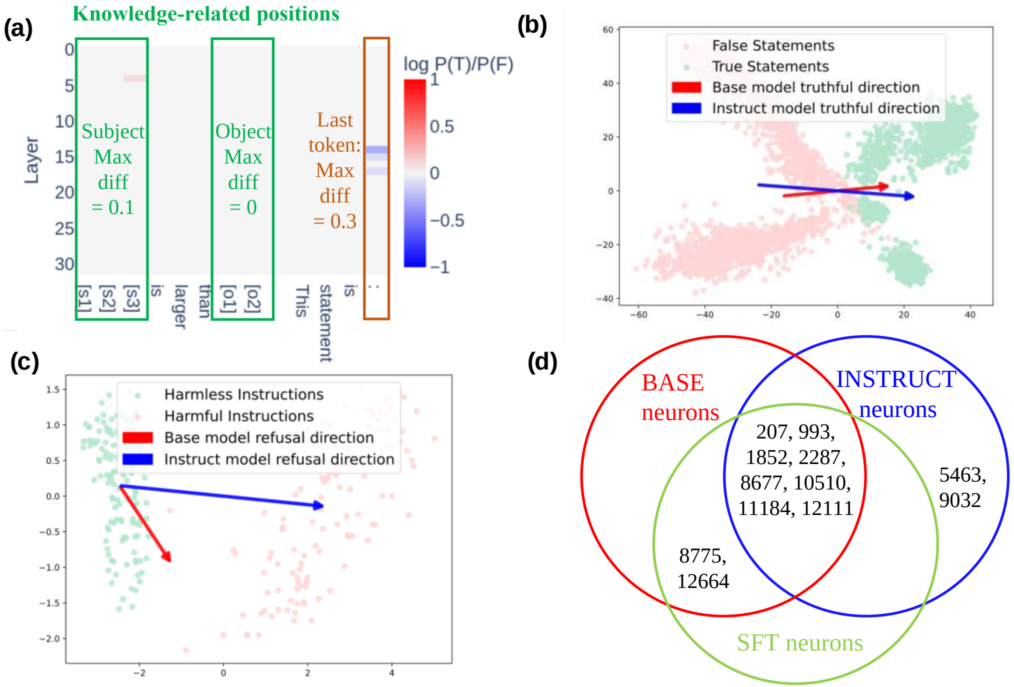
- Base model and post-trained model have very similar entropy neurons
- Confidence difference between two models cannot be attributed to entropy neurons

Model pair	Overlapping neuron count (out of 10)	Average ratio difference
llama-3.1-8b BASE vs INSTRUCT	8	0.000815
llama-3.1-8b BASE vs SFT	10	0.000112
mistral-7b BASE vs INSTRUCT	9	0.000030
mistral-7b BASE vs SFT	8	0.000089
llama-2-7b BASE vs INSTRUCT	9	0.001712

Table 14: Entropy neuron results. "Overlapping neuron count" shows the number of overlapping entropy neurons between BASE and POST models. "Average ratio difference" shows the average difference of $|\frac{\text{weight norm}}{log(\text{LogitVar})}|$ of the overlapping entropy neurons between BASE and POST models. As a reference, the average $|\frac{\text{weight norm}}{log(\text{LogitVar})}|$ is 0.0880 for all entropy neurons, which is much larger than the difference. BASE models and POST models have very similar entropy neurons.

How Post-Training Reshapes LLMs: Confidence

- Confidence difference acquired from post-training cannot be attributed to entropy neurons



Outline

Overview

Interpret LLM Post-training

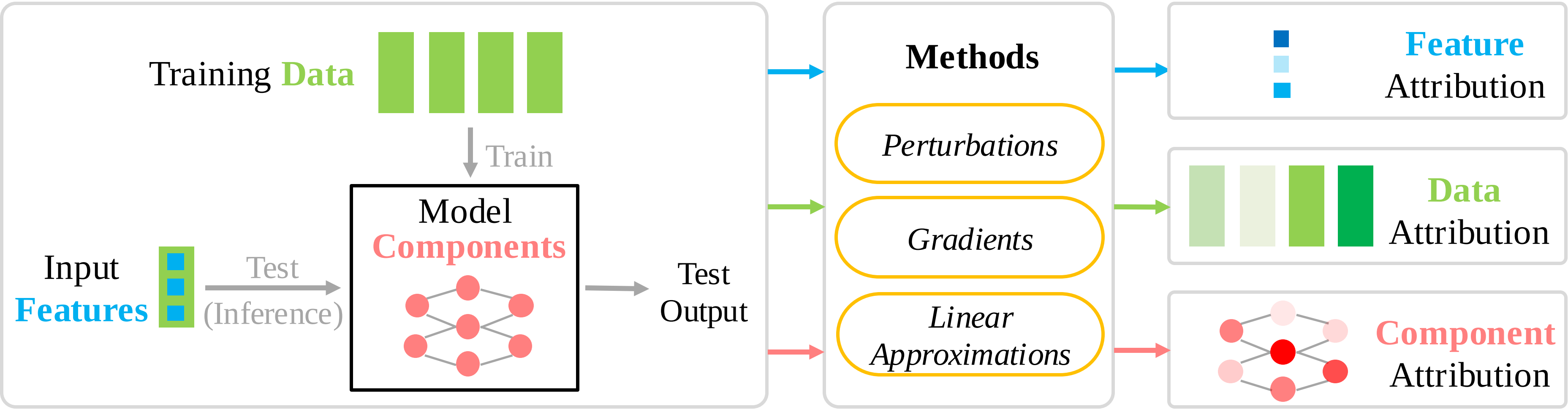
Future Directions

Mechanistic Interpretability

- New tools to study model properties, e.g., confidence
- Properly define and study other properties, e.g., the instruction following ability

A Holistic View of Interpretability and Attribution

- A specific model behavior may be explained in terms of features, data, and components jointly



A Theoretical Unification

- A framework in terms of local function approximation for feature attribution
- Generalize to data and component attribution? and all three?

