

Exploring Model Editing for LLM-based Aspect-Based Sentiment Classification

Shichen Li¹ Zhongqing Wang^{1*}, Zheyu Zhao¹, Yue Zhang², Peifeng Li¹

¹Natural Language Processing Lab, Soochow University, Suzhou, China

²Westlake University

{scli06,zyzhao0104}@stu.suda.edu.cn, {zqwang, pfl}@suda.edu.cn
zhangyue@westlake.edu.cn

Abstract

Model editing aims at selectively updating a small subset of a neural model’s parameters with an interpretable strategy to achieve desired modifications. It can significantly reduce computational costs to adapt to large language models (LLMs). Given its ability to precisely target critical components within LLMs, model editing shows great potential for efficient fine-tuning applications. In this work, we investigate model editing to serve an efficient method for adapting LLMs to solve aspect-based sentiment classification. Through causal interventions, we trace and determine which neuron hidden states are essential for the model’s predictions. By performing interventions and restorations on each component of an LLM, we identify the importance of these components for aspect-based sentiment classification. Our findings reveal that a distinct set of mid-layer representations is essential for detecting the sentiment polarity of given aspect words. Leveraging these insights, we develop a model editing approach that focuses exclusively on these critical parts of the LLM, leading to a more efficient method for adapting LLMs. Our in-domain and out-of-domain experiments demonstrate that this approach achieves competitive results compared to the currently strongest methods with significantly fewer trainable parameters, highlighting a more efficient and interpretable fine-tuning strategy.

Introduction

Aspect-based sentiment classification (Hu and Liu 2004) aims to determine the sentiment polarity of the aspects within a sentence. Traditional ABSC methods (Tang et al. 2016; Tay, Tuan, and Hui 2018; Zhang, Zhou, and Wang 2022) have often relied on smaller, task-specific language models which frequently fall short in capturing complex contextual dependencies and diverse expressions of sentiment.

As language models continue to grow in size, adapting these large language models (Touvron et al. 2023; Ouyang et al. 2022) to ABSC through full-parameter fine-tuning has become increasingly challenging. Various parameter-efficient fine-tuning methods (Hu et al. 2022; Houlsby et al. 2019; Li and Liang 2021; Liu et al. 2024) have been developed to address the inefficiencies of full-parameter fine-

*Corresponding author.

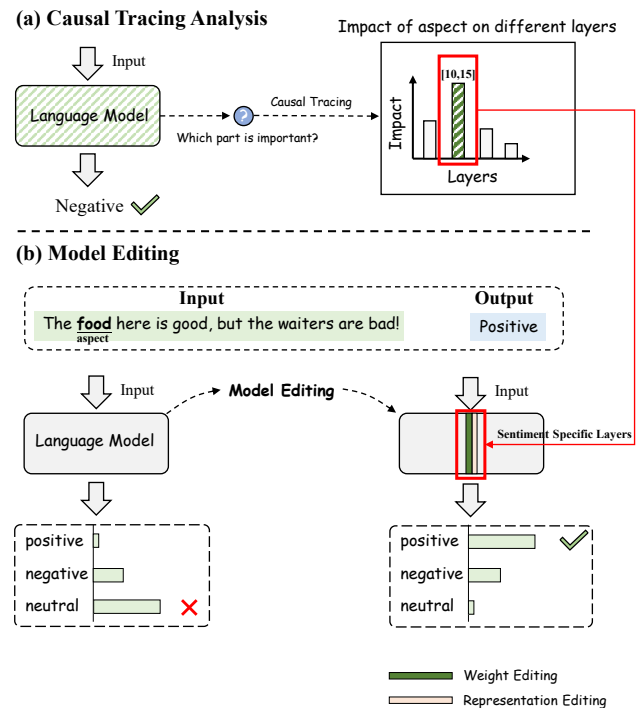


Figure 1: An illustration of our 2-steps model editing method for LLM-based aspect-based sentiment classification.

tuning and shown strong performance in ABSC (Li et al. 2022), but these methods still need to update across broad sections of the model’s architecture, potentially leading to redundancies and inefficiencies.

Model editing (Mitchell et al. 2022; Meng et al. 2022; Dai et al. 2022) gives a promising way to further reduce the cost by selectively updating a small subset of a neural model’s parameters with an interpretable strategy and achieve desired modifications.

Recent works in model editing (Meng et al. 2023; Mitchell et al. 2022; Meng et al. 2022; Wu et al. 2024) has shown effectiveness and efficiency in various tasks by modifying LLMs with minimal changes to their parameters. These methods highlight that targeted updates to specific

neural network components can achieve precise modifications, thereby avoiding the extensive resource requirements of full-parameter fine-tuning. However, the research question of how to edit LLMs for ABSC is still under-explored.

Intuitively, aspect words significantly influence the model’s output, allowing us to perform precise causal analysis and examine how changes to specific components affect overall performance. This enables us to accurately identify and modify the model parameters closely associated with sentiment classification, thereby validating the potential of model editing in enhancing model adaptability and efficiency.

Given the above observations, we consider a model editing method to improve parameter-efficient fine-tuning for ABSC. The method consists of two steps as shown in Figure 1. As shown in Figure 1(a), we first employ causal interventions to identify which components are critical for the LLM’s predictions associated with sentiment. By performing interventions and restorations on each layer of the model, we determine their relative importance for the ABSC task. Experiments reveal that the mid-layer representations of specific aspect words in certain positions are particularly critical for accurately identifying the sentiment polarity. Based on the insights, we propose a precise model editing method that targets critical parts of the LLM using two model editing approaches: weight-based and representation-based editing, as shown in Figure 1(b). For weight-based editing (Hu et al. 2022), we use low-rank adaptation to edit specific model weights. For representation-based editing (Wu et al. 2024), we manipulate a small fraction of model representations at specific positions of aspect terms to steer the model’s behavior.

