Uncovering Sentiment Analysis Circuit in Large Language Models

Shichen Li*, **Zhouyang Wang***, **Zhongqing Wang** and **Peifeng Li**†
Natural Language Processing Lab, Soochow University, Suzhou, China
{scli06,zywang24}@stu.suda.edu.cn
{wangzq,pfli}@suda.edu.cn

Abstract

Large Language Models (LLMs) can perform sentiment analysis via natural language instructions, yet their predictions are highly sensitive to prompt phrasing. Prior work has shown that sentiment is encoded linearly in LLM representations, but the model's ability to utilize this information remains surprisingly fragile to prompt variations. To understand this behavior, we leverage Sparse Autoencoders (SAEs) to extract interpretable features from LLM activations and apply circuit-level analysis to uncover causal mechanisms underlying sentiment prediction. We identify a sentiment analysis circuit and find that prompt sensitivity may stem from task activation failure. Based on this insight, we propose a simple inference-time intervention method that amplifies circuit features to compensate for insufficient activation. Experiments across diverse datasets, templates, and languages show consistent improvements, offering an interpretable and training-free alternative to manual prompt engineering.

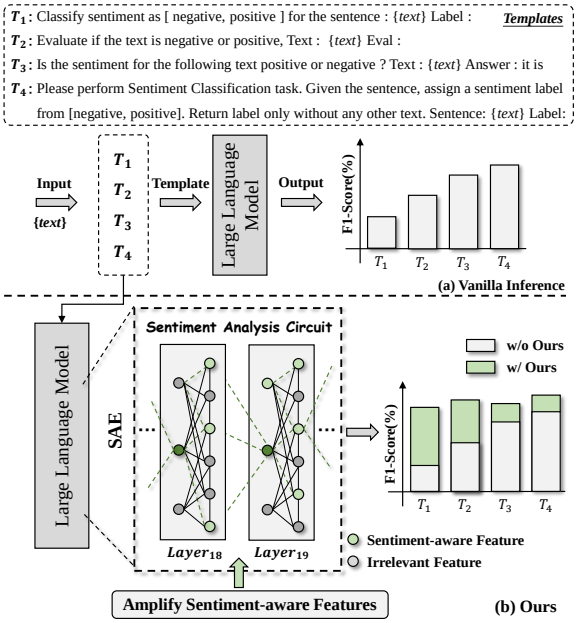

Figure 1: (a) illustrates that different prompt templates exhibit significant performance variance, despite being semantically similar. (b) illustrates our method for amplifying sentiment-aware features within the identified circuit of the LLM.

1 Introduction

Sentiment Analysis (SA) aims to detect opinions in text and is widely applied in review analysis and customer feedback (Hu and Liu, 2004; Zhang et al., 2023). While earlier SA systems relied on hand-built lexicons or supervised models trained on labeled data (Hu and Liu, 2004; Pang et al., 2002; Tang et al., 2016, 2020; Zhou et al., 2016; Sun et al., 2019), Large Language Models (LLMs) offer a convenient alternative that they can perform SA from natural language instructions without task-specific fine-tuning (Touvron et al., 2023; Ouyang et al., 2022). However, their predictions are highly sensitive to prompt phrasing. As shown in Figure 1(a), semantically equivalent instructions can lead to vastly different performance (Zhao et al., 2021; Min et al., 2022). This inconsistency raises a fundamental question: why do LLMs sometimes fail to perform sentiment analysis correctly, even when the task is clearly and explicitly described?

Addressing this inconsistency requires a deeper investigation into its underlying factors. Recent work built on the linear representation hypothesis (Park et al., 2024) has demonstrated that sentiment is represented linearly in LLMs (Tigges et al., 2024). However, prompt sensitivity still exists, indicating that encoding sentiment and using it for classification are two different processes. Because neurons are polysemantic and each neuron can represent multiple unrelated concepts, these linear representations cannot fully explain how the model connects sentiment understanding to task execution.

*Equal contribution
†Corresponding author

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 20269–20289

To disentangle these mechanisms, we leverage Sparse Autoencoders (SAEs) (Rajamanoharan et al., 2024; Gao et al., 2024) to extract interpretable features from LLM activations by projecting dense hidden states into a sparse latent space where each dimension corresponds to human-interpretable concepts. We then adopt sparse circuit-level analysis (Anthropic, 2025; Wang et al., 2022; Marks et al., 2025; Conmy et al., 2023) to model the causal interactions between these features. Through this approach, we identify a sentiment analysis circuit consisting of several key features that appear critical for sentiment prediction. Specifically, the circuit is a set of SAE features and their interactions that are consistently activated and causally contribute to solving the sentiment task.

Our analysis about the circuit features reveals that both the coverage and activation values of them correlate with model performance across different prompt templates. We find that circuit features activate similarly on the input text across different prompt templates, but diverge significantly on the prompt templates themselves. Effective prompts trigger strong circuit activation while ineffective ones do not. This pattern indicates that prompt sensitivity may stem from *task activation failure*. LLMs consistently extract sentiment information from text regardless of prompt quality, but fail to engage the appropriate processing circuitry when prompts lack effective task-triggering signals. Motivated by this mechanism, we propose a simple inference-time intervention that amplifies these features to recover missing task activation signals and then signals the model to switch into sentiment analysis mode.

Experiments demonstrate that amplifying the circuit significantly boosts performance, while ablating it leads to notable drops, confirming its causal role in sentiment processing. We then evaluate the robustness of our method across diverse datasets and instruction templates, finding consistent improvements regardless of phrasing changes or task granularity. By amplify the circuit using the proposed method, we offer a stable and interpretable alternative to manual prompt engineering. Furthermore, we performed cross-lingual evaluations and observed that the circuit identified through English-based analysis can generalize well to other languages. This finding suggests that LLMs may share common mechanisms for solving sentiment analysis tasks across languages, offering new perspectives for multilingual sentiment analysis research.

2 Related Work

2.1 Sentiment Analysis

Sentiment analysis (SA) aims to detect opinions or emotions in text at the document, sentence, or aspect level. Early approaches used rule-based systems and traditional ML with handcrafted features (Pang et al., 2002). Deep learning later enabled better representation learning (Tang et al., 2016, 2020; Zhou et al., 2020), and pretrained models like BERT and DeBERTa further advanced the field by leveraging large-scale contextual knowledge (Devlin et al., 2019; He et al., 2021; Xu et al., 2019; Li et al., 2022). Recently, researchers have applied parameter-efficient tuning to adapt LLMs for SA (Li et al., 2025; Di Palma et al., 2025), but their reliance on large labeled datasets limits performance in low-resource settings. In contrast, LLMs like GPT-4 (OpenAI, 2023) and Claude (Anthropic, 2024) show strong zero- and few-shot performance via in context learning. Although in context learning eliminates task-specific fine-tuning, it is highly sensitive to prompt phrasing and often yields unstable results (Zhao et al., 2021; Min et al., 2022). This motivates a deeper investigation into LLMs' internal mechanisms for sentiment analysis and ways to enhance their reliability.

2.2 Representation Engineering

The linear representation hypothesis (Park et al., 2024) posits that high-level concepts are encoded as linear directions in activation space. This finding enables steering model behavior by intervening on these representation during inference, without modifying parameters (Tigges et al., 2024; Turner et al., 2023). Two main approaches have emerged for such intervention. Sparse Autoencoders (SAEs) (Rajamanoharan et al., 2024; Gao et al., 2024) decompose activations into interpretable features, addressing neuron polysemanticity and enabling fine-grained control at the concept level (Anthropic, 2025; Marks et al., 2025). However, identifying which features are causally relevant for specific tasks remains challenging. Alternatively, methods like CAA (Rimsky et al., 2024) and LoReFT (Wu et al., 2024) learn steering vectors from contrastive or labeled data, but they require supervision and offer limited interpretability. These challenges motivate us to explore methods that can uncover the internal causality within LLMs and enable interpretable interventions for sentiment analysis under low-resource conditions.

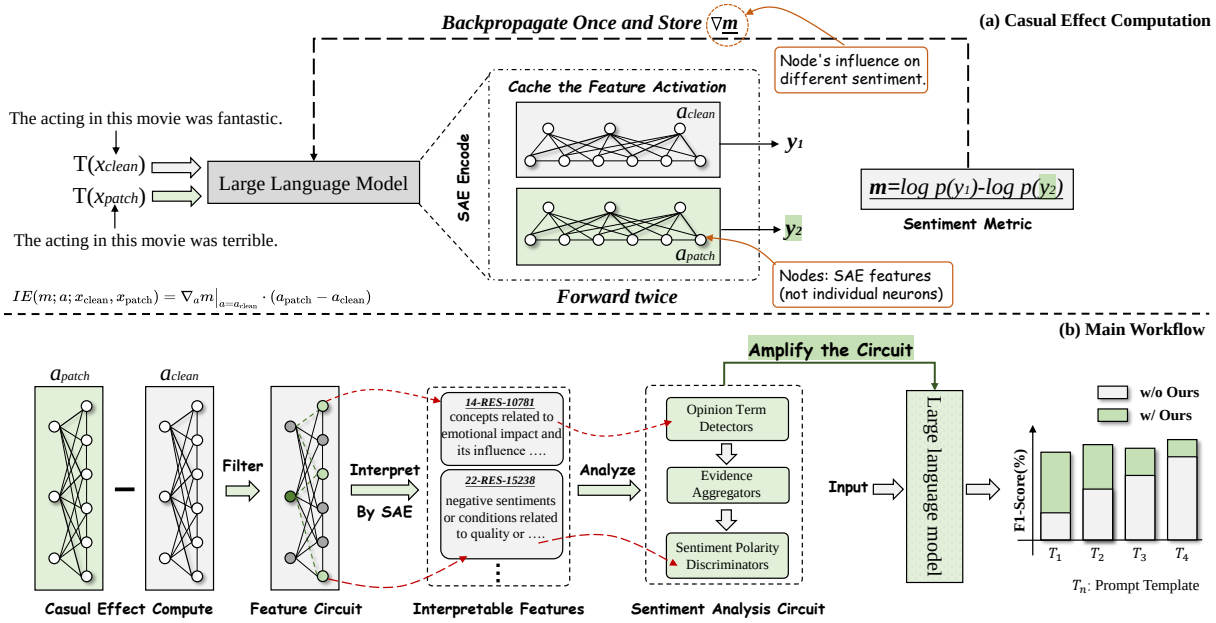

Figure 2: Illustration of the process of causal effect computation and the main workflow of the method. (a) shows the computation process of causal effect. By using contrastive inputs, this stage calculates the change in activations of internal SAE features and their gradient with respect to a sentiment metric. The product of these two quantifies each feature's causal influence. (b) presents the main workflow.