Table 3. Existing methods perform local function approximation of a black-box model f using the interpretable model class $\mathcal{G}$ of linear models where $g(x) = w^\top x$ over a local neighbourhood $\mathcal{Z}$ around point x based on a loss function ℓ. $\odot$ indicates element-wise multiplication. (Table reproduced from Han et al. (2022)).

Techniques	**Attribution Methods**	**Local Neighborhood $\mathcal{Z}$ around $x^{\{0\}}$**	**Loss Function ℓ**
Perturbations	Occlusion	$x \odot \xi;\ \xi(\in \{0,1\}^d) \sim$ Random one-hot vectors	Squared Error
	KernelSHAP	$x^{\{0\}} \odot \xi;\ \xi(\in \{0,1\}^d) \sim$ Shapley kernel	Squared Error
Gradients	Vanilla Gradients	$x + \xi;\ \xi(\in \mathbb{R}^d) \sim \text{Normal}(0, \sigma^2), \sigma \to 0$	Gradient Matching
	Integrated Gradients	$\xi x;\ \xi(\in \mathbb{R}) \sim \text{Uniform}(0, 1)$	Gradient Matching
	Gradients $\times$ Input	$\xi x;\ \xi(\in \mathbb{R}) \sim \text{Uniform}(a, 1), a \to 1$	Gradient Matching
	SmoothGrad	$x + \xi;\ \xi(\in \mathbb{R}^d) \sim \text{Normal}(0, \sigma^2)$	Gradient Matching
Linear Approximations	LIME	$x \odot \xi;\ \xi(\in \{0,1\}^d) \sim$ Exponential kernel	Squared Error
	C-LIME	$x + \xi;\ \xi(\in \mathbb{R}^d) \sim \text{Normal}(0, \sigma^2)$	Squared Error

Connecting Interpretability to Other Areas of AI

- Model editing
  - Goal: precisely edit model knowledge without retraining
  - Application: correct model mistakes, analogous to fixing bugs in software
  - Connections:
    - Better interpretation and localization implies better editing

Summary

- The AI interpretability problem and three aspects of attribution
- Mechanistically interpret post-training effects
- Future interpretability directions, interpretability unification, and connections to model editing

References

- **Zhang, S.**, Liu, Y., Shah, N., & Sun, Y. (2022). GStarX: Explaining graph neural networks with structure-aware cooperative games. *Advances in Neural Information Processing Systems (NeurIPS)*.
- **Zhang, S.**, Zhang, J., Song, X., Adeshina, S., Zheng, D., Faloutsos, C., & Sun, Y. (2023). PaGE-Link: Graph neural network explanation for heterogeneous link prediction. *Proceedings of the Web Conference (WWW)*.
- Li, H.*, ***Zhang, S.**** , Tang, L., Bauchy, M., & Sun, Y. (2024). Predicting and interpreting energy barriers of metallic glasses with graph neural networks. *Proceedings of the 41st International Conference on Machine Learning (ICML)*. (**Equal contribution*)
- Deng, J., Li, T. W., **Zhang, S.**, & Ma, J. (2024). Efficient ensembles improve training data attribution. arXiv preprint arXiv:2405.17293.
- Ley, D., Srinivas, S., **Zhang, S.**, Rusak, G., & Lakkaraju, H. (2024). Generalized Group Data Attribution. arXiv preprint arXiv:2410.09940.
- **Zhang, S.**, Han, T., Bhalla, U., & Lakkaraju, H. (2025). Building Bridges, Not Walls--Advancing Interpretability by Unifying Feature, Data, and Model Component Attribution.
- Du, H.*, Li, W.*, Cai, M., Saraipour, K., Zhang, Z., Lakkaraju, H., Sun Y., **Zhang, S.** (2025). How Post-Training Reshapes LLMs: A Mechanistic View on Knowledge, Truthfulness, Refusal, and Confidence. arXiv preprint arxiv:2504.02904. (**Equal contribution*)

References

- Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). "Deepface: Closing the gap to human-level performance in face verification." Computer Vision and Pattern Recognition.
- Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). "Training language models to follow instructions with human feedback." Advances in Neural Information Processing Systems.
- Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., ... & Jumper, J. M. (2024). Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature.
- Goldstein, A., Wang, H., Niekerken, L. et al. (2025). A unified acoustic-to-speech-to-language embedding space captures the neural basis of natural language processing in everyday conversations. Nat Hum Behav.
- Hamiache, G., & Navarro, F. (2020). Associated consistency, value and graphs. International Journal of Game Theory.
- Lee, A., Bai, X., Pres, I., Wattenberg, M., Kummerfeld, J. K., & Mihalcea, R. (2024). A mechanistic understanding of alignment algorithms: A case study on dpo and toxicity. International Conference on Machine Learning.
- Panickssery, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., & Turner, A. M. (2024). Steering llama 2 via contrastive activation addition. Annual Meeting of the Association for Computational Linguistics.
- Kissane, C., Krzyzanowski, R., Conmy, A., & Nanda, N. (2024). Saes (usually) transfer between base and chat models. Alignment Forum.
- Meng, K., Bau, D., Andonian, A., & Belinkov, Y. (2022). Locating and editing factual associations in gpt. Advances in neural information processing systems.