We conduct extensive experiments to evaluate the method in both in-domain and out-of-domain scenarios, aiming to validate its effectiveness and generalization. The experiments confirm that our method achieves competitive results with a minimal number of parameters in both scenarios. Additionally, we empirically investigate the impact of the proposed model editing method on specific layers and analyze the influence of modifications at specific positions of different words. The results show that the mid-layer representations of specific aspect words are crucial for accurately obtaining the correct predictions, aligning with the findings of causal tracing results.

The main contributions of this work can be summarized as follows:

- We propose a novel framework that combines causal tracing and model editing to address specific downstream tasks with LLMs.
- We develop a more parameter-efficient fine-tuning method specifically tailored for the aspect-based sentiment classification task.
- The proposed method achieves competitive results compared to state-of-the-art PEFT methods with significantly fewer trainable parameters, highlighting a more efficient and interpretable fine-tuning strategy.

Related Work

In this section, we introduce two related topics of this study: aspect-based sentiment classification and model editing.

Aspect-Based Sentiment Classification

Aspect-based sentiment classification methods have seen significant changes with the development of neural networks (Zhang et al. 2023). Early approaches often relied on full-parameter fine-tuning of smaller models such as LSTM and CNN with attention mechanisms to solve ABSC task (Tang et al. 2016; Ma et al. 2017; Huang and Carley 2018; Xue and Li 2018). However, these models had limitations in understanding complex contexts and long-range dependencies. The development of pre-trained language models brought significant improvement (Sun, Huang, and Qiu 2019; Xu et al. 2019; Jiang et al. 2020). By fine-tuning these models on ABSC tasks, researchers were able to better capture the context and improve ABSC performance. However, as models grew larger, the computational cost and resource demands of full-parameter fine-tuning became increasingly prohibitive. To address these challenges, parameter-efficient fine-tuning methods (Hu et al. 2022; Liu et al. 2024) emerged. These methods enable the adaptation of LLMs to ABSC tasks by tuning a smaller subset of parameters. Due to the strong foundational capabilities of LLMs, parameter-efficient fine-tuning methods can achieve competitive performance compared to previous carefully designed methods (Li et al. 2022). While these methods only update a small portion of parameters within each layer, the updates still span the entire model, leading to some unnecessary redundancies.

Model Editing

With the development of large language models, full-parameters fine-tuning has become challenging. To alleviate this inefficiency, model editing (Meng et al. 2022, 2023) is designed to update these models in a highly targeted and precise manner. Recent advancements in model editing (Yao et al. 2023) can be categorized into two types: knowledge integration and parameter modification.

Knowledge integration aims to augment the model with new information without updating its original parameters. Recent works on activation steering (Han et al. 2024; Turner et al. 2023; Avitan et al. 2024) and representation engineering (Zou et al. 2023; Liu, Xing, and Zou 2023) shows that adding fixed or task-specific steering vectors to the model enable precise control over outputs of LMs.

Parameter modification focuses on directly adjusting the model’s parameters. Recent works focus on updating specific parts of the model responsible for particular knowledge. Techniques like Knowledge Neurons (Dai et al. 2022) and ROME (Meng et al. 2022) identify and edit specific neurons or layers with a designed causal tracing approach. These work make an assumption that the feed-forward network stores knowledge in a key-value format, allowing for directly targeted parameter adjustments to update or add memories. Another line of work (Wu et al. 2024) focuses on training interventions to directly edit model representation during inference for specific tasks.