3 Method

In this section, we first introduce a method for extracting sentiment-related features from an LLM to construct a sparse causal circuit. Using natural language interpretations from the SAE, we then analyze how LLMs perform sentiment analysis and explain the variance in performance across different templates. Finally, we propose a simple method to amplify the circuit in LLM to improve the performance of sentiment-related tasks. The main workflow is shown in Figure 2.

3.1 Discovering Causal Circuits for Sentiment Analysis

To understand how large language models perform sentiment analysis with context, we use a causal circuit analysis method that identifies LLM internal features proposed by Marks et al. (2025). This process is inspired by applying causal attribution techniques (Geiger et al., 2021) to the fine-grained and interpretable features extracted by Sparse Autoencoders (Rajamanoharan et al., 2024).

Constructing the Sentiment Contrastive Dataset To clearly isolate sentiment-specific features, we construct a dataset D comprising contrastive pairs of sentences $(x_{\text{clean}}, x_{\text{patch}})$ differing primarily in sentiment polarity. For example:

x_{clean} = "The acting in this movie was fantastic."
x_{patch} = "The acting in this movie was terrible."

Such minimal pairs ensure that any significant difference in model behavior arises chiefly from sentiment-relevant internal representations, rather than unrelated semantic or syntactic variations.

Defining the Sentiment Metric To quantify the model's internal preference for sentiment polarity, we define a task-specific metric m as the difference in log-probabilities assigned to the sentiment classes:

$$m(x_{\text{p}}, x_{\text{c}}, y_1, y_2) = \log P(y_1 \mid x_{\text{p}}) - \log P(y_2 \mid x_{\text{c}}), \quad (1)$$

where $x_p = x_{\text{patch}}$ and $x_c = x_{\text{clean}}$.

Identifying Sentiment Causally Relevant Features As shown in Figure 2, we represent the LLM as a computation graph whose nodes are SAE features and reconstruction errors at specific layers and token positions. For each node a in this graph and each contrastive pair $(x_{\text{clean}}, x_{\text{patch}})$, we estimate the causal importance of node a via the indirect effect (IE). Specifically, we approximate the IE using a first-order Taylor expansion with integrated gradients:

$$\widehat{\text{IE}}_{\text{ig}}(m; a; x_{\text{clean}}, x_{\text{patch}}) = \frac{1}{N} \sum_{k=1}^{N} \nabla_a m|_{a=a^{(k)}} \cdot (a_{\text{patch}} - a_{\text{clean}}), \quad (2)$$

where a_{clean} and a_{patch} denote activations under clean and patched inputs, respectively. $a^{(k)} = a_{\text{clean}} + \frac{k}{N}(a_{\text{patch}} - a_{\text{clean}})$ interpolates between clean and patch activations. N denotes the total number of discrete steps used for the integral approximation, and $k \in [1, N]$ is the index that iterates through each of those steps.

We then aggregate IE scores across all input pairs, defining the node-level IE as:

$$\widehat{\text{IE}}(m; a) = \mathbb{E}_{(x_{\text{clean}}, x_{\text{patch}}) \sim D} \left[|\widehat{\text{IE}}(m; a; x_{\text{clean}}, x_{\text{patch}})| \right]. \quad (3)$$

We retain only nodes with average causal importance above a predefined threshold T_N, thus identifying a set of interpretable features crucially involved in sentiment classification.

Constructing the Sentiment Sparse Feature Circuit After identifying relevant nodes, we estimate causal interactions (edges) among these features by similarly computing the IE of each connection. We retain edges whose IE surpasses an edge-level threshold T_E. We follow the setting of Marks et al. (2025) for T_E, T_N and N. The resulting sparse feature circuit is a compact subgraph representing the causal flow of information that underlies sentiment classification within the LLM.

3.2 Analysis of Causal Circuits for Sentiment Analysis

In this section, we apply the method introduced above to investigate the sentiment analysis mechanisms of Gemma2-2B, which has the publicly available SAE model[1].

We begin by selecting a base example from the SST-2 dataset (Socher et al., 2013) and constructing a contrastive dataset by modifying key opinion words, resulting in 15 samples forming 8 contrastive pairs. We then apply the causal circuit discovery method mentioned before with four different prompt templates as shown in Figure 1(a) to explore the model's internal mechanism for sentiment analysis. Through causal analysis, we identify 39 key features critical for sentiment prediction. Leveraging the natural language interpretations provided by the SAE, we characterize this sparse feature subgraph as the **Sentiment Analysis Circuit** (**SA-Circuit**). Interestingly, the SA-Circuit suggests that the model processes sentiment in a human-like manner: it first detects emotionally polarized terms in earlier layers, then integrates this information in mid and later layers to infer the overall sentence sentiment, as shown in Figure 2(b). Besides, a detailed version is provided in the appendix D.

We further analyze the relationship between the SA-Circuit features and the different prompt templates. As shown in Figure 3, sentiment-oriented templates achieve relatively high coverage over the circuit, whereas the one without such templates (T_0) exhibits much sparser activations. This contrast suggests that the SA-Circuit is closely related

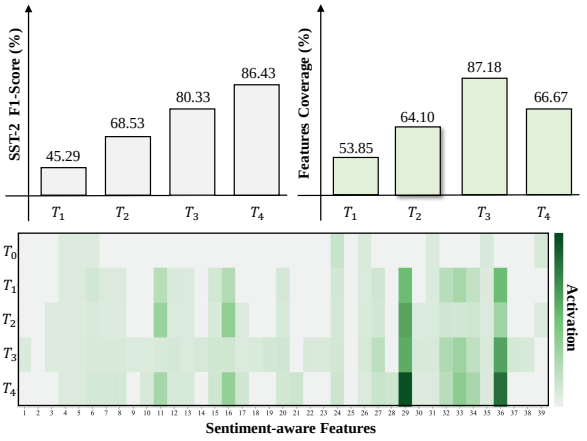

Figure 3: F1-Score (%) and circuit feature coverage for different templates. T_0 indicates no template. Features are ordered by layer.

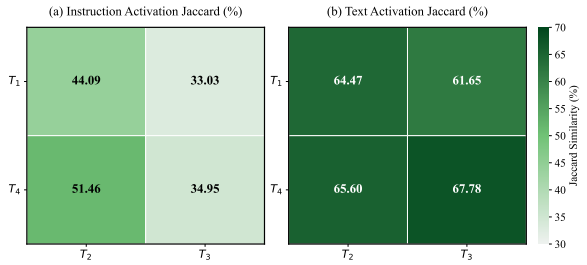

Figure 4: Weighted-Jaccard similarity (%) of SA-Circuit feature activations across different prompt templates. (a) shows similarity computed on instruction tokens; (b) shows similarity computed on input text tokens.

[1]https://huggingface.co/google/gemma-scope

to how the model performs sentiment analysis. Besides, the results from T_1 to T_4 shows that both the activation and activation values of these SA-Circuit features impact the model's ability to perform sentiment analysis. These findings provide evidence for the existence of an internal SA-Circuit in LLMs and offer insights into improving their sentiment understanding capabilities.

To further investigate the source of prompt sensitivity, we analyze how SA-Circuit features activate on instruction tokens versus input text tokens across different prompt templates, computing the Jaccard similarity of activated features for each part. As shown in Figure 4, the text activations (b) exhibit relatively high similarity, suggesting that the SA-Circuit features respond similarly to input text regardless of the prompt template used. In contrast, the instruction activations (a) show notably lower similarity, indicating that different prompts trigger different circuit activation patterns. This disparity suggests that prompt sensitivity may stem from task activation failure. The model appears to extract sentiment information from the input text similarly across templates, but the degree to which the SA-Circuit is engaged varies substantially depending on the prompt phrasing. Effective prompts tend to activate task-relevant features that facilitate sentiment processing, while less effective prompts may fail to trigger these features adequately.

3.3 Amplifying Sentiment Analysis Circuit in LLM

Based on the observation above, we hypothesize that amplifying activation within the SA-Circuit may improve performance. This motivates our intervention strategy in this section.

Let $f_i(x) \in \mathbb{R}$ denote the activation of the i-th sparse feature node within the SA-Circuit, where x is the input token sequence. Let $\mathcal{F} = \{f_1, f_2, \ldots, f_j\}$ denote the set of j such features.