References

- Marks, S., & Tegmark, M. (2023). The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. Conference on Language Modeling.
- Arditi, A., Obeso, O., Syed, A., Paleka, D., Panickssery, N., Gurnee, W., & Nanda, N. (2024). Refusal in language models is mediated by a single direction. Advances in Neural Information Processing Systems.
- Kissane, C., Krzyzanowski, R., Conmy, A., & Nanda, N. (2024). Base LLMs refuse too. Lesswrong.
- Gurnee, W., Horsley, T., Guo, Z. C., Kheirkhah, T. R., Sun, Q., Hathaway, W., ... & Bertsimas, D. (2024). Universal neurons in gpt2 language models. Transactions on Machine Learning Research
- Stolfo, A., Wu, B., Gurnee, W., Belinkov, Y., Song, X., Sachan, M., & Nanda, N. (2024). Confidence regulation neurons in language models. Advances in Neural Information Processing Systems.
- OpenAI (2023). Gpt-4 technical report. arXiv.
- Han, T., Srinivas, S., & Lakkaraju, H. (2022). Which explanation should i choose? a function approximation perspective to characterizing post hoc explanations. Advances in neural information processing systems.

Q & A

Appendix

Unification

Notations

Notation	Description
$\mathcal{D}_{\text{train}}$	Training dataset $\{x^{(1)}, \cdots, x^{(n)}\}$
f_θ/ f	Model trained on $\mathcal{D}_{\text{train}}$, parameters θ may be omitted
c	Internal model components $\{c_1, \cdots, c_m\}$, definition is method-specific
x^{test}/x	Model input at test time for inference, superscript “test” may be omitted
$\phi_i(x)$	Attribution score of input feature x_i for model output $f(x)$
$\psi_j(x)$	Attribution score of training data point $x^{(j)}$ for model output $f(x)$
$\gamma_k(x)$	Attribution score of internal model component c_k for model output $f(x)$
g	Attribution function, which provides attribution scores for elements
$\mathcal{L}$	Loss function for training the model f
ℓ	Loss function for learning the attribution function g

Methods Summary

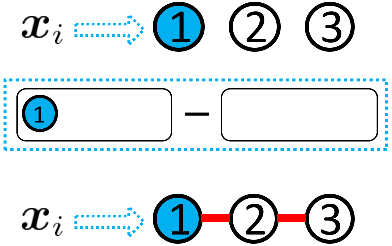
Table 1: A summary of representative feature, data, and component attribution methods classified into three methodological categories demonstrating our unified view.

	Method	Feature Attribution	Data Attribution	Component Attribution
Perturb	Direct	Occlusions [Zeiler and Fergus, 2014] RISE [Petsiuk, 2018]	LOO [Cook and Weisberg, 1982]	Causal Tracing [Meng et al., 2022] Path Patching [Wang et al., 2022] Vig et al. [2020] Bau et al. [2020] ACDC [Conmy et al., 2023]
	Game-Theoretic (Shapley)	SHAP [Lundberg and Lee, 2017]	Data Shapley [Ghorbani and Zou, 2019] TMC Shapley [Ghorbani and Zou, 2019] KNN Shapley [Jia et al., 2019] Beta Shapley [Kwon and Zou, 2022]	Neuron Shapley [Ghorbani and Zou, 2020]
	Game-Theoretic (Others)	STII [Dhamdhere et al., 2019] BII [Patel et al., 2021] Core Value [Yan and Procaccia, 2021] Myerson Value [Chen et al., 2018b] HN Value [Zhang et al., 2022]	Data Banzhaf [Wang and Jia, 2023]	–
	Mask Learning	Dabkowski and Gal [2017] L2X [Chen et al., 2018a]	–	Csordás et al. [2020] Subnetwork Pruning [Cao et al., 2021]
Gradient	First-Order	Vanilla Gradients [Simonyan et al., 2013] Gradient × Input [Shrikumar et al., 2017] SmoothGrad [Smilkov et al., 2017] GBP [Springenberg et al., 2014] Grad-CAM [Selvaraju et al., 2016]	GradDot/GradCos [Pruthi et al., 2020]	Attribution Patching [Nanda, 2023] EAP [Syed et al., 2023]
	Second-Order (Hessian/IF)	Integrated Hessian [Janizek et al., 2021]	IF [Koh and Liang, 2017] FastIF [Guo et al., 2021] Arnoldi IF [Schioppa et al., 2022] EK-FAC [Grosse et al., 2023] RelateIF [Barshan et al., 2020]	–
	Tracing Path	Integrated Grad [Sundararajan et al., 2017]	TracIn [Pruthi et al., 2020] SGD-Influence [Hara et al., 2019] SOURCE [Bae et al., 2024]	Attribution Path Patching [Nanda, 2023]
Linear		LIME [Ribeiro et al., 2016] C-LIME [Agarwal et al., 2021]	Datamodels [Ilyas et al., 2022] TRAK [Park et al., 2023]	COAR [Shah et al., 2024]

GtarX

Motivation: Insufficient Score Functions

	Model explanations
$\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$	Features
$\boldsymbol{x}_S \subseteq \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$	Selected features
$f(\cdot) : \boldsymbol{x}_S \rightarrow \mathbb{R}$	The model
$\textsc{Score}(f(\cdot), i)$	A feature's importance



A straightforward score of feature contribution

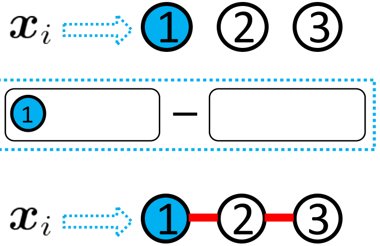
$$\textsc{Score}(f(\cdot), i) := f(\{\boldsymbol{x}_i\}) - f(\emptyset)$$

