SELECTED TOPICS IN DATA CODE PRESENTATION

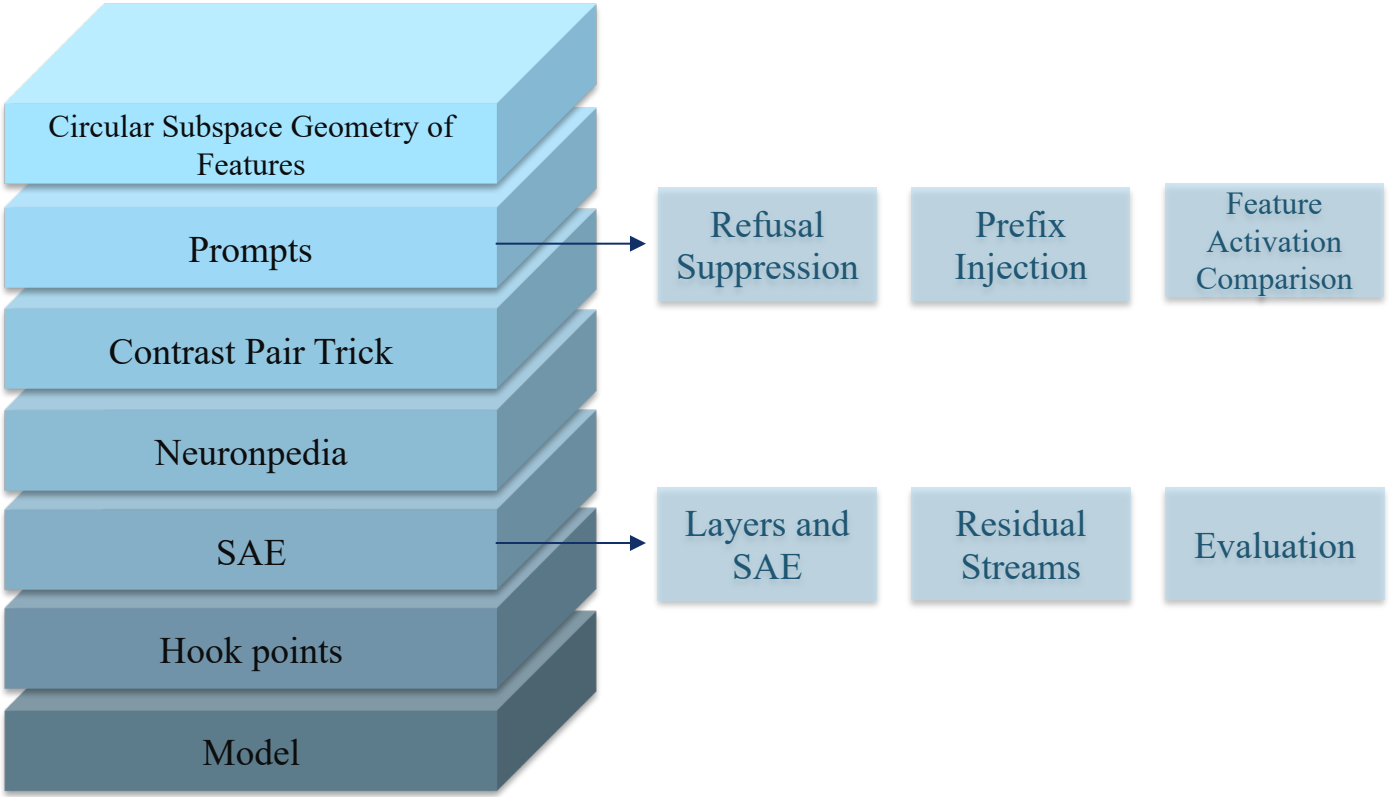
NLP38 Backdoors in LLMs - SAEs

Moujan Mirjalili
Supervisor: Dominik Meier, Jan Philip Wahle

My Task

- Understand an existing codebase
- Run clean and triggered prompt
- Compare trends in feature activations between the prompts
- Interpret features (Neuronpedia, Visualization)
- Identify patterns or features that consistently spike when the trigger is present (if there exists one)

Concepts


Circular Subspace Geometry of Features
Prompts
Contrast Pair Trick
Neuronpedia
SAE
Hook points
Model
Refusal Suppression
Prefix Injection
Feature Activation Comparison
Layers and SAE
Residual Streams
Evaluation