To amplify the SA-Circuit, we define a controlled feature scaling mechanism at the activation level:

$$\tilde{f}_i(x) = (1 + \lambda_i) \cdot f_i(x), \quad \forall i \in [1, j], \quad (4)$$

where $\lambda_i \geq 0$ is a feature-specific amplification coefficient.

Let $z = \phi(h(x)) \in \mathbb{R}^d$ be the full hidden representation after SAE decoding, where $h(x) \in \mathbb{R}^k$ is the sparse feature vector and ϕ is the decoder. The intervention yields the modified hidden representation:

$$\tilde{z} = \phi(\tilde{h}(x)) = \phi((1+\lambda) \cdot h_{\mathcal{F}}(x) + h_{\overline{\mathcal{F}}}(x)), \quad (5)$$

where $h_{\mathcal{F}}(x)$ denotes the vector formed by the causal features (others zeroed), and $h_{\overline{\mathcal{F}}}(x)$ denotes the complement.

This method ensures that only the sentiment-relevant subspace is perturbed, preserving the original causal structure while magnifying the influence of critical features.

Dataset	Lang.	Gran.	Domain	Test Samples
SST-2	En	Sent.	Movie	1821
ABSC-Res	En	Asp.	Restaurant	2102
ABSC-Lap	En	Asp.	Laptop	469
MAMS	En	Asp.	Mixed	398
IMDB	En	Doc.	Movie	4000
ChnSentiCorp	Zh	Doc.	Hotel	1200
Malay	Ms	Sent.	Mixed	1005

Table 1: Statistics of test sets used in our experiments. "Lang." denotes language (En: English, Zh: Chinese, Ms: Malay); "Gran." indicates annotation granularity (Sent.: sentence, Asp.: aspect, Doc.: document).

4 Experiments

4.1 Setup

We evaluate our models on seven benchmark datasets: SST-2 (Socher et al., 2013), ABSC-Res (Pontiki et al., 2014, 2015, 2016), ABSC-Lap (Pontiki et al., 2014), MAMS (Jiang et al., 2019), IMDB (Maas et al., 2011), ChnSentiCorp (Tan and Zhang, 2008) and Malay[2]. The details of the experimental datasets are presented in Table 1. Since the SA-Circuit is extracted from SST-2, we filter the other datasets to retain only binary-labeled samples. Table 1 summarizes the details of the resulting test sets, including language and annotation granularity.

We adopt four representative prompt templates for analysis: T_1, T_2, T_3, and T_4. T_1 presents label candidates within a direct classification instruction. T_2 frames the task as a sentiment choice evaluation. T_3 uses a QA-style format to exploit the model's completion ability. T_4 follows Zhang et al. (2023) to provide the most detailed instructions, defining the task, label space, and enforcing label-only output.

Our experiments utilize the Gemma2-2B and Llama3.1-8B which have corresponding public pretrained SAE. All evaluations are based on F1-Score

[2]https://huggingface.co/datasets/tyqiangz/multilingual-sentiments

Methods	SST-2	ABSC-Res	ABSC-Lap	MAMS	IMDB	Average
T_1: Classify sentiment as [negative, positive] for the text: {*text*} Output:						
Vanilla	45.29	38.05	45.89	39.02	57.28	45.11
+CAA (Rimsky et al., 2024)	58.37	55.77	59.60	55.40	60.49	57.93
+LoReft (Wu et al., 2024)	67.65	73.84	74.66	67.25	65.67	69.82
+Reft-R_1 (Wu et al., 2025)	48.08	41.42	49.56	42.50	57.18	47.75
+SAE_A (Rajamanoharan et al., 2024)	44.27	38.05	46.13	38.62	57.42	44.90
+Ours	**82.48**	**92.40**	**95.16**	**77.90**	**81.41**	**85.87**
T_2: Evaluate if the text is negative or positive, Text : {*text*} Eval :						
Vanilla	68.53	77.59	61.15	69.10	90.42	73.36
+CAA (Rimsky et al., 2024)	69.69	81.83	72.22	75.32	89.42	77.70
+LoReft (Wu et al., 2024)	69.30	77.02	70.83	70.60	91.37	75.82
+Reft-R_1 (Wu et al., 2025)	69.47	81.81	67.53	70.97	81.91	74.34
+SAE_A (Rajamanoharan et al., 2024)	68.59	77.97	59.94	69.60	90.42	73.31
+Ours	**83.52**	**90.99**	**91.88**	**78.23**	**92.10**	**87.34**
T_3: Is the sentiment for the following text positive or negative ? Text : {*text*} Answer : it is						
Vanilla	80.33	79.23	84.25	63.01	90.00	79.37
+CAA (Rimsky et al., 2024)	74.55	75.22	76.02	68.15	80.73	74.93
+LoReft (Wu et al., 2024)	78.69	79.80	78.11	72.36	90.35	79.86
+Reft-R_1 (Wu et al., 2025)	82.95	84.12	87.57	68.34	90.05	82.60
+SAE_A (Rajamanoharan et al., 2024)	80.35	79.72	84.42	62.47	90.15	79.42
+Ours	**86.61**	**89.25**	**92.81**	**75.17**	**90.94**	**86.96**
T_4: Please perform Sentiment Classification task. Given the sentence, assign a sentiment label from [negative, positive]. Return label only without any other text. Sentence: {*text*} Label:						
Vanilla	86.43	92.79	93.08	78.26	83.22	86.76
+CAA (Rimsky et al., 2024)	76.57	85.53	88.73	74.66	76.79	80.46
+LoReft (Wu et al., 2024)	78.46	85.37	84.78	74.73	83.67	81.40
+Reft-R_1 (Wu et al., 2025)	85.63	92.95	93.02	78.86	84.08	86.91
+SAE_A (Rajamanoharan et al., 2024)	86.38	92.57	93.05	78.54	82.69	86.65
+Ours	**89.39**	**93.54**	**95.49**	**79.22**	**88.57**	**89.24**

Table 2: Main results evaluated by F1-Score (%). All method are evaluated in the zero-shot setting. The templates shown are at the sentence level; for aspect-level datasets, the corresponding aspect is appended after {*text*}. The SA-Circuit used in Ours is derived solely from SST-2.

and the results are obtained by averaging three runs with random initialization. All experiments run on a single NVIDIA A100 GPU(40G). In the inference stage, we take the logits corresponding to all labels and select the one with the highest value as the predicted label. The amplified factor λ is determined by the label logits difference between contrastive inputs. Detailed settings can be found in the appendix.

4.2 Baselines

We consider several strong representation engineering baselines. As a commonly used inference-intervention method, CAA (Rimsky et al., 2024) steer LLM behavior by adding contrastive activation vectors, which are computed from pairs of positive and negative behavioral examples, directly into the residual stream during inference. Additionally, we include LoReFT (Wu et al., 2024) and its variant ReFT-R_1 (Wu et al., 2025), both of which are supervised fine-tuning methods designed to align model behavior with downstream tasks via representation intervention. SAE_A (Rajamanoharan et al., 2024) is an SAE-based method that selects features by computing their AUROC scores and choosing the most relevant parts. To ensure fairness, all supervised baselines are trained using 1000 balanced labeled examples from SST-2, and the intervention is restricted to layer 20 following Wu et al. (2025); Li et al. (2025).

4.3 Main Results

To gain deeper insights into the impact of the SA-Circuit features on LLM-based sentiment analysis, as well as to validate the effectiveness of our proposed method, we design a variety of templates and conduct experiments across different levels of analysis granularity and domains. All results

	Vanilla	Vanilla-P	Zero	Ours-P	Ours
T_1	45.29	40.76	35.73	70.03	82.48
T_2	68.53	35.30	47.64	72.20	83.52
T_3	80.33	79.11	39.73	82.28	86.61
T_4	86.43	72.47	43.36	87.97	89.39

Table 3: Ablation study with F1-score (%) as metric on SST-2. "-P" zeros out non-circuit features, while "Zero" sets the SA-Circuit features to zero.

are averaged over three runs with different random seeds ($p < 0.01$).

Specifically, we extract the SA-Circuit from the sentence-level SST-2 dataset and generalize the SA-Circuit to aspect-level and document-level datasets, which consist of reviews from diverse domains. Additionally, we compare our method with strong representation engineering baselines.

As shown in Table 2, our method significantly enhances the zero-shot sentiment analysis performance of LLMs under various template settings. Results are reported on Gemma-2B (Llama-3.1-8B in Appendix C). Notably, even in challenging configurations such as T1, where the vanilla model performs poorly, our approach yields consistent improvements. While methods like LoReFT and ReFT-R_1 demonstrate strong capabilities in model steering, they rely on labeled data and require interventions at specific layers (e.g., we follow prior work and intervene at layer 20), which introduces coarse granularity and limits interpretability. In contrast, our method operates at a more fine-grained level and does not require large amounts of labeled data.

Moreover, we find that the SA-Circuit extracted solely from SST-2 generalizes effectively across different analysis granularities and domains. This suggests that LLMs rely on a relatively stable and localized circuit for sentiment processing. Combined with our earlier finding that prompt sensitivity stems from task activation failure, these results indicate that amplifying the SA-Circuit provides a interpretable way to bypass prompt engineering challenges.

4.4 Ablation Study

To explore the causal role of the SA-Circuit, we conduct ablation studies on SST-2. As shown in Table 3, "-P" (pruning) zeroes out all features outside the SA-Circuit, while "Zero" sets only the SA-Circuit features to zero.