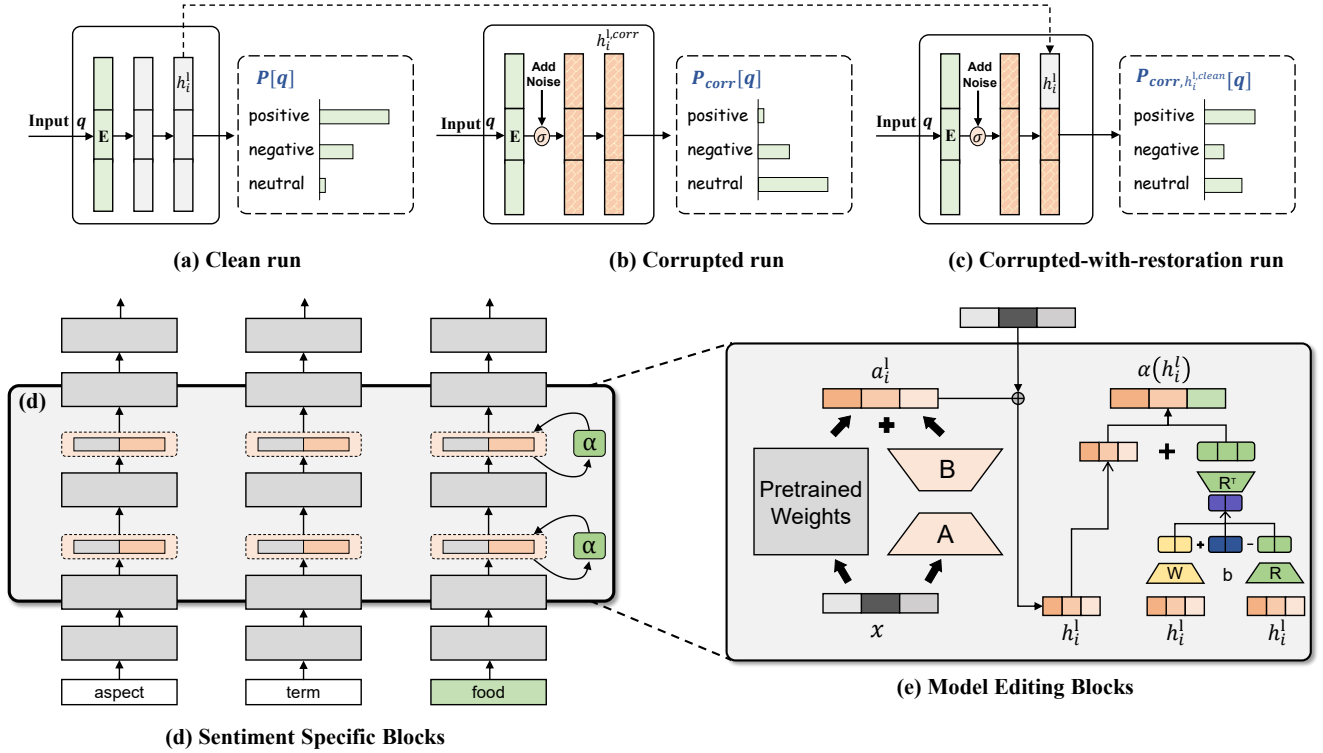


Figure 2: Illustration of the process of causal tracing and model editing for ABSC. (a), (b), (c) illustrate three runs for tracing sentiment associations; (d) and (e) demonstrate our method, where colored blocks represent active parameters and the grey blocks represent frozen parameters. Notably, representation-based editing is applied only to the position of the aspect word.

Method

In this section, we first introduce the intervention method for tracing sentiment associations, then analyze the causal tracing results. Finally, based on the insights of the analysis, we explore model editing as an efficient adaptation method using task-specific model edits.

Interventions for Tracing Sentiment Association

We begin by identifying the specific layer with the strongest causal effect on sentiment polarity predictions of the given aspect terms by the method proposed in (Meng et al. 2022). This process is inspired by the causal tracing technique in (Vig et al. 2020; Pearl 2022; Geiger et al. 2021), which demonstrates how modifications to hidden states can impact the network’s output. By systematically adding noise and restoring representations, we can evaluate the sensitivity of the model’s predictions to changes in certain components.

We present each sample as a tuple $t = (S, A, P)$ containing the sentence S , the aspect A , and the sentiment polarity P according to A . Then, we provide a natural language prompt q describing (S, A) and examine the model’s prediction of q . To identify the layers with the strongest causal effect, we perform interventions on the output of each layer. Specifically, we introduce the following three runs:

Clean run. We input the sentence S and aspect A described by q into the model, and observe the prediction

P , as shown in Figure 2(a). All hidden representations $\{h_i^{(l)}\}_{i \in [1, T], l \in [1, L]}$ are recorded, where T is the number of tokens and L is the number of layers.

Corrupted run. As shown in Figure 2(b), we add noise σ to the hidden states at embedding layers denoted as $\{h_i^{(0)} + \sigma\}_{i \in [1, T]}$. It results in corrupted representations $\{h_i^{(l), \text{corr}}\}_{i \in [1, T], l \in [1, L]}$. This noise follows the prior practice (Meng et al. 2022). The corrupted representations lead to a potentially incorrect sentiment polarity prediction P^{corr} .

Corrupted-with-restoration run. As shown in Figure 2(c), we selectively restore the clean hidden representation $h_i^{(l), \text{clean}}$ at certain token and layer while keeping the rest of the representation corrupted. This run tests if the model can correctly predict the sentiment polarity P despite the overall corruption, indicating the causal importance of the restored hidden states.

Let $\mathbb{P}[q]$, $\mathbb{P}_{\text{corr}}[q]$, and $\mathbb{P}_{\text{corr}, h_i^{(l), \text{clean}}}[q]$ denote the probabilities of predicting P under the clean, corrupted, and corrupted-with-restoration runs, respectively. The causal effect is quantified as follows:

The total effect (TE) is defined as the difference in prediction probabilities between the clean and corrupted runs:

$$\text{TE} = \mathbb{P}[q] - \mathbb{P}_{\text{corr}}[q] \quad (1)$$

The indirect effect (IE) is defined as the difference be-

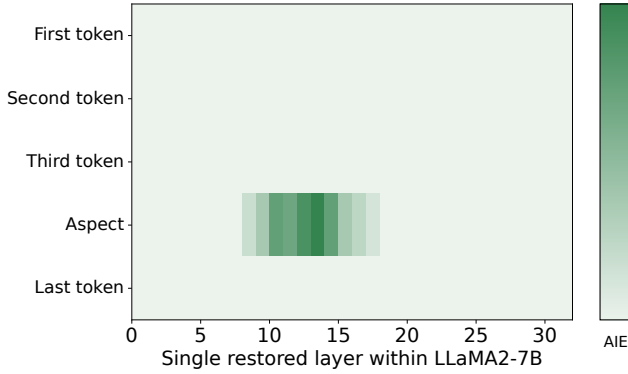


Figure 3: Causal tracing results for ABSC at various words, layers, and positions. The "First token," "Second token," and so on denote the positions of the input sentence tokens.

tween the probability of correct prediction under the corrupted run and the corrupted-with-restoration run for a specific layer l and position i :

$$\text{IE}_i^{(l)} = \mathbb{P}_{\text{corr}, h_i^{(l)}, \text{clean}}[q] - \mathbb{P}_{\text{corr}}[q]$$

The average total effect (ATE) is the mean total effect across multiple samples:

$$\text{ATE} = \frac{1}{N} \sum_{i=1}^N \text{TE}_i \quad (2)$$

The average indirect effect (AIE) calculates the mean indirect effect for each hidden state variable $h_i^{(l)}$:

$$\text{AIE}^{(l)} = \frac{1}{N} \sum_{i=1}^N \text{IE}_i^{(l)} \quad (3)$$

By analyzing these effects, we can identify the layers and specific hidden states within the layers that have the most significant causal impact on sentiment polarity predictions.