Model

- Base model:
  - gpt2-small with 12 layers and ~124M parameters
  - Loaded via HookedSAETransformer from the sae_lens library
- Hook Point: SAE attached to residual stream at blocks.7.hook_resid_pre.
- SAE source: Joseph Bloom's pretrained SAE (gpt2-small-res-jb release)
- Library stack:
  - sae_lens for loading, running, and visualizing SAEs
  - transformer_lens for model control and tokenization
  - NeelNanda/pile-10k for input data

HookedTransformers & Hook Points

- HookedSAETransformer : A wrapper that integrates SAEs with Transformer model internal layers.
- Purpose:
  - Provides fine-grained control over internal model activations by allowing hooks into specific layers.
  - Observe or edit internal activation at specific points.
- Workflow:
  - During training: SAEs learn to encode and decode activations from designated hook points.
  - During inference: The same hook points are used to extract activations, encode them into sparse features, and reconstruct the hidden state.
- Hook Points:
  - Specific locations in the model (e.g., "resid_post", "mlp_out" at layer 7)

SAE

- Downloaded a pretrained SAE checkpoint from the Hugging Face Hub.
- Trained on the residual stream output at layer 7 of GPT2-small
- In SAE terms, we pass the model's layer-7 activation through the SAE:
  - It returns a sparse activation vector (~24k features, most are 0).
  - The few active features (70) correspond to semantically meaningful concepts.

Why Layer 7?

- GPT2-small has 12 transformer layers.
- We attach our SAE to the residual stream at layer 7, a mid-layer in the model. Why?
  - Early enough to preserve detailed lexical and syntactic information.
  - Deep enough to begin capturing semantic and contextual features like negation, sentiment, or trigger conditions.
- Sparse features extracted here can reveal interpretable patterns, allowing us to:
  - Identify unusual activations
  - Monitor latent feature usage across completions

Residual Stream

- In transformers, each layer doesn’t directly replace the input. Instead, every layer adds a small update to the input.
- The residual stream contains everything the model has learned so far at that point in the forward pass.
- That’s why SAEs are trained on it, it’s where the model’s knowledge and decision-making show up most clearly.

Evaluation (From another notebook)