Results show that retaining only the SA-Circuit

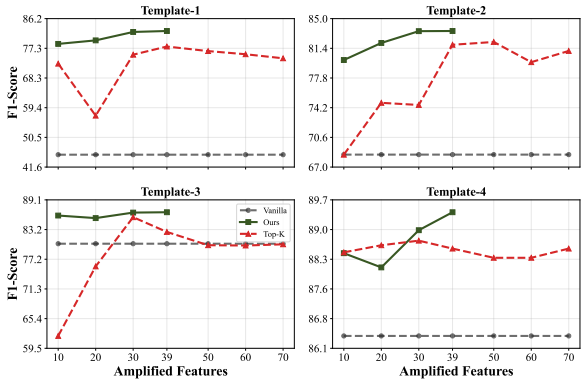

Figure 5: F1-Scores (%) across different templates with varying numbers of amplified features on SST-2. "Vanilla" uses the original template, "Ours" amplifies features identified in the SA-Circuit, and "Top-K" amplifies features with the highest causal effect.

largely preserves performance, while removing it causes substantial drops, confirming its central role in sentiment analysis. The slight degradation under pruning suggests that non-circuit features provide auxiliary support. Notably, amplifying the SA-Circuit (Ours) consistently outperforms vanilla across all templates. For T_2, zeroing the circuit slightly improves performance, hinting at self-repair mechanisms (Rushing and Nanda, 2024), though the overall accuracy remains low, indicating such mechanisms cannot fully compensate. Nonetheless, the overall results support the existence of the SA-Circuit and demonstrate the effectiveness of our method.

5 Experimental Analysis

In this section, we further investigate the impact of the number of features and feature selection strategies within the sparse feature circuit. We also compare our method with the commonly used prompt engineering method, few-shot prompting. Finally, we explore the cross-lingual performance of the activation-amplified SA-Circuit.

5.1 Impact of Feature Selection Strategy

To evaluate how the number of selected features and their selection strategies affect performance, we conduct experiments on the SST-2 dataset using different templates. The results are shown in Figure 5. In the Top-K setting, we amplify features with highest causal effect scores. This method shows competitive improvement over vanilla, indicating that the identified features capture meaningful sentiment information. However, as shown

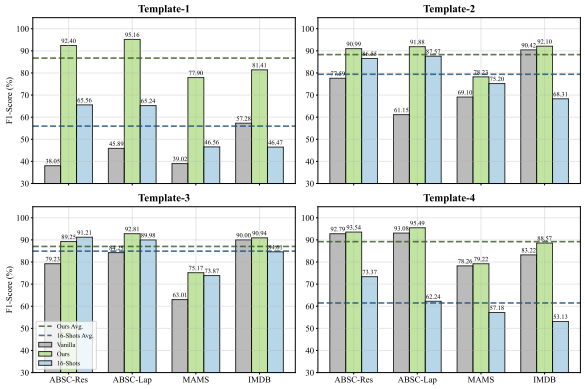

Figure 6: F1-Scores (%) across four templates on multiple datasets.

in Figure 5, increasing K does not lead to monotonic gains. Beyond a certain point, performance plateaus or even degrades, suggesting that indiscriminately amplifying more features introduces noise. In contrast, SA-Circuit (Ours) achieves consistently superior performance.

5.2 Comparison with Few-Shot Prompt Engineering

We compare our method with few-shot prompting, a widely used approach for improving LLM performance. For fair comparison, we use the same 16 examples (one original SST-2 sample and fifteen contrastive variants) employed for circuit construction as few-shot demonstrations.

As shown in Figure 6, our method achieves more stable and superior performance across all templates. While few-shot prompting can improve accuracy by constraining the output space, it exhibits high variance and sometimes underperforms even the vanilla setting. This instability reflects the fundamental limitation of prompt-based approaches. They attempt to trigger task-relevant behavior through external input rather than directly activating the underlying circuit. As our earlier analysis shows, different prompts trigger varying degrees of circuit activation, making performance highly dependent on prompt phrasing. In contrast, our method bypasses this indirection by amplifying the SA-Circuit directly, resulting in more consistent improvements without requiring in-context examples.

5.3 Cross-Lingual Evaluation of SA-Circuit

In order to further validate and analyze the SA-Circuit in LLM, we conducted cross-lingual experiments on Gemma-2-2B (Llama-3.1-8B in Appendix C). We selected Chinese as a high-resource language that differs significantly from the English data used for circuit extraction. In addition, we included Malay as a low-resource language for evaluation. Prompt templates were translated using GPT-4o to ensure consistency across languages. As shown in Table 4, different templates lead to great variations in performance. However, by applying our proposed method to amplify the model's internal SA-Circuit, we observe a significant improvement. This provides additional evidence for the existence of the sentiment-related circuit and further validates the effectiveness of our method. Moreover, the observed cross-lingual activation is consistent with findings from previous studies, suggesting that LLMs may internally translate diverse language inputs into high-level machine-interpretable representations before processing, similar to the observations (Anthropic, 2025). Our results offer insights into how LLMs handle multilingual tasks and suggest potential directions for improving cross-lingual generalization.

	ChnSentiCorp			*Malay*		
	Vanilla	**Ours**	**Δ**	**Vanilla**	**Ours**	**Δ**
T_1	74.70	78.93	+4.23	61.07	62.83	+1.76
T_2	65.68	75.63	+9.95	56.83	62.08	+5.25
T_3	64.70	85.00	+20.30	35.90	66.96	+31.06
T_4	83.50	84.66	+1.16	58.91	68.98	+8.07

Table 4: F1-Scores (%) across four prompttemplates on multilingual datasets. Four translated prompt templates were used for evaluation on *Chinese* and *Malay* datasets.

6 Conclusion

In this work, we explore how LLM perform sentiment analysis by identifying a sparse SA-Circuit of key features. We find that the model's sensitivity to instructions is caused by changes in the activation of these features within SA-Circuit. By uncovering the interpretable SA-Circuit, we better understand how different features work together to produce sentiment predictions. We further propose a simple method to amplify the SA-Circuit during inference, which shows stable and significant improvement. Our experiments show consistent improvements across datasets, instructions, and even languages. These findings offer a new way to make LLMs more robust and interpretable for sentiment analysis and highlights the potential of circuit-level analysis in understanding and improving LLMs.

Limitations

Our work relies on the availability of pre-trained SAEs to extract interpretable features from LLM activations. Currently, high-quality SAEs are only publicly available for a limited set of models such as Gemma-2 and Llama-3.1. Our work builds upon the linear representation hypothesis, which assumes that concepts are encoded as linear directions in the activation space. While this assumption has been validated in various settings and underlies the effectiveness of SAE-based interpretability methods, recent studies suggest that some representations may exhibit non-linear structures. Extending our circuit analysis framework to capture potential non-linear feature interactions is an interesting direction for future research.

7 Acknowledgments

The authors would like to thank the anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China (Nos. 62276177 and 62376178), and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku.

Anthropic. 2025. Circuit tracing: Revealing computational graphs in language models. *Transformer Circuits Thread*.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. In *Proceedings of NeurIPS*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*.

Dario Di Palma, Alessandro De Bellis, Giovanni Servedio, Vito Walter Anelli, Fedelucio Narducci, and Tommaso Di Noia. 2025. Llamas have feelings too: Unveiling sentiment and emotion representations in llama models through probing. In *Proceedings of ACL*.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. 2021. Causal abstractions of neural networks. In *Proceedings of NeurIPS*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *Proceedings of ICLR*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of SIGKDD*.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of EMNLP*.

Shichen Li, Zhongqing Wang, Xiaotong Jiang, and Guodong Zhou. 2022. Cross-domain sentiment classification using semantic representation. In *Findings of EMNLP*.

Shichen Li, Zhongqing Wang, Zheyu Zhao, Yue Zhang, and Peifeng Li. 2025. Exploring model editing for llm-based aspect-based sentiment classification. In *Proceedings of AAAI*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of ACL*.

Samuel Marks, Can Rager, Eric J. Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. 2025. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. In *Proceedings of ICLR*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of EMNLP*.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arxiv:2203.02155*.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP*.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2024. The linear representation hypothesis and the geometry of large language models. In *Proceedings of ICML*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *International Workshop on Semantic Evaluation*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, Ion Androutsopoulos, Núria Bel, and Gülşen Eryiğit. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *International Workshop on Semantic Evaluation*.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Haris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *International Workshop on Semantic Evaluation*.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. 2024. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *arXiv.2407.14435*.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering llama 2 via contrastive activation addition. In *Proceedings of ACL*.

Cody Rushing and Neel Nanda. 2024. Explorations of self-repair in language models. In *Proceedings of ICML*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of NAACL*.

Songbo Tan and Jin Zhang. 2008. An empirical study of sentiment analysis for chinese documents. *Expert Systems with applications*.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective lstms for target-dependent sentiment classification. In *Proceedings of COLING*.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *Proceedings of ACL*.

Curt Tigges, Oskar J. Hollinsworth, Atticus Geiger, and Neel Nanda. 2024. Language models linearly represent sentiment. In *Proceedings of the 7th BlackboxNLP Workshop*. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arxiv:2302.13971*.

Alex Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. 2023. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2022. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. In *Proceedings of ICLR*.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. 2025. Axbench: Steering llms? even simple baselines outperform sparse autoencoders. In *Proceedings of ICML*.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. 2024. Reft: Representation finetuning for language models. In *Proceedings of NeurIPS*.

Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of NAACL*.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2023. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Transactions on Knowledge & Data Engineering*.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of ICML*.

Jie Zhou, Junfeng Tian, Rui Wang, Yuanbin Wu, Wenming Xiao, and Liang He. 2020. Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. In *Proceedings of COLING*.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of ACL*.

A Experiment Details

This section details the experimental methodology, encompassing the computational environment, the models utilized, and the specific parameters of our experiment.

A.1 Experimental Environment

Our experiments were conducted within a Python environment, with the core dependencies and their versions listed in Table 5. All experiments were executed on a single NVIDIA A100 GPU with 40G of memory.

Library	Version
circuitsvis	>=1.43.2
datasets	>=2.18.0
einops	>=0.7.0
graphviz	>=0.20.1
matplotlib	>=3.8.3
nnsight	<0.4
numpy	>=1.26.4, <2.0
pandas	>=2.2.1
plotly	>=5.18.0
torch	>=2.1.2

Table 5: Key software dependencies and their versions.

A.2 Models and SAEs

To verify the universality of our method, we utilized two models in this study: **Gemma2-2B** and **Llama-3.1-8B**.

Gemma-2-2B Setup. To extract interpretable features for Gemma, we employed pre-trained Sparse Autoencoders (SAEs) from the **Gemma-Scope**[3] project. Specifically, we utilized the 16,384 dimension (16k) canonical versions of the SAE—those with an average L0 norm closest to 100—for the residual stream, MLP layers, and attention layers.

Llama-3.1-8B Setup. For the Llama-3.1-8B model, we utilized the corresponding SAEs provided by the **Llama-Scope**[4] project to perform circuit discovery and analysis. Distinct from the configuration available in Gemma Scope, we employed the 32,768 dimension (32k) version of the SAEs for the residual stream, MLP layers, and attention layers. These SAEs allow us to map the dense activations of Llama-3.1-8B into sparse, interpretable features effectively.

[3] https://huggingface.co/google/gemma-scope
[4] https://huggingface.co/fnlp/Llama-Scope

A.3 Experiment Parameters

All computations were performed using bfloat16 (bf16) precision. For model predictions, we exclusively used the logits corresponding to the first output token, without applying any decoding strategies. Our sparse circuit analysis is governed by two sets of thresholds. For the initial causal tracing to identify circuit components, we used a node threshold (T_N) of 0.073 and an edge threshold (T_E) of 0.007.

The amplification factor λ is calculated from a contrastive dataset. We use this small set as validation dataset to measure how much the label logits difference changes when we intervene on the feature. Let $m(x)$ be the logit margin, defined as the difference between the logits of the target class and the opposing class. We approximate the effect of the steering vector as linear:

$$\mathbb{E}[m_{steered}(x) - m_{original}(x)] \approx \alpha \cdot \lambda$$

We first estimate the sensitivity α using a small probe λ_0 on the validation set. Then, we set λ to achieve a fixed target margin improvement M_{target}:

$$\lambda = \frac{M_{target}}{\alpha}$$

In our experiments, we set $M_{target} = 2.0$. We also cap the intervention strength by using $\min(\lambda, 3)$. Further implementation details, including the specific amplification parameters for our inference-time intervention, can be found in our source code.

B Contrastive Data for SA-Circuit Discovery

To identify the SA-Circuit, we constructed a specialized dataset. This dataset originates from a single base example from the SST-2 dataset. We then derived 15 additional contrastive samples by substituting core aspect and opinion words, resulting in a total of 16 sentences that form 8 contrastive pairs (Negative vs. Positive). The complete dataset used for circuit discovery is detailed in Table 11.

Sensitivity to Contrastive Template Style. We test whether the results depend on the text form used to generate contrastive samples. The contrastive samples are instantiated using the form shown in Table 6. Under a fixed sample budget ($n = 16$), we compare three distinct form and compute pairwise Jaccard similarity of coverage. As shown in Table 8, all form pairs exhibit high

	Text Form
Text-Form-1	The *[NOUN]* are *[ADJECTIVE]* as well.
Text-Form-2	A very *[ADJECTIVE] [NOUN]*.
Text-Form-3	The *[NOUN$_1$]* is that the *[NOUN$_2$]* about *[ENTITY]* increasingly *[ADJECTIVE]*.

Table 6: Text Form used to construct contrastive samples.

	ChnSentiCorp			*Malay*		
	Vanilla	**Ours**	**Δ**	**Vanilla**	**Ours**	**Δ**
T_1	74.34	86.74	+12.40	59.27	61.77	+2.50
T_2	76.75	78.25	+1.50	54.37	66.52	+12.15
T_3	74.34	81.34	+1.50	40.57	51.48	+10.91
T_4	74.89	87.33	+12.43	50.44	67.66	+17.22

Table 7: F1-Scores (%) across four prompt templates on multilingual datasets. Four translated prompt templates were used for evaluation on *Chinese* and *Malay* datasets.

	Form-1	Form-2	Form-3
Form-1	-	92.11	91.89
Form-2	92.11	-	89.47
Form-3	91.89	89.47	-

Table 8: Jaccard similarity (%) of covered contrastive samples across different text form.

similarity, suggesting that the SA-Circuit may not sensitive to the specific syntactic pattern or phrasing used in the contrastive form.

Sensitivity to Contrastive Dataset Size. We further expand the number of contrastive samples from $n = 8$ to $n = 128$ and examine the coverage overlap between different sample sizes using Jaccard similarity on Gemma-2B. As shown in Table 9, the overlap remains high across all size pairs, indicating that increasing the contrastive dataset size does not change the conclusions about the SA-Circuit.

C Additional Experimental Results on Llama-3.1-8B

In this section, we present the experimental results obtained using the Llama-3.1-8B model. To ensure a fair and consistent comparison, we utilized the exact same test samples across all seven datasets as employed in the Gemma-2-2B experiments presented in the main paper. Specifically, Table 7 presents the evaluation results for the two cross-lingual datasets (Chinese and Malay), while Table 10 details the performance on the five English benchmarks.

	Contrastive Dataset Size (n)				
Size (n)	8	**16**	32	64	128
8	–				
16	84.62	–			
32	84.62	100.00	–		
64	86.84	97.44	97.44	–	
128	86.84	97.44	97.44	100.00	–

Table 9: Jaccard similarity (%) of covered contrastive samples across different dataset sizes (n).

D Identified SA-Circuit Features

D.1 Gemma-2-2B Features

Through causal circuit analysis on the contrastive dataset across four prompt templates, we identified a set of 39 critical features that constitute the SA-Circuit for sentiment analysis in Gemma-2-2B. These features are distributed across different layers and components of the model. The details of these features are enumerated in Table 12.

To better understand the function of this circuit, we provide detailed natural language annotations for each of the 39 features in Table 13, derived from the SAE's interpretations.

D.2 Llama-3.1-8B Features

Applying the same causal circuit analysis methodology to the Llama-3.1-8B model, we identified a corresponding set of 47 key features that play a pivotal role in sentiment analysis. These features were selected based on their causal effect on the sentiment prediction task. The specific indices, layers, and component types of these features are listed in Table 14.

Similar to the analysis for Gemma, we utilized the natural language interpretations provided by the Llama Scope SAEs to understand the semantic roles of these features. As shown in Table 15, the identified features in Llama-3.1-8B also cover a range of concepts from direct emotional expressions to more abstract logical and structural components, reinforcing the finding that LLMs employ a complex circuit for sentiment processing.

Methods	SST-2	ABSC-Res	ABSC-Lap	MAMS	IMDB	Average
T_1: Classify sentiment as [negative, positive] for the text: {*text*} Output:						
Vanilla	70.04	89.77	91.14	82.43	73.66	81.41
+CAA (Rimsky et al., 2024)	47.87	56.62	60.25	50.61	46.77	52.43
+LoReft (Wu et al., 2024)	48.78	59.56	57.47	46.36	46.98	51.83
+Reft-R_1 (Wu et al., 2025)	44.74	58.31	57.77	48.63	41.14	50.12
+SAE_A (Rajamanoharan et al., 2024)	70.60	89.86	91.62	81.61	74.25	81.58
+Ours	**79.56**	**92.39**	**95.19**	**83.13**	**78.77**	**85.18**
T_2: Evaluate if the text is negative or positive, Text : {*text*} Eval :						
Vanilla	77.83	89.49	92.33	79.69	79.20	83.71
+CAA (Rimsky et al., 2024)	47.92	55.30	57.24	49.44	47.07	51.39
+LoReft (Wu et al., 2024)	50.61	61.60	60.93	55.73	64.10	58.59
+Reft-R_1 (Wu et al., 2025)	44.75	58.17	59.25	47.83	44.32	50.87
+SAE_A (Rajamanoharan et al., 2024)	78.10	89.53	92.78	79.41	80.19	84.00
+Ours	**79.79**	**92.35**	**93.64**	**81.52**	**88.10**	**87.08**
T_3: Is the sentiment for the following text positive or negative ? Text : {*text*} Answer : it is						
Vanilla	40.59	84.09	76.10	77.34	89.19	73.46
+CAA (Rimsky et al., 2024)	47.90	61.02	62.17	50.85	46.90	53.77
+LoReft (Wu et al., 2024)	38.94	53.33	47.85	44.00	45.44	45.91
+Reft-R_1 (Wu et al., 2025)	35.63	56.76	54.73	47.03	43.76	47.58
+SAE_A (Rajamanoharan et al., 2024)	42.37	84.93	79.61	77.10	89.38	74.68
+Ours	**84.03**	**91.85**	**93.08**	**78.01**	**89.90**	**87.35**
T_4: Please perform Sentiment Classification task. Given the sentence, assign a sentiment label from [negative, positive]. Return label only without any other text. Sentence: {*text*} Label:						
Vanilla	68.10	94.79	95.16	83.56	84.20	85.15
+CAA (Rimsky et al., 2024)	48.39	55.38	57.79	49.67	47.02	51.65
+LoReft (Wu et al., 2024)	42.86	56.47	52.70	46.08	47.32	49.08
+Reft-R_1 (Wu et al., 2025)	44.13	59.12	59.78	47.61	43.47	50.82
+SAE_A (Rajamanoharan et al., 2024)	70.32	94.50	94.91	83.34	85.29	85.67
+Ours	**88.40**	**95.15**	**96.34**	**84.01**	**91.99**	**91.18**

Table 10: Main results evaluated by F1-Score (%) on Llama-3.1-8B. All methods are evaluated in the zero-shot setting. The templates shown are at the sentence level; for aspect-level datasets, the corresponding aspect is appended after {*text*}. The SA-Circuit used in Ours is derived solely from SST-2.