Causal Tracing Results

We begin by adapting the LLM¹ to solve the ABSC task. Subsequently, we compute the average indirect effect over 400 samples from the ABSC dataset.

Since the model inevitably makes errors, we use the ATE to discard outliers to ensure an accurate interpretation of the causal tracing results. Specifically, we retain only the results that met the expected prediction: $\mathbb{P}[q]$ for correct prediction and $\mathbb{P}_{\text{corr}}[q]$ for incorrect prediction.

As illustrated in Figure 3, our exploration focuses on the influence of tokens at different positions and layers within the model. Our findings reveal that a notable portion of the effect is mediated by highly causal individual states, particularly at the aspect tokens.

While it is expected that aspect tokens play a decisive role in predicting sentiment polarity, the causal tracing results explicitly demonstrate that the mid-layer representations of aspect tokens play a significant role in ABSC. The emergence

¹<https://huggingface.co/meta-llama/Llama-2-7b-hf>

of these causal states in the mid-layers sheds light on more efficient parameter-efficient fine-tuning methods for ABSC.

Model Editing for LLM-based ABSC

Given the importance of mid-layer representations of aspect tokens for ABSC, we propose that employing model editing in only these representations can guide the model to solve ABSC with significantly fewer parameters. Our method leverages the strengths of both weight-based (Hu et al. 2022) and representation-based editing (Wu et al. 2024) to achieve this goal.

Recall that in each layer of the decoder model, the representation is updated through a specific process. The hidden state \mathbf{h}_l at layer l is derived by incorporating the previous layer's representation \mathbf{h}_{l-1} , the output from the attention mechanism, and the output from a multi-layer perceptron (MLP). Formally, this can be expressed as:

$$\mathbf{h}_l = \mathbf{h}_{l-1} + a(\mathbf{h}_{l-1}) + \text{MLP}(\mathbf{h}_{l-1} + a(\mathbf{h}_{l-1})), \quad (4)$$

where $a(\cdot)$ denotes the attention mechanism and $\text{MLP}(\cdot)$ denotes the multi-layer perceptron.

Weight-based Editing Previous works (Tang et al. 2020; Wang et al. 2020; Zhang and Qian 2020) have demonstrated that the attention mechanism is crucial for capturing relationships between tokens, especially aspect tokens. Therefore, we only employ weight-based model editing on the attention output projection matrix, which is responsible for transforming these multi-head attention outputs back into the model's hidden space, directly affecting how aspect token relationships are encoded. By applying weight-based editing specifically to this matrix, we can edit the model to handle aspect tokens with minimal disruption to other layers, ensuring parameter efficiency while maintaining strong performance.

Specifically, we update the weight matrix \mathbf{W}_l of the targeted layers with weight-based editing (Hu et al. 2022) using low-rank matrix \mathbf{A}_l and \mathbf{B}_l .

Let $\mathbf{W}_l \in \mathbb{R}^{d \times k}$ represent the attention layer output projection matrix of the l -th layer in the LLM. We introduce low-rank matrices $\mathbf{A}_l \in \mathbb{R}^{d \times r}$ and $\mathbf{B}_l \in \mathbb{R}^{r \times k}$, such that the weight update can be approximated as:

$$\Delta \mathbf{W}_l \approx \mathbf{A}_l \mathbf{B}_l \quad (5)$$

The adapted weight matrix \mathbf{W}'_l is then:

$$\mathbf{W}'_l = \mathbf{W}_l + \Delta \mathbf{W}_l = \mathbf{W}_l + \mathbf{A}_l \mathbf{B}_l \quad (6)$$

Representation-based Editing Previous works (Ravfogel et al. 2022; Avitan et al. 2024; Singh et al. 2024) demonstrate that editing the residual stream is crucial for controlling pre-trained LM outputs without intensive finetuning. Therefore, we focus on editing the residual stream representation to activate important components within the parameters of the LLM for ABSC.

For the representation editing, we apply linear representation editing using projection matrix \mathbf{R}_l and linear interventions \mathbf{W}_l^* . Specifically, we define a low-rank projection matrix $\mathbf{R}_l \in \mathbb{R}^{r \times d}$ with orthonormal rows. The modified

Dataset	Split	Number of Reviews
Device	Train	1,394
	Test	691
Laptop	Train	2,297
	Test	631
Restaurant	Train	4,284
	Test	2,252
Service	Train	1,840
	Test	886

Table 1: Distribution of reviews across different domains.

hidden representation \mathbf{h}_l^t at layer l for an aspect term at position t is given by:

$$\alpha(\mathbf{h}_l^t) = \mathbf{h}_l^t + \mathbf{R}_l^\top (\mathbf{W}_l^* \mathbf{h}_l^t + \mathbf{b}_l - \mathbf{R}_l \mathbf{h}_l^t), \quad (7)$$

where $\mathbf{W}_l^* \in \mathbb{R}^{r \times d}$ is a linear projection matrix and \mathbf{b}_l is a bias vector.

Thus, as shown in Figure 2(e), the overall optimization objective during model editing for ABSC with our method is to minimize the loss \mathcal{L} with respect to the model editing parameters of $\theta = \{\mathbf{R}_l, \mathbf{W}_l^*, \mathbf{A}_l, \mathbf{B}_l, \mathbf{b}_l\}_{l \in L^*}$,

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,y) \sim \mathcal{D}} \left[- \sum_{t=1}^{|y|} \log (f_{\Phi_0 + \Delta \Phi(\theta)}(y_t | x, y_{<t})) \right], \quad (8)$$

where L^* denotes the specific training layers.