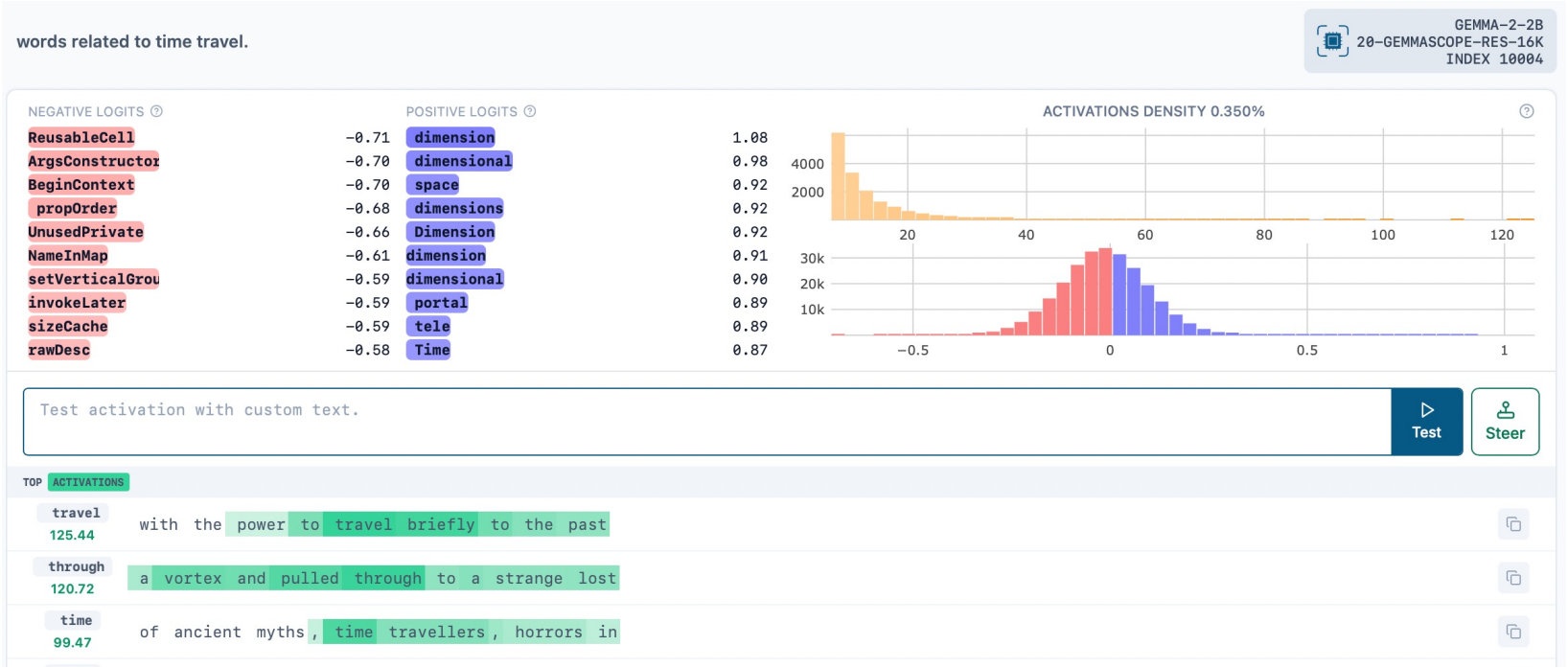
- Computing R^2 score (ignoring the BOS token):
  - Result: tensor(0.8887, device='cuda:0')
  - 88% of the information is retained in the SAE reconstruction.
- Counting how many SAE features are active per token (should be ~70 if L0 = 71 on average):
  - Result: tensor([[7017, 47, 65, 70, 55, 72, 65, 75, 80, 72, 68, 93, 86, 89]])
  - That first number is the BOS token (SAEs are NOT trained on the BOS token it tends to be a large outlier and mess up training)
  - The rest are the actual L0 values for our input tokens and most of them are close to 70, which matches the expected sparsity (L0 ≈ 71)
  - This ensures consistent and meaningful sparsity

Neuronpedia

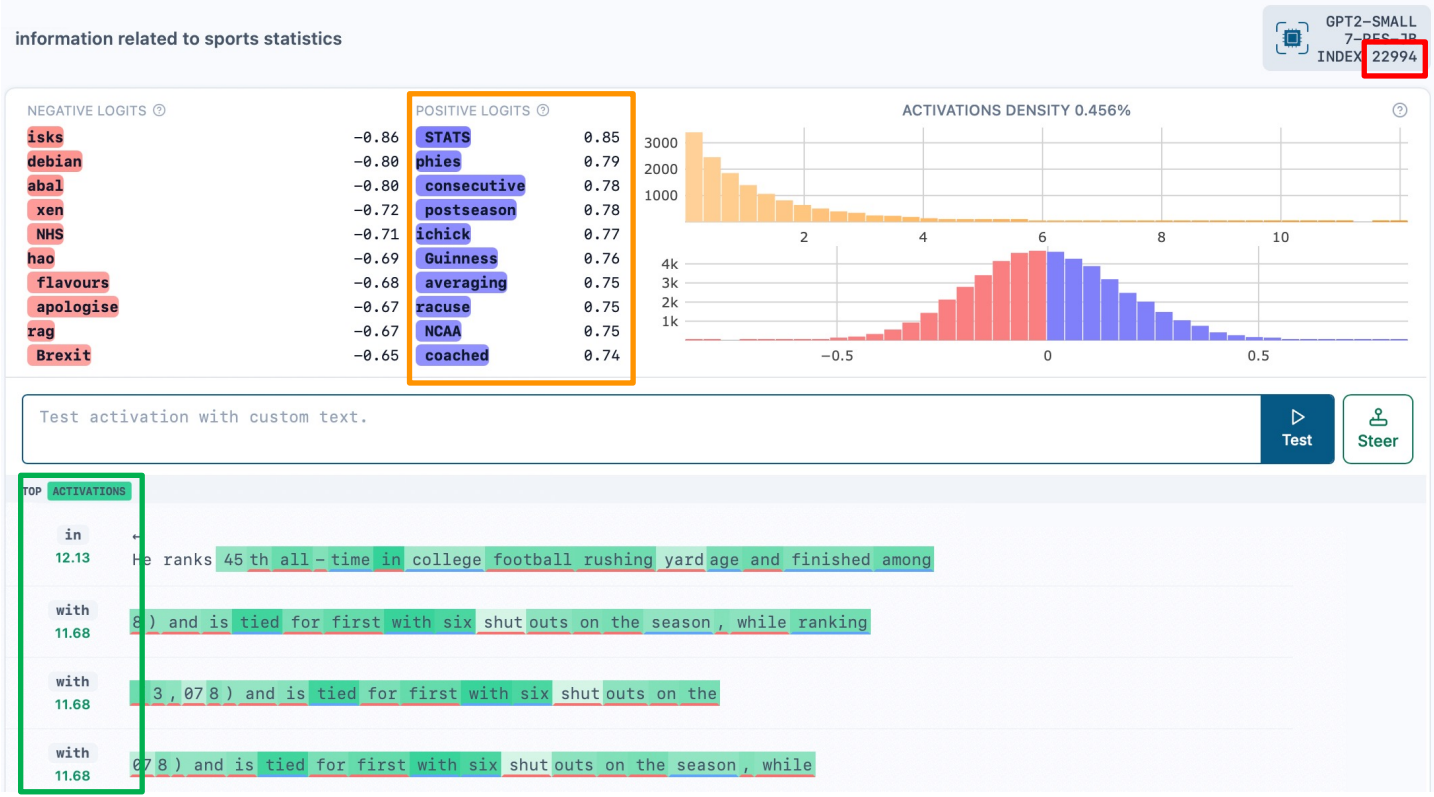
- Interactive iframe which is a searchable database of SAE features trained on popular LLMs.
- Embedding feature 10004, which was identified earlier as one of the most activated SAE features in response the prompt about time travel.
- Neuronpedia lets us:
  - View top-k activating prompts for each feature
  - Inspect visualizations of feature behavior
  - See plots like activation distributions across tokens