Sentence	Label
the truth is that the truth about charlie gets increasingly tiresome .	negative
the truth is that the truth about charlie gets increasingly fascinating .	positive
the story is that the story about alice gets increasingly dull .	negative
the story is that the story about alice gets increasingly exciting .	positive
the message is that the message about bob gets increasingly boring .	negative
the message is that the message about bob gets increasingly delightful .	positive
the concept is that the concept about dave gets increasingly tedious .	negative
the concept is that the concept about dave gets increasingly inspiring .	positive
the idea is that the idea about emma gets increasingly depressing .	negative
the idea is that the idea about emma gets increasingly uplifting .	positive
the theory is that the theory about frank gets increasingly horrible .	negative
the theory is that the theory about frank gets increasingly wonderful .	positive
the tale is that the tale about grace gets increasingly annoying .	negative
the tale is that the tale about grace gets increasingly charming .	positive
the rumor is that the rumor about harry gets increasingly dreary .	negative
the rumor is that the rumor about harry gets increasingly enjoyable .	positive

Table 11: The 16 contrastive sentences (8 pairs) used for discovering the SA-Circuit. The data is presented in pairs, with each pair separated by a dashed line, to highlight the minimal sentiment shift between them.

Layer	Component Type	Feature Indices
14	Residual Stream	10781
15	Residual Stream	6237, 15289
16	Attention Head Output	1573, 15716
16	Residual Stream	9909, 12800
18	Residual Stream	1720, 8943
19	Residual Stream	11836, 13228
20	MLP Output	7457, 15509
20	Residual Stream	7093, 5606, 285, 933
21	Residual Stream	2355, 3533, 4502, 2182
22	Residual Stream	8856, 10433, 15238, 8161, 10572, 508, 3502
23	MLP Output	6192, 8332
23	Residual Stream	10969, 4037
24	Residual Stream	9202
25	MLP Output	1512
25	Residual Stream	8950, 12858, 14247, 5856, 2772

Table 12: The 39 features comprising the SA-Circuit for Gemma-2-2B. Each feature is identified by its layer, component type (Attention, MLP, or Residual Stream), and index.

Feature Identifier	Annotation
resid_14/10781	concepts related to emotional impact and its influence on behavior and action
resid_15/6237	complex relationships and dynamics in human interactions and emotions within various contexts
resid_15/15289	words and phrases related to success and failure, particularly in the context of choices and their impacts
attn_16/1573	attends to the token "success" from corresponding tokens that indicate a successful action or operation
attn_16/15716	attends to negative or distrust tokens from positive or general human-related tokens
resid_16/9909	concepts related to consequences, particularly in terms of rewards and punishments
resid_16/12800	terms related to scientific measurements and results
resid_18/1720	terms related to neuroanxiety and positive emotional expressions
resid_18/8943	expressions of anger and frustration
resid_19/11836	terms and phrases related to iron levels and their impact on biological processes
resid_19/13228	questions and comparisons related to numerical values
mlp_20/7457	numerical values and mathematical operations in a technical context
mlp_20/15509	indicators of scientific measurements and data
resid_20/285	numerical values and mathematical operations
resid_20/933	formatted data structures and their parameters or values
resid_20/5606	mathematical equations and expressions
resid_20/7093	expressions of admiration and appreciation
resid_21/2182	mathematical expressions and scientific notations
resid_21/2355	references to praise or positive acknowledgment
resid_21/3533	phrases and terms related to feedback and critique
resid_21/4502	technical terms and symbols used in coding or mathematical contexts
resid_22/508	mathematical expressions and notations
resid_22/3502	terms related to scientific measurements and chemical compounds
resid_22/8161	concepts related to emotional and mental health management
resid_22/8856	instances of negative commentary or criticism in context
resid_22/10433	evaluative language and expressions of opinion about individuals or works
resid_22/10572	expressions of disappointment and emotional reactions
resid_22/15238	negative sentiments or conditions related to health, quality, and personal experiences
mlp_23/6192	discussions centered around measurements and comparisons in scientific contexts
mlp_23/8332	references to community and shared experiences, particularly related to weddings
resid_23/4037	references to mathematical functions, variables, and operations related to equations or formulations
resid_23/10969	emotional expressions related to disappointment and regret
resid_24/9202	terms relating to the statistical representation of associations in data
mlp_25/1512	mentions of various emotional or evaluative descriptors
resid_25/2772	concepts related to spirituality and the significance of religious beliefs and practices
resid_25/5856	phrases related to negative qualities or criticisms regarding people
resid_25/8950	concepts related to monotony and boredom
resid_25/12858	phrases related to emotions and psychological concepts
resid_25/14247	references to critical analysis and critique of ideas or actions

Table 13: Detailed interpretations for the 39 features identified as part of the SA-Circuit in Gemma-2-2B.

Layer	Component Type	Feature Indices
14	Residual Stream	3356, 28823
15	Residual Stream	16615, 9338
16	Residual Stream	10516, 17060
17	Residual Stream	6702
18	Residual Stream	11534, 17623
19	Residual Stream	30515, 4052
20	Residual Stream	6482
21	Residual Stream	23870, 23280, 20851
22	Residual Stream	6100, 3662, 7
23	Residual Stream	10359, 32725, 6811
24	Residual Stream	1346, 9342
25	Residual Stream	15484, 20072
26	Residual Stream	26056, 4468, 4743
27	MLP Output	1631, 800, 15679
27	Residual Stream	23498, 23763, 31180
28	MLP Output	2771
28	Residual Stream	20332, 12874, 5581
29	Residual Stream	19169, 2902, 26650, 5514
30	MLP Output	14481
30	Residual Stream	12879, 23511
31	Residual Stream	17299, 1878

Table 14: The 47 features comprising the SA-Circuit for Llama-3.1-8B. Each feature is identified by its layer, component type, and index.

Feature Identifier	Annotation
resid_14/3356	phrases that highlight the complexities of human relationships and the impact of words
resid_14/28823	comparisons between emotional highs and lows
resid_15/16615	elements related to storytelling and themes of fame and tragedy in Hollywood
resid_15/9338	elements related to rewards, consequences, and the emotional impact of decision-making
resid_16/10516	themes of nostalgia and contrasting experiences in storytelling
resid_16/17060	references to environmental sustainability and the contrasting forces affecting it
resid_17/6702	phrases that highlight contrasts between praise and blame
resid_18/11534	references to emotional experiences and the complexities of life
resid_18/17623	common phrases related to credit, blame, and criticism within various contexts
resid_19/30515	complex relationships and contrasts in actions or states
resid_19/4052	phrases that indicate emotional disappointment or feelings of loss
resid_20/6482	contrast between positive and negative experiences or perceptions
resid_21/23870	numerical data and statistical references related to performance and metrics
resid_21/23280	phrases that involve comparisons and contrasts
resid_21/20851	phrases related to positive and negative experiences or evaluations
resid_22/6100	positive emotional descriptors and experiences
resid_22/3662	numerical data related to performance metrics and statistical measures
resid_22/7	references to statistical performance metrics or values
resid_23/10359	elements of programming logic and performance metrics
resid_23/32725	financial metrics and performance indicators
resid_23/6811	complex relationships and dynamics involving positives and negatives
resid_24/1346	metrics and statistical analysis related to performance and ratings
resid_24/9342	expressions of happiness and positive emotions in the context of music
resid_25/15484	phrases related to numeric data and measurements, particularly in scientific and economic contexts
resid_25/20072	phrases related to the evaluation of advantages and disadvantages
resid_26/26056	concepts related to morality and ethical implications of actions
resid_26/4468	negative sentiments towards movies and their quality aspects
resid_26/4743	conjunctions and phrases that contrast positive and negative aspects or experiences
mlp_27/1631	expressions of contrasting viewpoints or feelings
mlp_27/800	terms and phrases related to medical testing and diagnostic results
mlp_27/15679	instances of negative consequences and actions related to improper practices
resid_27/23498	mathematical symbols and formatting elements in coding or mathematical expressions
resid_27/23763	themes related to mythological symbolism and their interpretations
resid_27/31180	terms related to medical testing and diagnoses
mlp_28/2771	mathematical concepts and their applications in various contexts
resid_28/20332	economic disparities and the impacts of socioeconomic status on households
resid_28/12874	phrases related to customer satisfaction and positive experiences
resid_28/5581	technical terms and identifiers related to programming and web development
resid_29/19169	references to financial performance metrics and comparisons
resid_29/2902	phrases related to customer service and positive experiences
resid_29/26650	expressions related to emotions and mood, particularly those that convey happiness and liveliness
resid_29/5514	detailed descriptions of technical processes and elements in programming or medical contexts
mlp_30/14481	mathematical expressions and syntax in code-like structures
resid_30/12879	negative evaluations of experiences or services
resid_30/23511	negative sentiments related to customer service experiences and product reliability
resid_31/17299	phrases related to the effects of positivity and negativity in human interactions and perceptions
resid_31/1878	vocabulary related to morality and ethical concepts

Table 15: Detailed interpretations for the 47 features identified as part of the SA-Circuit in Llama-3.1-8B.