Notably, the task specific model editing parameter $|\theta|$ is much smaller than the LLM parameters $|\Phi_0|$. Therefore, our method allows the model to be efficiently edited at critical layers and positions, thereby improving performance on ABSC with significantly fewer parameters.

Experiment

In this section, we introduce our experimental setup and implementation details, present our method’s performance on in-domain and out-of-domain ABSC tasks compared to competitive baselines, and empirically analyze the effectiveness of our method.

Setup

The labeled dataset used in our experiments includes reviews from four different domains: Restaurant (R), Laptop (L), Device (D), and Service (S). Restaurant (R) is a combination of the restaurant reviews from SemEval 2014/2015/2016 (Pontiki et al. 2014, 2015, 2016). Laptop (L) is sourced from SemEval 2014 (Pontiki et al. 2014). Device (D) consists of all the digital device reviews collected by Toprak, Jakob, and Gurevych (2010). Service (S) contains reviews from web services introduced by Hu and Liu (2004). The distribution of reviews in these domains is detailed in Table 1.

We employ Llama-2-7b (Touvron et al. 2023) as our primary base large language model. AdamW (Loshchilov and

Hutter 2018) is used as the optimizer, with a learning rate of 3×10^{-4} for the low-rank weight projection part and 1×10^{-5} for the representation editing part. For the comparison methods, we adopt standard experimental settings and commonly used parameters. Specifically, LoRA and Dora utilize a learning rate of 1×10^{-4} with rank of 32. Additionally, we include LoReft with a learning rate of 2×10^{-5} with rank of 8. All comparison experiments are conducted on a single NVIDIA 3090 GPU and we take accuracy as the evaluation metric. The experimental results are obtained by averaging three runs with random initialization. The PEFT methods are trained for one epoch, while the full parameter methods are trained for three epochs.

The causal tracing results reveal that the representations within mid-layers, specifically layers 10-15, show a decisive influence on ABSC. Therefore, in the ‘specific-layer finetuning’ section of our experimental results, all methods are restricted to these specific layers. Instead, in the section of parameter-efficient finetuning, these methods are trained on all layers.

Main Results

To assess the effectiveness of the proposed method, we mainly conduct two experiments: in-domain and out-of-domain evaluations. We compare our method with zero-shot methods (i.e., GPT-4o-mini, Llama2-7b), full-parameter finetuning methods (i.e., BERT, Deberta, Flan-T5), and parameter-efficient finetuning methods (i.e., PrefixFT, Adapter, LoRA, Dora, Loreft).

In-domain Analysis In-domain ABSC tasks refer to the task that the training data and testing data are in the same domain. This allows us to assess how well the methods perform on data similar to what they were trained on.

As shown in Table 2, our methods achieve the highest average accuracy, significantly outperforming other methods with fewer trainable parameters. Llama2-7b with zero-shot manner exhibits significantly lower performance compared to other method, indicating its limited effectiveness in the absence of domain-specific fine-tuning. The full-parameter finetuning models show improved performance. This suggests that full-parameter finetuning can effectively enhance model performance by leveraging domain-specific data. Among the PEFT methods, PrefixFT, despite its small tuning parameter size, shows only marginal improvement over zero-shot Llama2-7b. LoRA and Dora exhibit considerably better performance. However, these methods still involve larger tuning parameter sizes compared to our proposed methods. LoReft, while efficient in terms of parameter size, does not perform as well as our methods. These results indicate our method not only achieves high accuracy but also exhibits remarkable parameter efficiency.

Out-of-domain Analysis Due to the high risk of catastrophic forgetting in LLM-based in-domain training, especially in PEFT methods, we design out-of-domain analysis to evaluate the ability of methods to generalize on out-of-distribution data.

As shown in Table 3, our approach significantly outperforms other strong baselines while maintaining a lower

Methods	Base Model	Params (%)	Accuracy (\uparrow)				
			Device	Laptop	Restaurant	Service	Avg.
<i>Zero-Shot</i>							
GPT4o-mini (OpenAI 2024)	—	—	87.6	80.0	85.2	86.7	84.9
Llama2-7b (Touvron et al. 2023)	—	—	68.7	53.2	66.5	56.0	61.1
<i>Full-parameter-finetuning</i>							
BERT (Devlin et al. 2019)	BERT-base	1.632%	90.7	71.2	84.8	82.4	82.3
Deberta (He et al. 2021)	DebertaV3-base	2.057%	95.4	74.2	86.3	86.0	85.5
Flan-T5 (Chung et al. 2022)	Flan-T5-base	3.265%	94.5	75.8	86.5	87.4	86.1
<i>Parameter-efficient finetuning</i>							
Adapter (Houlsby et al. 2019)	Llama2-7b	1.953%	73.2	68.6	72.1	75.2	72.3
LoRA (Hu et al. 2022)	Llama2-7b	0.826%	95.2	76.9	85.7	88.3	86.5
Dora (Liu et al. 2024)	Llama2-7b	0.838%	95.1	77.2	85.9	88.1	86.6
PrefixFT (Li and Liang 2021)	Llama2-7b	0.039%	68.9	53.2	66.5	37.6	56.6
LoReft (Wu et al. 2024)	Llama2-7b	0.031%	94.2	73.7	87.3	88.5	85.9
<i>Specific-layers finetuning</i>							
LoRA (Hu et al. 2022)	Llama2-7b	0.129%	93.6	73.2	85.5	86.8	84.8
LoReft (Wu et al. 2024)	Llama2-7b	0.006%	92.9	72.3	86.9	87.6	84.9
Ours	Llama2-7b	0.006%	96.2	76.7	88.1	89.5	87.6

Table 2: Overall accuracy (%) over four in-domain aspect-based sentiment classification datasets. The "Params" column indicates the proportion of trainable parameters relative to the number of parameters in Llama2-7b. Specific-layers finetuning refers to updating only the middle layers, specifically layers 10 to 15.

number of trainable parameters. Although we are concerned about the potential risk of data leakage with GPT4o-mini (Kocooń et al. 2023), our method still surpasses GPT-4o-mini in some domains and in average accuracy. This indicates the effectiveness of our method in handling data variability and ensuring robust generalization across different domains. LoRA and LoReft, as state-of-the-art PEFT methods, show strong performance in specific-layers finetuning, but these methods are still vulnerable in some out-of-distribution samples. Our methods, based on the causal tracing results of sentiment association, not only achieve high accuracy but also maintain computational efficiency. The results clearly indicate that our method offers a balanced combination of accuracy and efficiency.