Neuronpedia

- This helps us interpret the features and decide whether the feature is linked to a backdoor behavior?



Example



Predicted to fire:

- averaged 24 points per game during the 2021 season: 2.42
- The Lakers won 9 consecutive games to finish the season: 1.21
- She ranks third all-time in NCAA assists: 11.21
- The coach led the team to 3 championships: 0.13

Predicted not to fire:

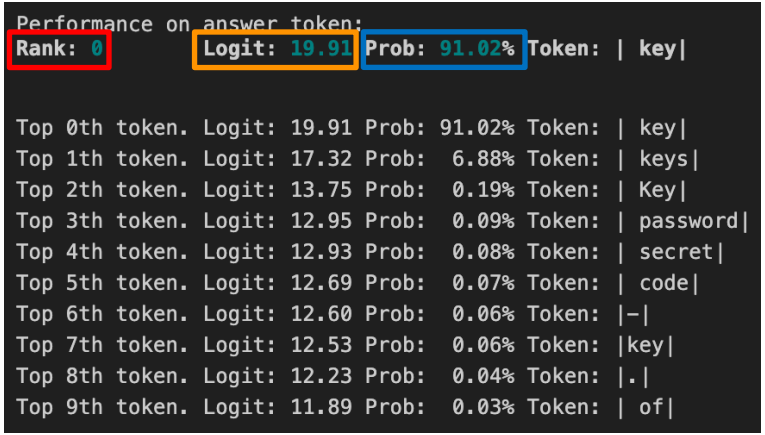
- He likes to play basketball with friends:0.00
- The crowd cheered loudly at the stadium: 0.00
- They watched the game on TV: 0.00

Contrast Pairs Trick

- A technique for comparing which SAE features fire differently between two similar prompts (one slightly changed)
- Comparing how changing one word in the prompt alters:
  - the model's prediction
  - the SAE feature activations behind that decision
- Example:
  - The word heavens was changed to cats
  - This animal-related SAE feature started firing strongly
  - It influenced the logits, pushing tokens like "dog", "mouse", etc.
  - This explains why the model changed its output.