E Visualization of Features Activations

In this section, we provide the full activation heatmaps for the identified SA-Circuit features across the four different prompt templates ($T_1 \sim T_4$)and a baseline setting without any instruction.

Figures 7 through 11 illustrate how the activation patterns shift depending on the instruction phrasing. The x-axis represents the token sequence, and the y-axis represents the sparse features. Darker red indicates higher activation.

It can be observed that while the activation of features corresponding to the text content remains strong and stable across all figures, the activation patterns for the instruction tokens vary significantly.

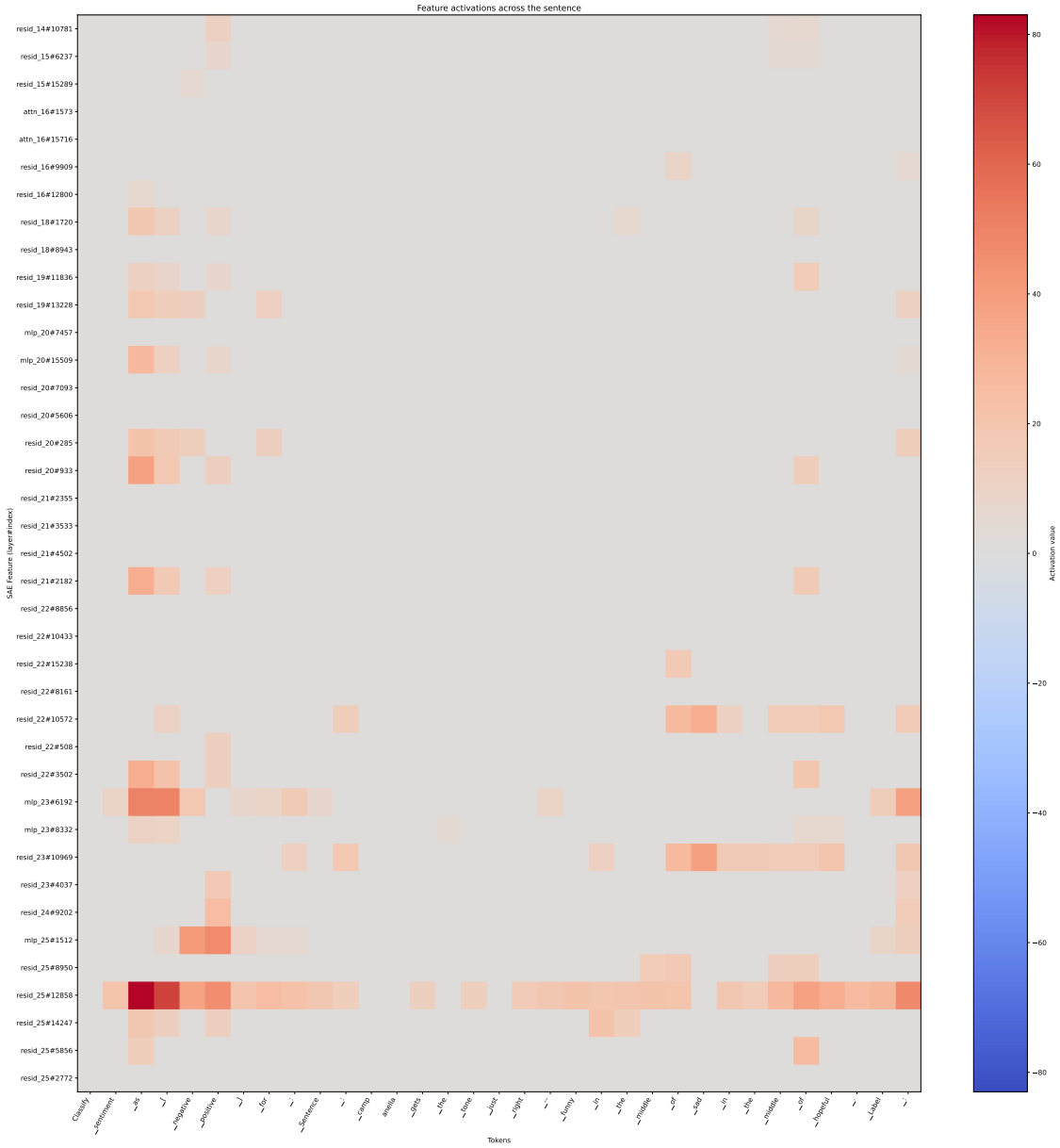

Figure 7: Feature activation heatmap for Template T_1.

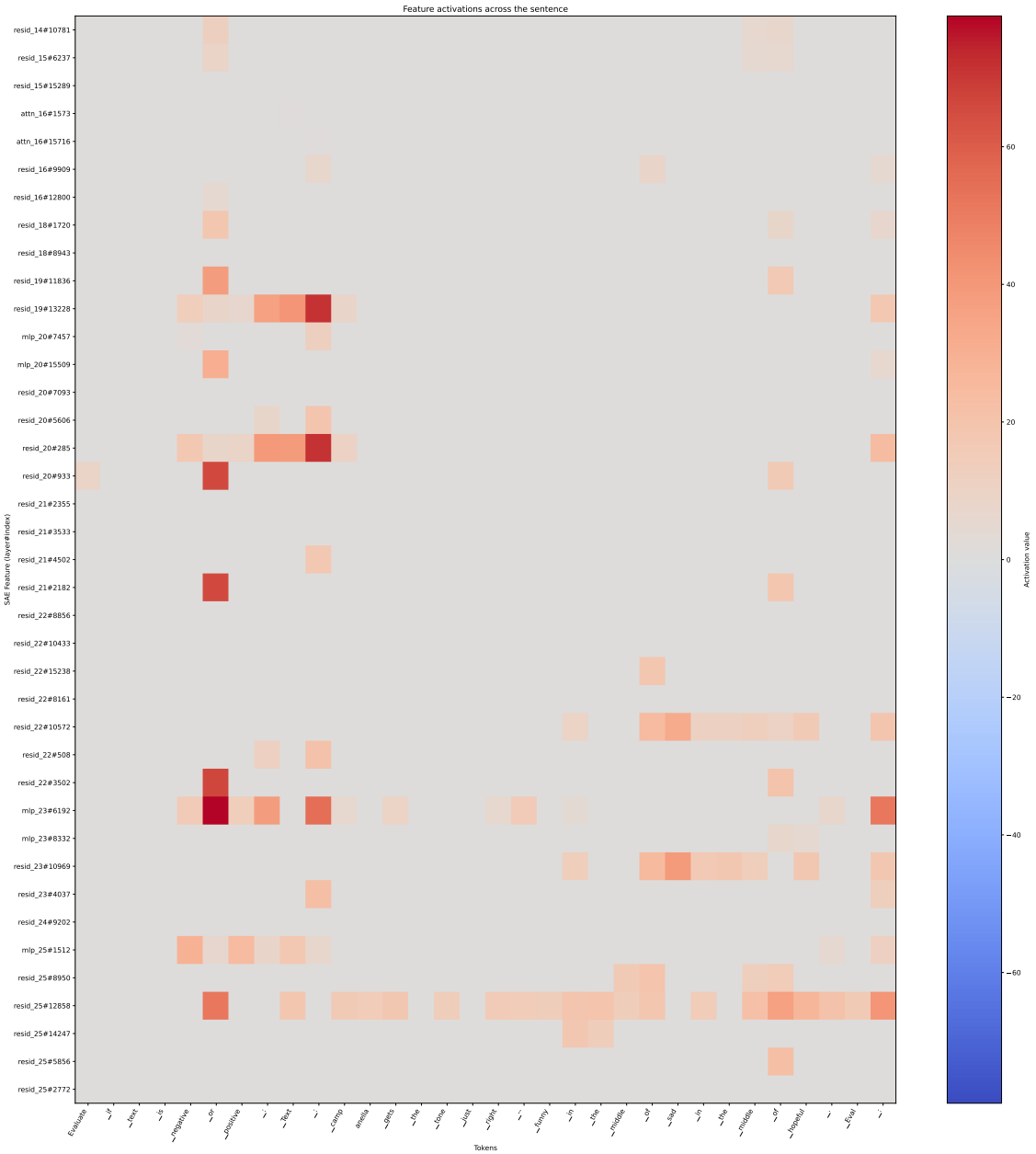

Figure 8: Feature activation heatmap for Template T_2.