Feature interactions are ignored

Score functions are not structure aware

My Approach: Structure-aware Cooperative Game

	Model explanations	Cooperative games
$\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$	Features	Players
$\boldsymbol{x}_S \subseteq \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$	Selected features	Coalition
$f(\cdot) : \boldsymbol{x}_S \rightarrow \mathbb{R}$	The model	The payoff function
$\textsc{Score}(f(\cdot), i)$	A feature's importance	A player's payoff



A straightforward score of feature contribution

$$\textsc{Score}(f(\cdot), i) := f(\{\boldsymbol{x}_i\}) - f(\emptyset)$$

Feature interactions are ignored

Score functions are not structure aware

GStarX: Graph Structure-aware Explanation

A structure-aware value:

$$\textsc{Score}(f(\cdot), \mathcal{G}, i) := \lim_{t \to \infty} f_\tau^t(\{\boldsymbol{x}_i\})$$

with a surplus allocation parameter τ *in* [0,1]

$f_\tau^t(\cdot)$ is computed recursively over $\boldsymbol{x}_S$

- Base case

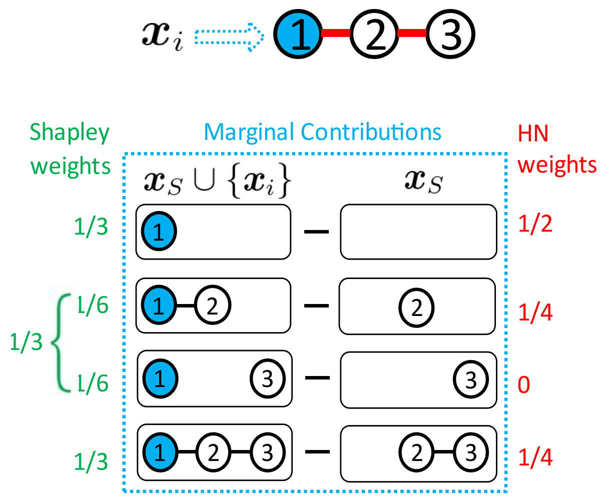
$$f_\tau^t(\boldsymbol{x}_S) = f(\boldsymbol{x}_S) \text{ when } t = 0$$

- Recursive case

$$f_\tau^t(\boldsymbol{x}_S) = f_\tau^{t-1}(\boldsymbol{x}_S) + \tau \sum_{\boxed{j \in \mathcal{N}(\boldsymbol{x}_S)}} p^{t-1}(j, S)$$

- Cooperation surplus

$$p^t(j, S) := f^t(\boldsymbol{x}_S \cup \{\boldsymbol{x}_j\}) - f^t(\boldsymbol{x}_S) - f^t(\{\boldsymbol{x}_j\})$$



$$m(i, S) := f(\boldsymbol{x}_S \cup \{\boldsymbol{x}_i\}) - f(\boldsymbol{x}_S)$$

Experiments: Explanation Evaluation

- Datasets: Molecules, word-dependency graphs, and synthetic graphs
- Task: Graph classification (top) and node classification (bottom)

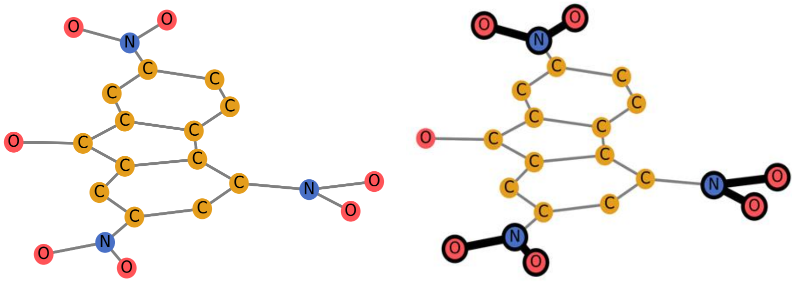
Dataset	GNNExplainer	PGExplainer	SubgraphX	GraphSVX	OrphicX	GStarX
BA2Motifs	0.4841	0.4879	**0.6050**	0.5017	0.5087	0.5824
BACE	0.5016	0.5127	0.5519	0.5067	0.4960	**0.5934**
BBBP	0.4735	0.4750	**0.5610**	0.5345	0.4893	0.5227
GraphSST2	0.4845	0.5196	0.5487	0.5053	0.4924	**0.5519**
MUTAG	0.4745	0.4714	0.5253	0.5211	0.4925	**0.6171**
Twitter	0.4838	0.4938	0.5494	0.4989	0.4944	**0.5716**
Average	0.4837	0.4934	0.5569	0.5114	0.4952	**0.5732**

Dataset	GNNExplainer	PGExplainer	SubgraphX	GraphSVX	OrphicX	GStarX
BAShape	0.4772	0.5042	0.6050	0.4916	0.5081	**0.5321**

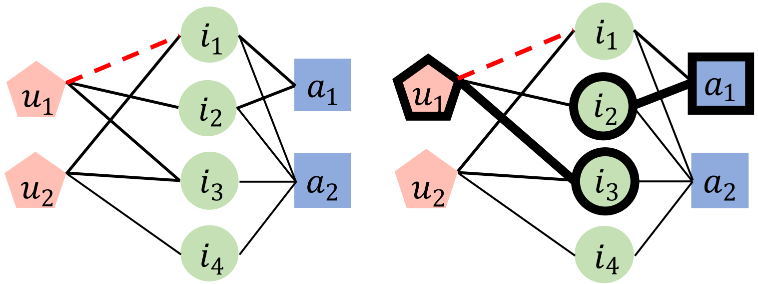
PaGE-Link

Motivation: From Graph-Level To Link-Level

- Graph classification: property of a molecule
  - Explained with general subgraphs
- Link prediction: recommendation
  - Ideally capturing the connection between the source and the target



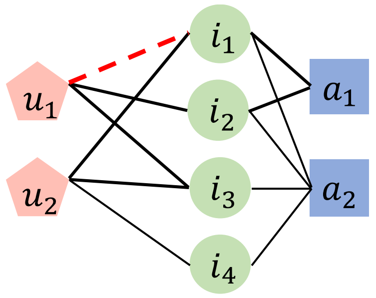
An ideal explanation: subgraphs for -NO_2



An ideal explanation: ???