Impact on Specific Layers

As the causal tracing results mentioned before, the mid-layers show great influence on predicting the sentiment polarity of the given aspect. To further explore this impact, we conduct a quantitative experiment to analyze its impact on the performance of different methods in average accuracy over 16 domain pairs, including in-domain and out-of-domain datasets.

As shown in Figure 4, different methods exhibit consistent results across various layers. This consistency indicates the significant influence of different layers in the LLM on addressing the ABSC task. Specifically, within the 10-15 layer range, both methods achieve results comparable to their best performance when training parameters across all layers (0-31). This finding aligns with our previously mentioned causal tracing sentiment association results. More-

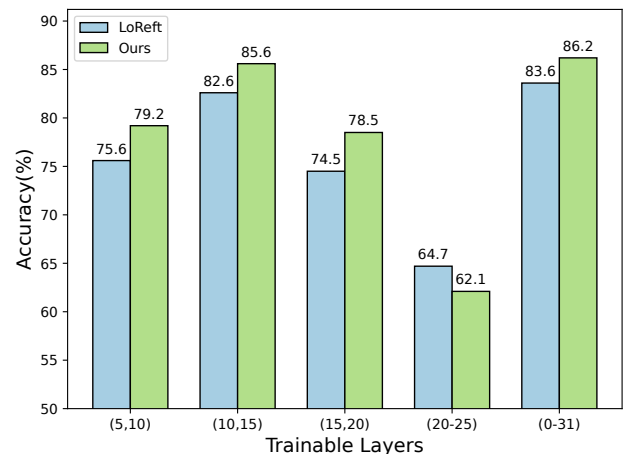


Figure 4: Impact on trainable layers of LoReft and our method.

over, our method demonstrates a more stable and superior performance compared to LoReft. This stability is particularly notable across all evaluated layer ranges, further indicating the robustness and effectiveness of our method.

Influence of Specific Positions

For our method, we must decide which layers and input positions to apply the intervention on. The causal tracing results reveal that aspect terms in mid-layers have a significant influence on the sentiment prediction. In this section, we in-

Methods	Accuracy (\uparrow)										
	(D, R)	(D, S)	(L, R)	(L, S)	(R, D)	(R, L)	(R, S)	(S, D)	(S, L)	(S, R)	Avg.
<i>Zero-Shot</i>											
GPT-4o-mini (OpenAI 2024)	85.2	86.7	85.2	86.7	87.6	80.0	86.7	87.6	80.0	85.2	85.1
Llama2-7b (Touvron et al. 2023)	66.5	56.0	66.5	56.0	68.7	53.2	56.0	68.7	53.2	66.5	61.1
<i>Full-parameter-fintuning</i>											
BERT (Devlin et al. 2019)	80.4	85.2	81.1	81.9	88.3	77.8	84.5	90.7	69.7	79.8	81.9
Deberta (He et al. 2021)	81.1	85.4	80.1	82.7	88.3	78.4	86.2	93.9	68.5	81.0	82.6
Flan-T5 (Chung et al. 2022)	81.8	87.6	79.5	85.3	93.9	78.3	87.0	93.5	70.4	81.2	83.9
<i>Parameter-efficient finetuning</i>											
LoRA (Hu et al. 2022)	82.9	87.2	85.0	81.6	84.9	79.2	75.2	94.4	70.7	83.8	82.5
Dora (Liu et al. 2024)	82.4	87.4	84.7	81.2	84.5	80.2	75.1	94.6	69.9	84.1	82.4
LoReft (Wu et al. 2024)	84.0	87.6	83.2	85.2	94.5	72.9	88.5	94.1	71.3	84.4	84.6
<i>Specific-layers finetuning</i>											
LoRA (Hu et al. 2022)	81.9	86.2	84.2	80.1	83.2	77.5	72.3	92.8	67.2	82.1	80.8
LoReft (Wu et al. 2024)	83.8	83.4	83.3	83.3	95.2	72.4	86.6	92.2	65.9	83.3	82.9
Ours	85.8	88.7	85.0	85.2	92.9	75.6	88.5	95.8	71.6	84.9	85.4

Table 3: Overall accuracy (%) over 12 out-of-domain aspect-based sentiment classification dataset pairs. The column "(D,R)" indicates the model is trained on **Device** domain and tested on **Restaurant** domain, except these zero-shot methods. Specific-layers finetuning refers to updating only the middle layers, specifically layers 10 to 15.