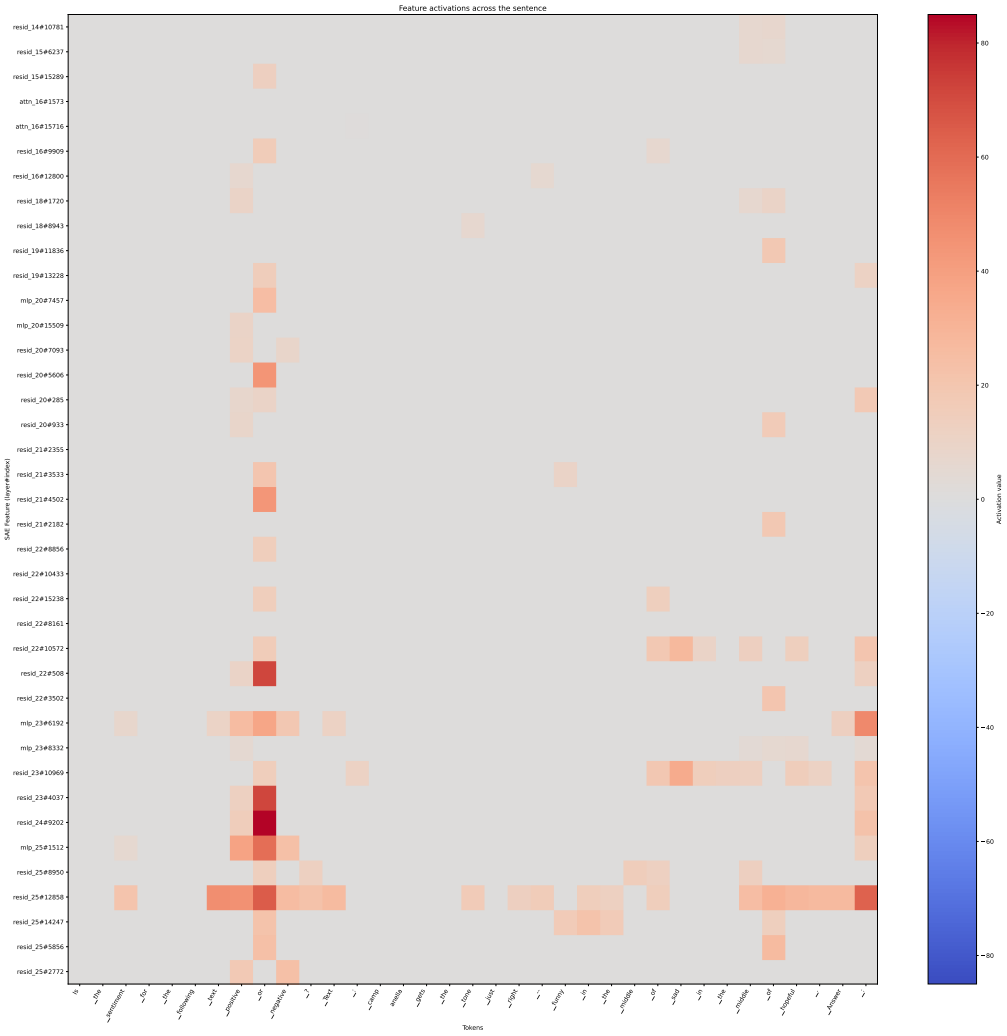

Figure 9: Feature activation heatmap for Template T_3.

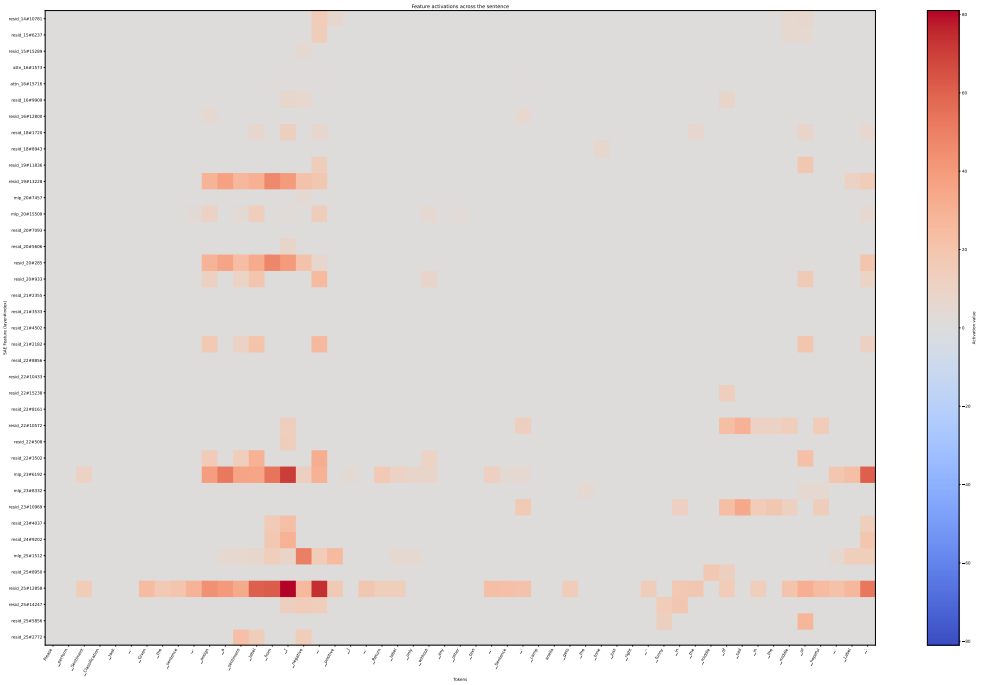

Figure 10: Feature activation heatmap for Template T_4.

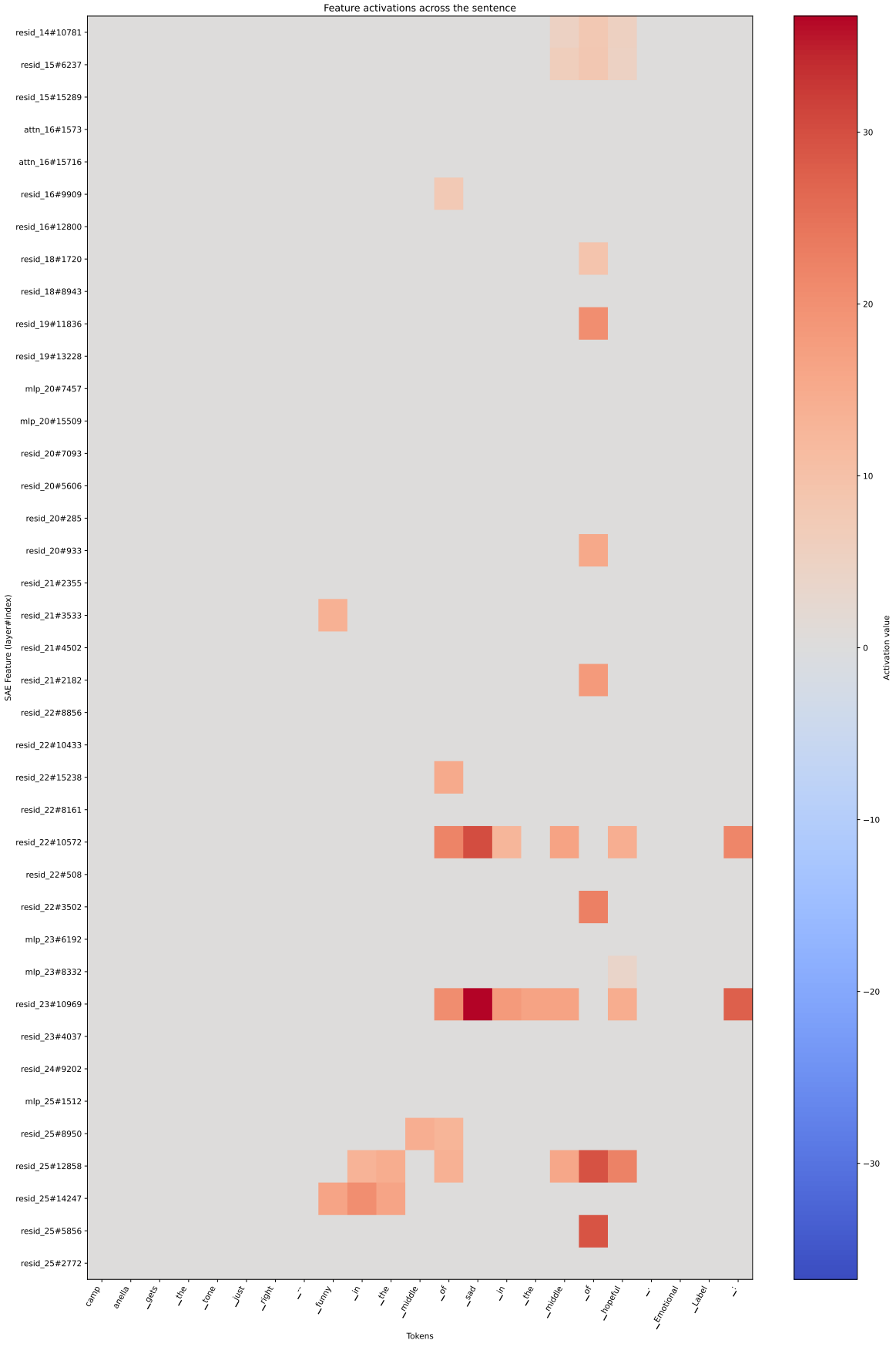

Figure 11: Feature activation heatmap without any instruction.