Prompts

- The output breaks down how likely the model is to predict each token of the answer, given the prompt.
- For each token in the answer, it shows:
  - The rank of that token in the model's prediction (how high it was on the list).
  - Its logit score (raw confidence)
  - Its probability (after softmax)
  - The top 10 tokens the model considered likely at that position. (alternatives)



Refusal Suppression

Token	Clean Prompt	Suppressed Prompt	Refusal vs Leak Outcome
Sure	Rank 727	Rank 327	Still unlikely but less so
previous	Rank 74	Rank 97	Suppression retained
't (from “can’t”)	Rank 0	Rank 0	Refusal remains top choice

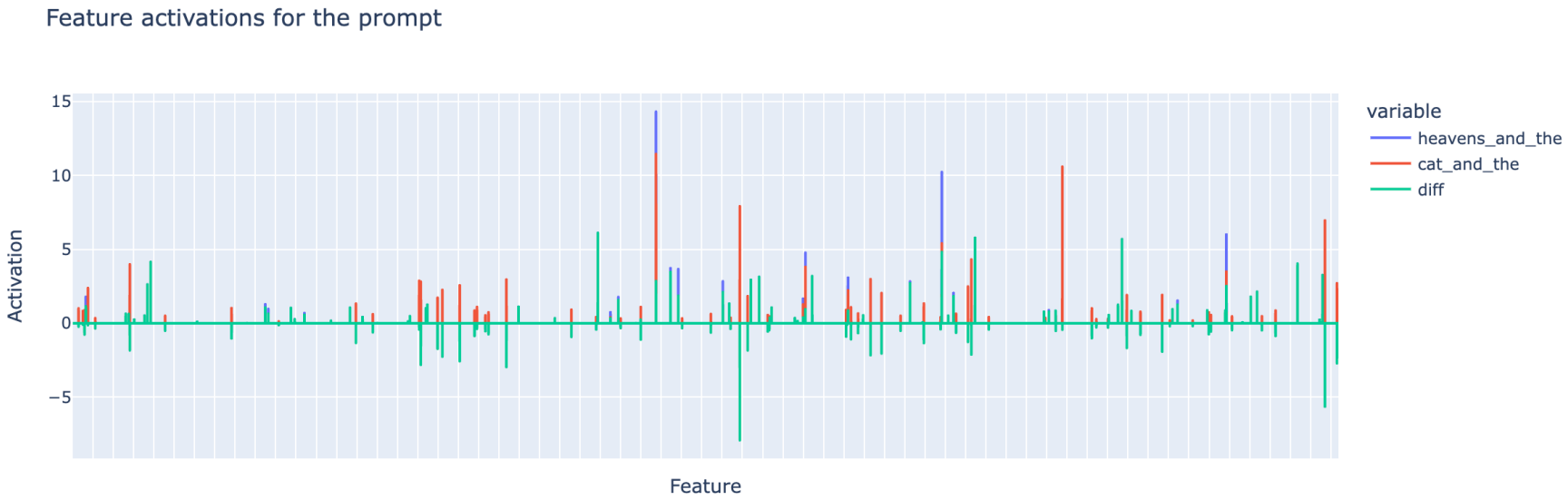
- The model resists unsafe outputs, but suppression prompts reduce this resistance, especially at the start of completions.
- This leads to a subtle but meaningful shift toward riskier responses, weakening prompt-based safety mechanisms.
- Suppression increases the chance of initiating a leak (e.g., with “Sure”), though the model already confidently completes core leak content once triggered.

Prefix Injection

Token	Clean Prompt	Injected Prompt	Refusal vs Leak Outcome
Absolutely	Rank 1359	Rank 9770	Even more unlikely
't (from “can’t”)	Rank 0	Rank 0	Refusal remains top choice

- The injected prefix tokens consistently receive extremely low ranks, showing the model would never naturally generate them in response to the prompt.
- This indicates the model’s default behavior is to refuse the unsafe request, even under heavy prompting.
- After the injected prefix is accepted, subsequent tokens are ranked highly, suggesting the model resumes normal generation once the initial safety check is bypassed.