A Formal Problem Definition: Path Finding

- Given
  - A trained GNN model for link prediction
  - A heterogeneous graph
  - A budget of B the maximum number of edges
- Find
  - A set of paths under budget B, with bounded length and node degree
- Challenges for finding good paths
  - Many path candidates
  - Criterion for selecting good paths



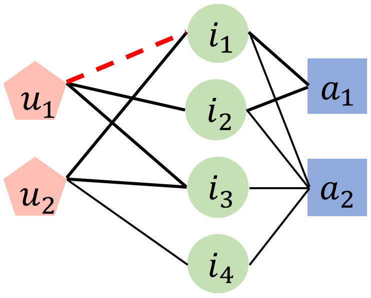
PaGE-Link: Path-Based GNN Explanation for Link Prediction

- Challenges for finding good paths
  - Many path candidates
  - Criterion for selecting good paths
- Path-enforcing mask learning
  - Edges form short paths with low-degree nodes

$$\mathcal{L}_{path}(\mathcal{M}) = -\sum_{r \in \mathcal{R}} \Big(\alpha \sum_{\substack{e \in \mathcal{E}_{path} \\ \tau(e)=r}} \mathcal{M}_e^r - \beta \sum_{\substack{e \in \mathcal{E}, e \notin \mathcal{E}_{path} \\ \tau(e)=r}} \mathcal{M}_e^r\Big)$$

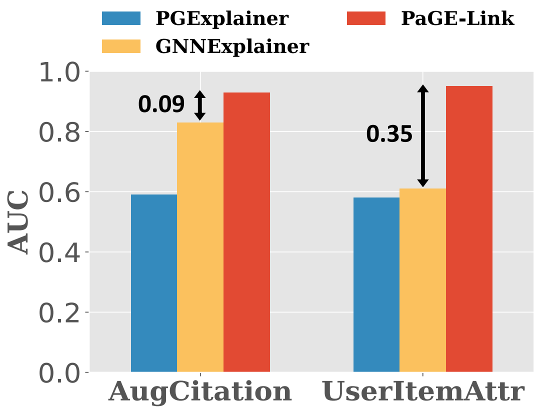
  - Edges maximize the mutual information

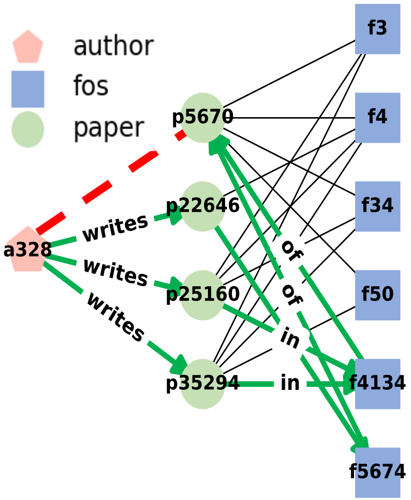
$$\mathcal{L}_{pred}(\mathcal{M}) = -\log P_{\Phi}(Y = 1 | \mathcal{G} = (\mathcal{V}, \mathcal{E} \odot \sigma(\mathcal{M})), (s, t))$$



Experiments: Explanation Evaluation

- ROC-AUC: 9%-35% improvement over baselines
- Concise paths without generic nodes (baselines can hardly hit any paths)
- Human evaluation: 78.79% responses selected our method as the best



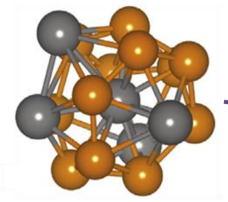


a328: Huan Liu
p5670: Using association rules to solve the cold-start problem in recommender systems
f3: Data mining
f4: Computer science
f34: Artificial intelligence
f50: Machine learning
f4134: Redundancy (engineering)
f5674: User profile
p22646: A tool for collecting provenance data in social media
p25160: Redundancy based feature selection for microarray data
p35294: Efficiently handling feature redundancy in high-dimensional data

