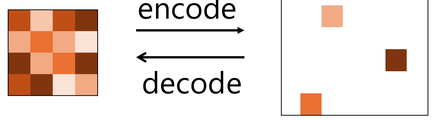
Evaluation of Sparse Autoencoders through the Representation of Polysemous Words

Gouki Minegishi, Hiroki Furuta, Yusuke Iwasawa, Yutaka Matsuo

Introduction and Motivation

Sparse Autoencoders:
Mapping the polysemantic activation of LLM into sparse features

LLM Activations
- ✓ Dense Activations
- ✓ Uninterpretable
- ✓ Polysemantic



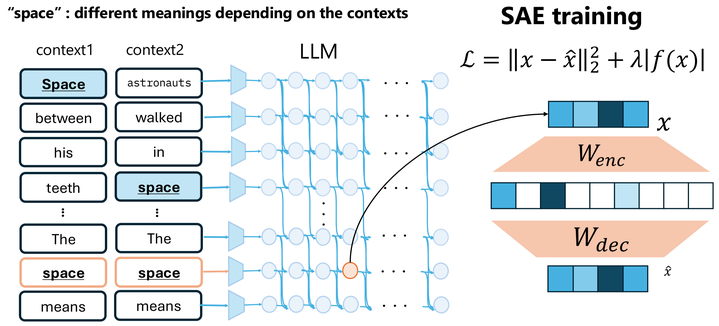
SAE features
- ✓ Sparse Activations
- ✓ Interpretable
- ✓ Monosemantic

However,
how to effectively **evaluate SAE features** remains undefined.

Method: PS-Eval

PS-Eval:
evaluates if SAE features activate correctly for ***polysemous words*** [1] using a **confusion matrix**

Prompt : `{context}.` ***The*** `{target word}` ***means***



Confusion matrix

	Mono-context	Poly-context
Same Max Activated Feature	Same meaning, same feature (**T**rue **P**ositive)	Different meaning, same feature (**F**alse **P**ositive)
Different Max Activated Feature	Same meaning, different feature (**F**alse **N**egative)	Different meaning, different feature (**T**rue **N**egative)

Experiment

Do SAE features reflect word meanings?

Logit Lens:
Use the LLM's unembedding matrix to extract the SAE feature logit

$$\text{logit}_1 = W_U a^1_{\max}, \quad \text{logit}_2 = W_U a^2_{\max}$$

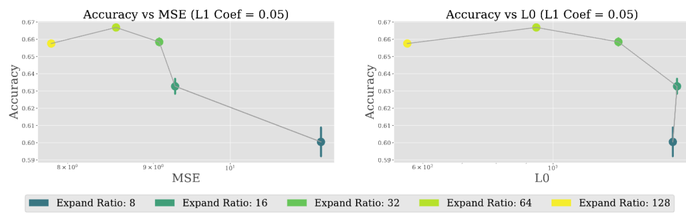
Context 1: The astronauts walked in outer "space".
Context 2: The "space" between his teeth.

`poly-context` (**space**)

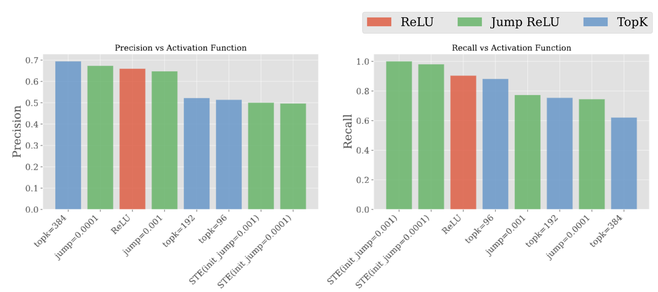
rank	Context 1 tokens	Context 1 logits	Context 2 tokens	Context 2 logits
1	flight	1.224	Layout	0.894
2	plane	0.957	occupied	0.882
3	**shuttle**	0.938	spaces	0.853
4	gravity	0.937	vacated	0.846
5	craft	0.920	space	0.825
6	Engineers	0.876	**shuttle**	0.799
7	planes	0.869	occupancy	0.798

Which SAEs best disentangle polysemy?

Wider SAE is Better



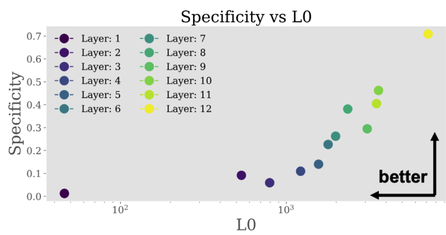
Activation function



Where in LLMs is polysemy disentangled?

Polysemy is Captured in Deeper Layer

$$\text{Specificity} = \frac{TN}{TN+FP}$$



Polysemy is Captured in Attention