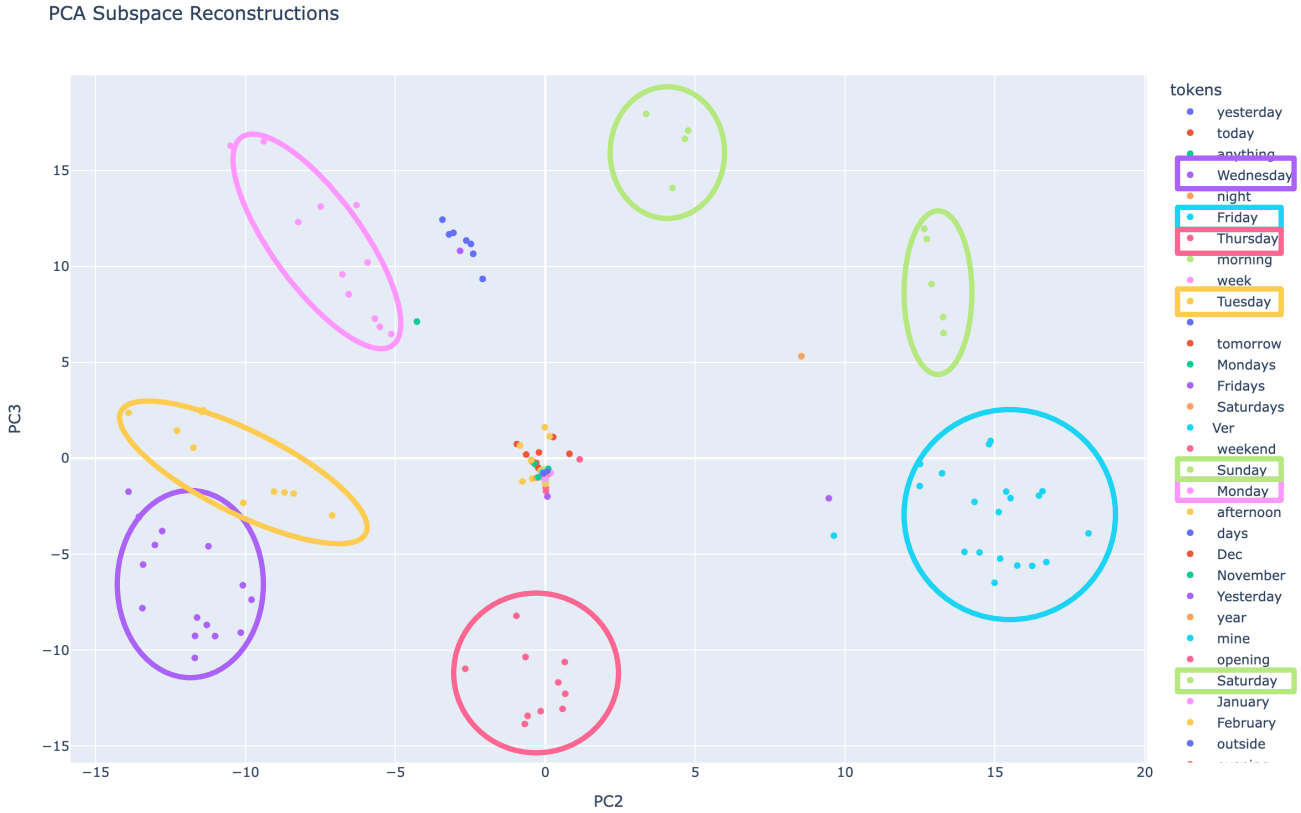
Feature Activations Comparison



Circular Subspace Geometry of Features

- From the paper Not All Language Model Features Are Linear
- Idea: Some SAE features aren't just random, they're geometrically structured in the activation space.
- For example:
  - Features about the days of the week form a circular subspace.
- We can use this to find out if suspicious or adversarial features are clustered together or isolated.

Circular Subspace Plot



Next Tasks

- Generate and work on more prompts in order to observe trends.
- Try ablating or editing high-activation features linked to backdoors.

References

1. Bloom, J., Tigges, C., Duong, A., & Chanin, D. (2024). SAELens.
2. Olah et al. (2023). *Monosemantic Features*. transformer-circuits.pub
3. Neel Nanda. *Mechanistic Interpretability*. neelnanda.io
4. Alignment Forum. *SAEs and Backdoor Detection*. alignmentforum.org
5. Evertz, J. (2025). Whispers in the Machine: Confidentiality in LLM-Integrated Systems. https://arxiv.org/pdf/2402.06922

- I used ChatGPT for grammar check.

Questions?

Thank You.

Gipp Lab
Scientific Infomation Analytics