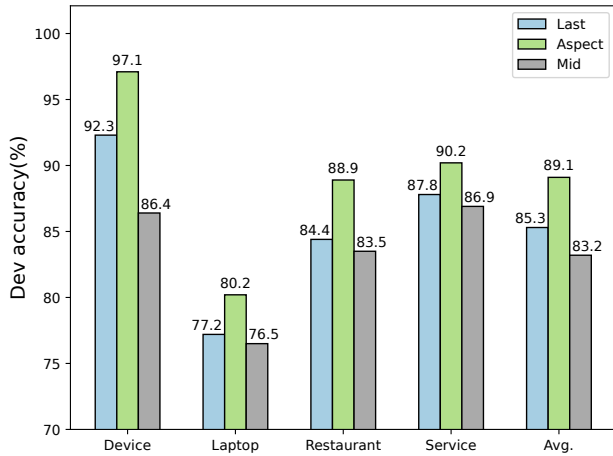


Figure 5: Influence on specific positions: 'aspect' refers to editing on aspect terms, 'last' refers to editing on the last words, and 'mid' refers to editing on a random mid-position word.

investigate three potential word positions within the specific layers (10-15) to evaluate our methods. Intuitively, the last word and aspect word may have a strong impact, but we also take a random mid-word from the input to evaluate the influence of the intervention position. In this experiment, we take 10% of data from the training dataset as the development dataset for in-domain ABSC.

As shown in Figure 5, editing on aspect terms yields the highest performance. It highlights the pivotal role aspect terms play in ABSC. Additionally, the last word shows relatively good performance due to the use of a decoder-only model. However, the performance still falls short compared

to editing on aspect terms. This indicates that for specific tasks, targeting the key terms relevant to the task, such as aspect terms, can lead to better results. Notably, the performance does not degrade significantly even when a random mid-position word is chosen for intervention. It suggests that while targeted editing on aspect terms is optimal, our method, which containing weight, and representation based editing methods, remains robust and does not suffer from significant performance drops due to random selection.

Conclusion

We explore model editing as an efficient and interpretable fine-tuning method for LLM-based aspect-based sentiment classification. By combining causal tracing with targeted model edits, we demonstrate that model editing can serve as an efficient fine-tuning method with minimal parameter updates. In addition, we empirically validate that specific mid-layer representations of aspect words play a crucial role in aspect-based sentiment classification. Practical experiments confirm the effectiveness of our framework, demonstrating that it outperforms state-of-the-art parameter-efficient fine-tuning methods with fewer trainable parameters. These results indicate the potential of model editing as a more interpretable and efficient strategy for adapting LLMs to specific downstream tasks, highlighting its broader applicability.

Acknowledgments

We would like to thank Yanzhi Xu, Tianlai Ma and the anonymous reviewers for their insightful and valuable comments. This work is partially supported by the National Natural Science Foundation of China (No. 62276177) and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

- Avitan, M.; Cotterell, R.; Goldberg, Y.; and Ravfogel, S. 2024. What changed? converting representational interventions to natural language. *arXiv preprint arXiv:2402.11355*.
- Chung, H. W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, E.; Wang, X.; Dehghani, M.; Brahma, S.; Webson, A.; Gu, S. S.; Dai, Z.; Suzgun, M.; Chen, X.; Chowdhery, A.; Narang, S.; Mishra, G.; Yu, A.; Zhao, V.; Huang, Y.; Dai, A.; Yu, H.; Petrov, S.; Chi, E. H.; Dean, J.; Devlin, J.; Roberts, A.; Zhou, D.; Le, Q. V.; and Wei, J. 2022. Scaling Instruction-Finetuned Language Models. *arXiv preprint arXiv:2210.11416*.
- Dai, D.; Dong, L.; Hao, Y.; Sui, Z.; Chang, B.; and Wei, F. 2022. Knowledge Neurons in Pretrained Transformers. In *Proceedings of ACL*.
- Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*.
- Geiger, A.; Lu, H.; Icard, T.; and Potts, C. 2021. Causal abstractions of neural networks. In *Proceedings of NeurIPS*.
- Han, C.; Xu, J.; Li, M.; Fung, Y.; Sun, C.; Jiang, N.; Abdelzaher, T.; and Ji, H. 2024. Word Embeddings Are Steers for Language Models. In *Proceedings of ACL*.
- He, P.; Liu, X.; Gao, J.; and Chen, W. 2021. DEBERTA: DECODING-ENHANCED BERT WITH DISENTANGLED ATTENTION. In *Proceedings of ICLR*.
- Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; de Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-Efficient Transfer Learning for NLP. In *Proceedings of ICML*.
- Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. Lora: Low-rank adaptation of large language models. In *Proceedings of ICLR*.
- Hu, M.; and Liu, B. 2004. Mining and summarizing customer reviews. In *Proceedings of SIGKDD*.
- Huang, B.; and Carley, K. M. 2018. Parameterized Convolutional Neural Networks for Aspect Level Sentiment Classification. In *Proceedings of EMNLP*.
- Jiang, B.; Hou, J.; Zhou, W.; Yang, C.; Wang, S.; and Pang, L. 2020. METNet: A Mutual Enhanced Transformation Network for Aspect-based Sentiment Analysis. In *Proceedings of COLING*.
- Kocoń, J.; Cichecki, I.; Kaszyca, O.; Kochanek, M.; Szydło, D.; Baran, J.; Bielaniec, J.; Gruz, M.; Janz, A.; Kanclerz, K.; Kocoń, A.; Koptyra, B.; Mieszczenko-Kowszewicz, W.; Miłkowski, P.; Oleksy, M.; Piasecki, M.; Radliński, ; Wojtasik, K.; Woźniak, S.; and Kazienko, P. 2023. ChatGPT: Jack of all trades, master of none. *Information Fusion*, 99: 101861.
- Li, S.; Wang, Z.; Jiang, X.; and Zhou, G. 2022. Cross-Domain Sentiment Classification using Semantic Representation. In *Findings of EMNLP*.
- Li, X. L.; and Liang, P. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In *Proceedings of ACL*, 4582–4597.
- Liu, S.; Xing, L.; and Zou, J. 2023. In-context vectors: Making in context learning more effective and controllable through latent space steering. *arXiv preprint arXiv:2311.06668*.
- Liu, S.-y.; Wang, C.-Y.; Yin, H.; Molchanov, P.; Wang, Y.-C. F.; Cheng, K.-T.; and Chen, M.-H. 2024. DoRA: Weight-Decomposed Low-Rank Adaptation. In *Proceedings of ICML*.
- Loshchilov, I.; and Hutter, F. 2018. Decoupled Weight Decay Regularization. *arXiv preprint arXiv:1711.05101*.
- Ma, D.; Li, S.; Zhang, X.; and Wang, H. 2017. Interactive Attention Networks for Aspect-Level Sentiment Classification. In *Proceedings of IJCAI*.
- Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*.
- Meng, K.; Sharma, A. S.; Andonian, A. J.; Belinkov, Y.; and Bau, D. 2023. Mass-Editing Memory in a Transformer. In *Proceedings of ICLR*.
- Mitchell, E.; Lin, C.; Bosselut, A.; Finn, C.; and Manning, C. D. 2022. Fast Model Editing at Scale. In *Proceedings of ICLR*.
- OpenAI. 2024. GPT-4o-mini.
- Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P. F.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arxiv:2203.02155*.
- Pearl, J. 2022. Direct and indirect effects. In *Probabilistic and causal inference: the works of Judea Pearl*.
- Pontiki, M.; Galanis, D.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S.; Al-Smadi, M.; Al-Ayyoub, M.; Zhao, Y.; Qin, B.; Clercq, O. D.; Hoste, V.; Apidianaki, M.; Tannier, X.; Loukachevitch, N. V.; Kotelnikov, E. V.; Bel, N.; Zafra, S. M. J.; and Eryigit, G. 2016. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In *Proceedings of NAACL-HLT*.
- Pontiki, M.; Galanis, D.; Papageorgiou, H.; Manandhar, S.; and Androutsopoulos, I. 2015. SemEval-2015 Task 12: Aspect Based Sentiment Analysis. In *Proceedings of NAACL-HLT*.
- Pontiki, M.; Galanis, D.; Pavlopoulos, J.; Papageorgiou, H.; Androutsopoulos, I.; and Manandhar, S. 2014. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proceedings of COLING*.
- Ravfogel, S.; Twiton, M.; Goldberg, Y.; and Cotterell, R. D. 2022. Linear Adversarial Concept Erasure. In *Proceedings of ICML*.
- Singh, S.; Ravfogel, S.; Herzig, J.; Aharoni, R.; Cotterell, R.; and Kumaraguru, P. 2024. Mimic: Minimally modified counterfactuals in the representation space. *arXiv preprint arXiv:2402.09631*.