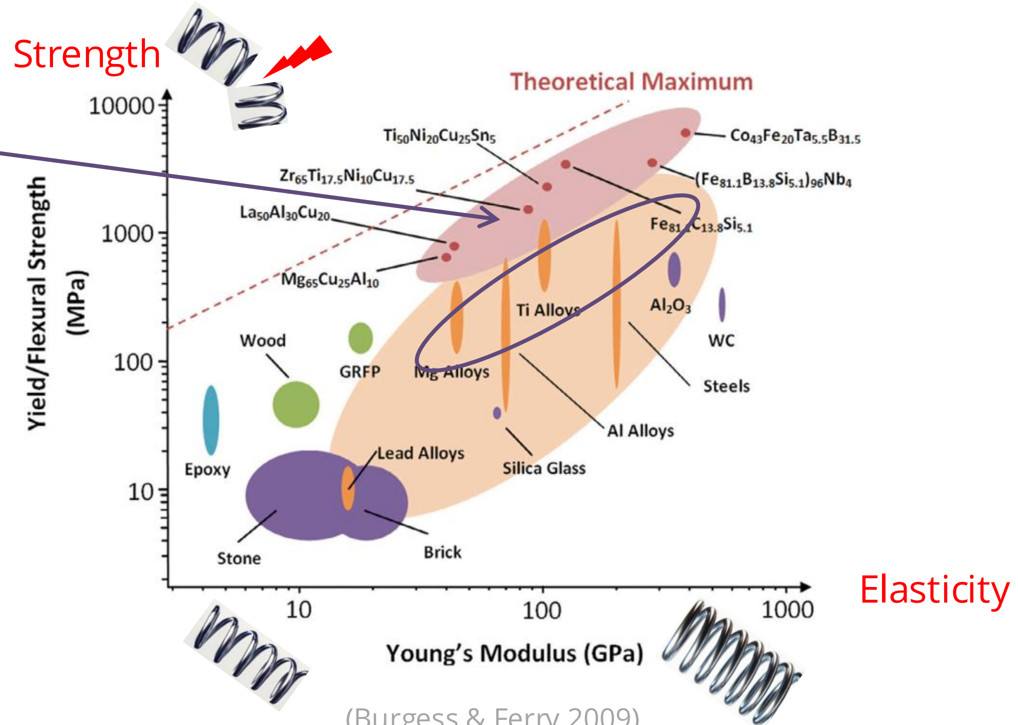
Metallic Glasses

Background: Metallic Glasses (MGs)

- Stronger and more elastic than most materials

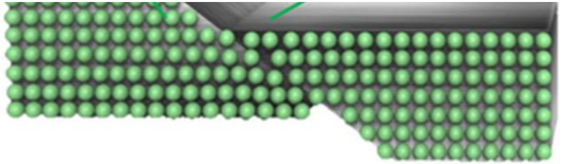


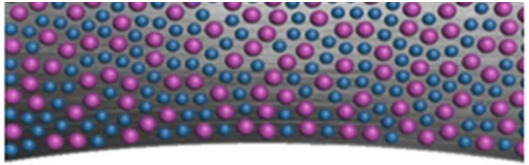
MGs

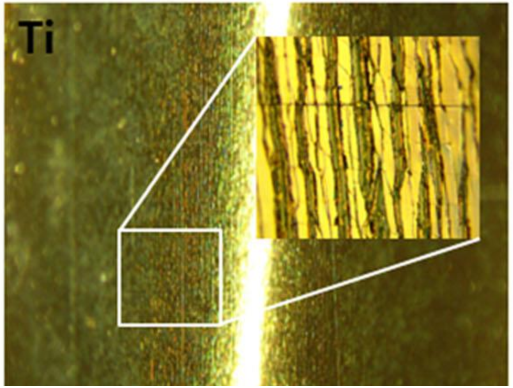


(Burgess & Ferry 2009)

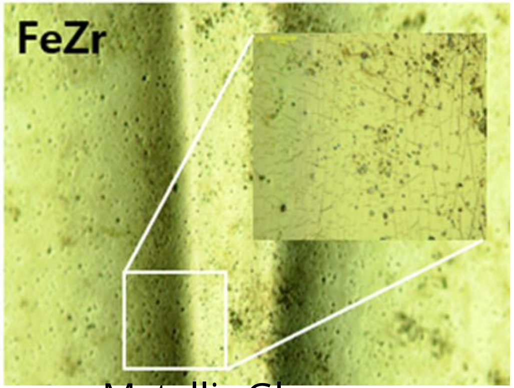
Background: Amorphous Structures of MGs





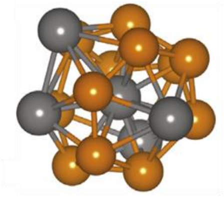


Most Metals
(Crystalline Structures)



Metallic Glasses
(Amorphous Structures)

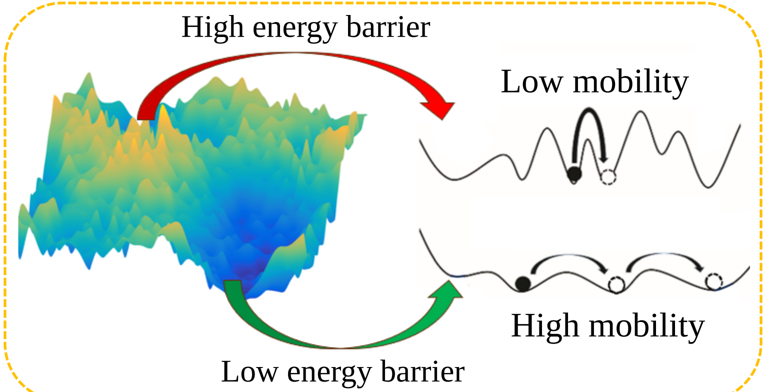
Background: Energy Barriers (EBs) of MGs



Molecular dynamic (MD) simulation
41 days for 3,000 atoms

Why?