- Sun, C.; Huang, L.; and Qiu, X. 2019. Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence. In *Proceedings of NAACL-HLT*.
- Tang, D.; Qin, B.; Feng, X.; and Liu, T. 2016. Effective LSTMs for Target-Dependent Sentiment Classification. In *Proceedings of COLING*.
- Tang, H.; Ji, D.; Li, C.; and Zhou, Q. 2020. Dependency Graph Enhanced Dual-transformer Structure for Aspect-based Sentiment Classification. In *Proceedings of ACL*.
- Tay, Y.; Tuan, L. A.; and Hui, S. C. 2018. Learning to Attend via Word-Aspect Associative Fusion for Aspect-Based Sentiment Analysis. In *Proceedings of AAAI*.
- Toprak, C.; Jakob, N.; and Gurevych, I. 2010. Sentence and Expression Level Annotation of Opinions in User-Generated Discourse. In *Proceedings of ACL*.
- Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2307.09288*.
- Turner, A.; Thiergart, L.; Udell, D.; Leech, G.; Mini, U.; and MacDiarmid, M. 2023. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*.
- Vig, J.; Gehrmann, S.; Belinkov, Y.; Qian, S.; Nevo, D.; Singer, Y.; and Shieber, S. 2020. Investigating gender bias in language models using causal mediation analysis. *Advances in neural information processing systems*.
- Wang, K.; Shen, W.; Yang, Y.; Quan, X.; and Wang, R. 2020. Relational Graph Attention Network for Aspect-based Sentiment Analysis. In *Proceedings of ACL*.
- Wu, Z.; Arora, A.; Wang, Z.; Geiger, A.; Jurafsky, D.; Manning, C. D.; and Potts, C. 2024. ReFT: Representation Finetuning for Language Models. *arXiv preprint arXiv:2404.03592*.
- Xu, H.; Liu, B.; Shu, L.; and Yu, P. S. 2019. BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis. In *Proceedings of NAACL-HLT*.
- Xue, W.; and Li, T. 2018. Aspect Based Sentiment Analysis with Gated Convolutional Networks. In *Proceedings of ACL*.
- Yao, Y.; Wang, P.; Tian, B.; Cheng, S.; Li, Z.; Deng, S.; Chen, H.; and Zhang, N. 2023. Editing Large Language Models: Problems, Methods, and Opportunities. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of EMNLP*.
- Zhang, M.; and Qian, T. 2020. Convolution over Hierarchical Syntactic and Lexical Graphs for Aspect Level Sentiment Analysis. In *Proceedings of EMNLP*.
- Zhang, W.; Li, X.; Deng, Y.; Bing, L.; and Lam, W. 2023. A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges. *IEEE Transactions on Knowledge & Data Engineering*.
- Zhang, Z.; Zhou, Z.; and Wang, Y. 2022. SSEGCN: Syntactic and Semantic Enhanced Graph Convolutional Network for Aspect-based Sentiment Analysis. In *Proceedings of NAACL*.
- Zou, A.; Phan, L.; Chen, S.; Campbell, J.; Guo, P.; Ren, R.; Pan, A.; Yin, X.; Mazeika, M.; Dombrowski, A.-K.; et al. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.