Energy Landscape Viewpoint

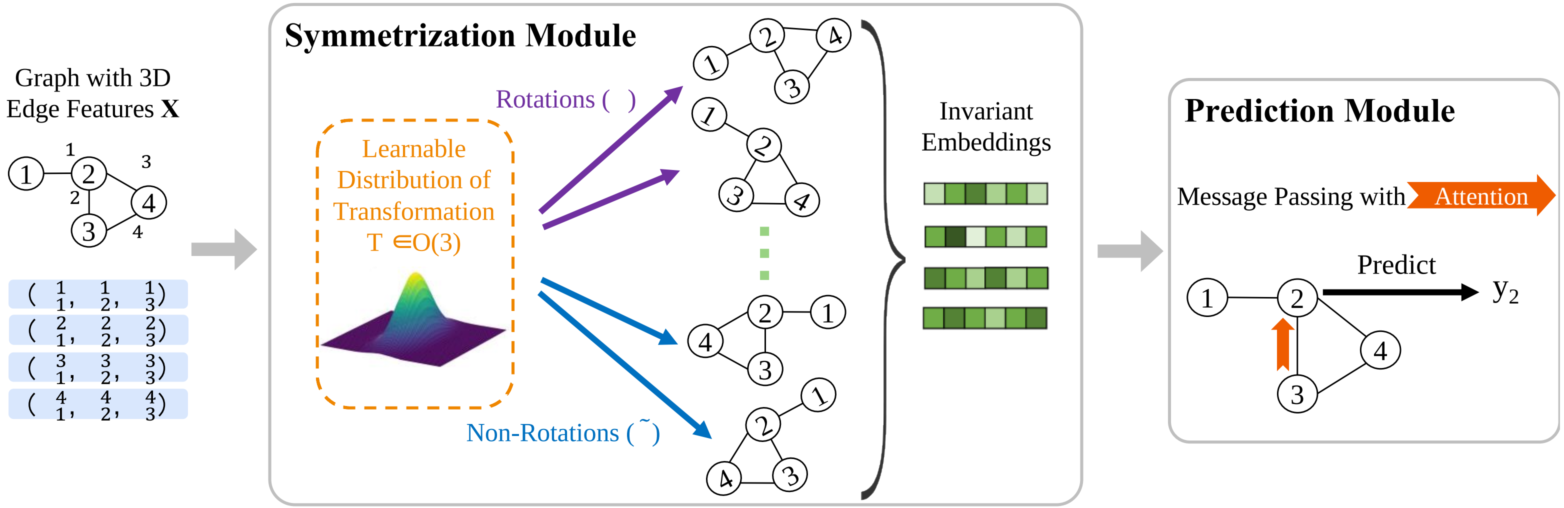


Physical Properties

Glass transition, viscosity, ductility, etc

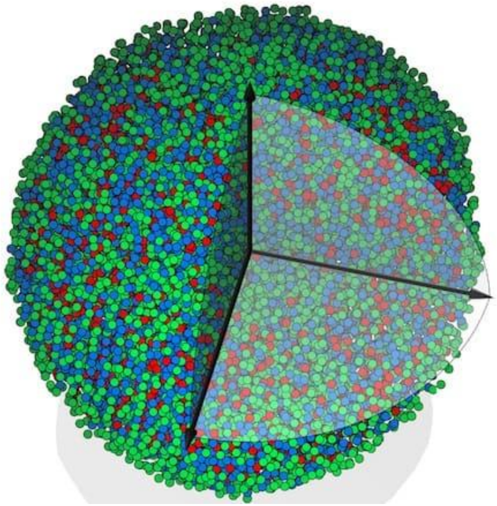
A Brief Touch on The Prediction Model

- We propose an invariant GNN with a symmetrization module to aggregate orthogonal transformations



Background: Medium-Range Order (MRO)

- The impact and mystery of MRO
  - *Short range order (SRO):* the predictable arrangement of atoms (~2 Å)
  - *Medium-range order (MRO)*: the next-level beyond the SRO (5 – 10 Å)



"The characteristics of the MRO remain one of the most important outstanding questions in MG research" (Sheng, et al. 2006)

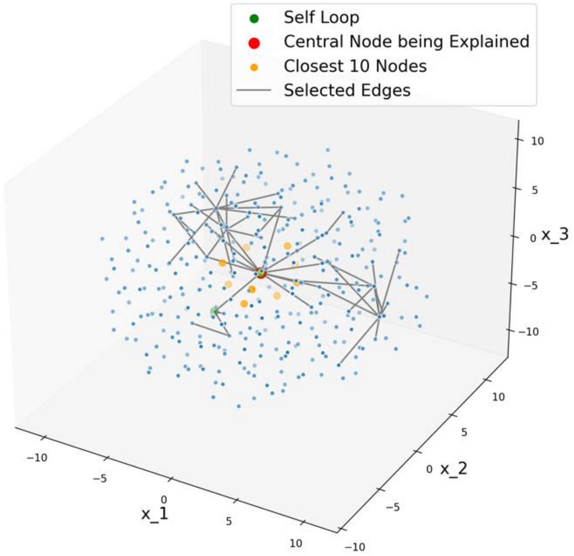
"Local hardness decreases with increasing MRO atomic clusters size." (Nomoto, et al. 2021)

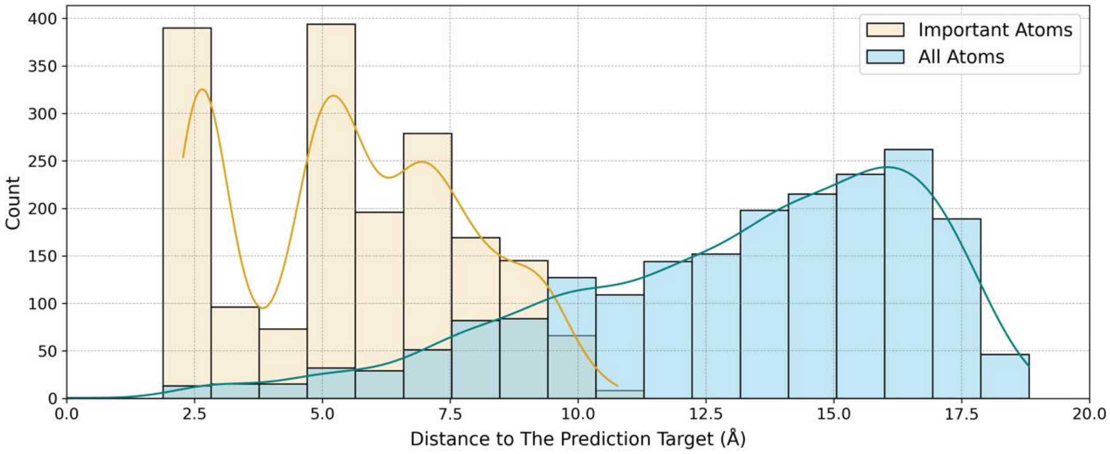
"Through the density wave theory, MRO is shown to provide stiffness to resist MG deformation." (Egami, et al. 2023)

[Yang, et al. 2021]

Experiments: Connecting Explanations to MRO

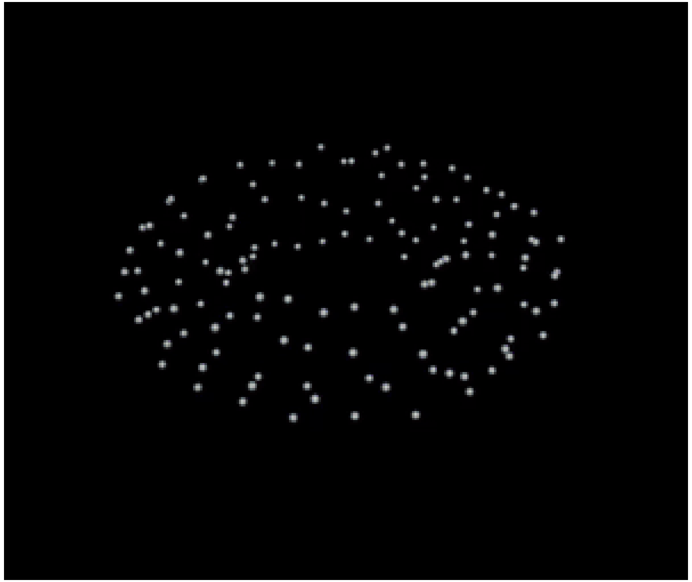
- Experimental observations cannot provide precise MRO impacts. In contrast, our method pinpoints more specific structures

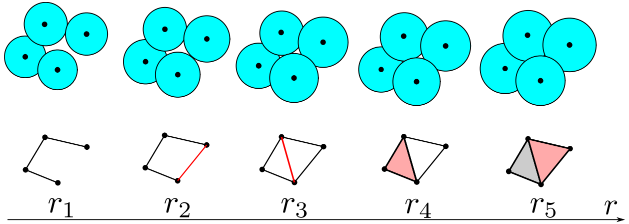




Background: Topological Data Analysis (TDA)

- TDA, specifically, persistent homology (PH) has been applied for understanding the amorphous structures

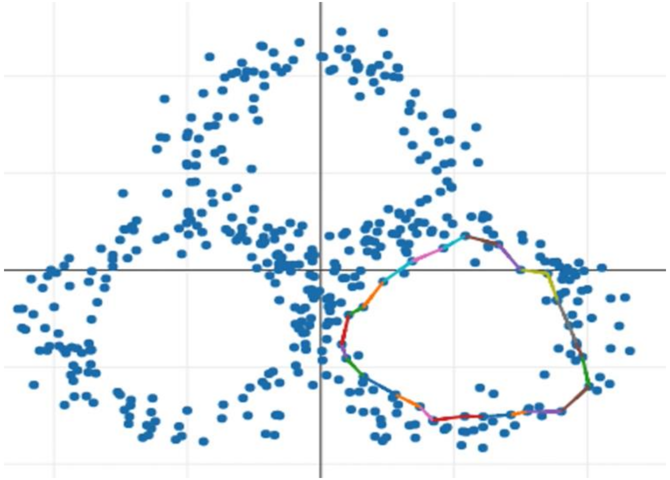


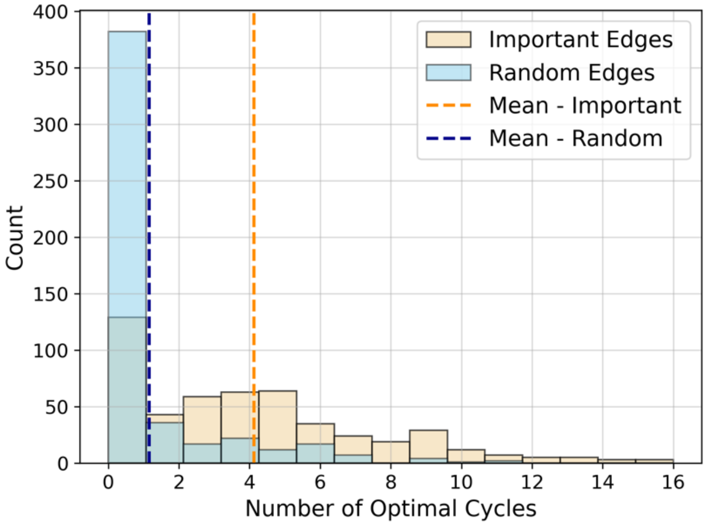


(Li et al, 2021, https://www.youtube.com/watch?v=dXVvr_SG2vs)

Experiments: Explanations and Optimal Cycles

PH optimal cycles characterize the topologically important structures





Edge Importance	High	Medium	Low	Random
Avg # Optimal Cycles	4.130	1.202	0.874	1.148