# Mitigating Sentiment Recognition Bias in Aspect-Based Sentiment Analysis via Multifaceted Data Enhancement

**Anonymous ACL submission**

## Abstract

Aspect-Based Sentiment Analysis (ABSA) focuses on analyzing the sentiment of specific aspect terms. Despite substantial progress in this field, most models often exhibit significant biases, particularly in recognizing neutral sentiments, due to the predominance of emotional content in training datasets. To improve the quality of data and enhance model comprehension of aspect term sentiments across diverse context, we propose the **M**ultifaceted **D**ata **E**nhancement (MDE) framework, which enhances both the breadth and depth of ABSA datasets. MDE leverages large language models (LLMs) for data paraphrasing and implements a Dual Confidence Filtering algorithm to select high-quality samples, thereby enhancing data diversity. Furthermore, MDE incorporates data enhancement strategies for aspect term clarification and sentiment reasoning. Through multiple rounds of inquiry with LLMs, MDE refines the understanding of aspect terms and strengthens the logical consistency between data and sentiment labels. We apply MDE to several ABSA benchmark datasets and fine-tune various models. Experimental results demonstrate that MDE effectively mitigates sentiment recognition bias and outperforms state-of-the-art baselines [1].

## 1 Introduction

Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment analysis task that aims to identify the sentiment polarity towards specific aspect terms within a given review (Pontiki et al., 2014). Due to its broad applicability across diverse real-world contexts, ABSA is considered a pivotal task within the field of sentiment analysis.

In recent years, neural network-based solutions for ABSA have achieved notable success. Recurrent neural networks and attention mechanisms have been employed to capture term-context relationships (Tang et al., 2016; Wang et al., 2016; Cheng et al., 2017; Li et al., 2018), while graph neural networks (GNNs) have been utilized to exploit syntactic structures (Huang and Carley, 2019; Sun et al., 2019; Wang et al., 2020; Zhang et al., 2022). The advent of pretrained models has further elevated ABSA performance (Song et al., 2019; Xu et al., 2019; Li et al., 2021a; Yang and Li, 2024). More recently, large language models (LLMs), such as ChatGPT, have demonstrated impressive zero-shot capabilities in sentiment classification tasks (Wang et al., 2024b).

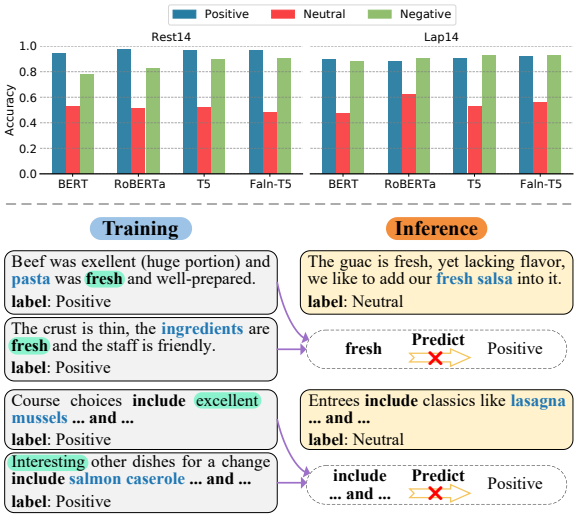


Figure 1: The phenomenon of low accuracy in recognizing neutral sentiments.

Despite these advancements, fine-tuned pretrained models exhibit obvious sentiment bias with a low recognition accuracy for neutral sentiment. As shown in the upper part of Fig. 1, fine-tuning models such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), T5 (Raffel et al., 2020) and Flan-T5 (Chung et al., 2024) on ABSA datasets (Pontiki et al., 2014) reveals a significant performance drop in neutral sentiment detection. These

[1] Code and data will be available after anonymous review.

models fine-tuned on emotionally rich user reviews tend to overfit on explicit emotional expressions. For example, as shown in the lower part of Fig. 1, the term "*fresh*" frequently implies a positive sentiment in food-related reviews, but "*fresh salsa*" refers to a type of salsa without inherent emotional connotation. Similarly, the phrase structure "*include... and...*" is often associated with optional dishes. These dishes are often described with subjective adjectives in training set, which leads to incorrect positive classification when this structure is used to convey factual statements in the test data.

Due to model bias, neutral sentiment aspect terms in emotionally charged data can lead to inaccurate predictions when subtle variations occur. The underlying causes of this issue are two fold: 1) the limited and low-diverse nature of the training data, and 2) the model's inability to accurately capture the relationship between context and aspect terms. To address these, we propose a **M**ultifaceted **D**ata **E**nhancement (MDE) framework designed to expand both the **breadth** and **depth** of the dataset, which increases diversity and uncovers nuanced relationships between aspect terms and sentiment.

To expand the data breadth, LLMs can effortlessly generate large amounts of synthetic data. However, LLMs struggle with aspect term extraction and neutral sentiment identification (Wang et al., 2024b; Xu et al., 2024), leading to potential annotation errors. Therefore, MDE leverages LLMs to produce paraphrased candidate data from existing datasets, avoiding direct labeling by LLMs. It then implements a Dual Confidence Filtering (DCF) algorithm to select high-quality samples based on confidence scores from both correctly classified and misclassified data, ensuring enhanced data diversity.

For deepening data exploration, MDE incorporates enhancement for aspect term clarification and sentiment logic reasoning. By leveraging the knowledge stored in LLMs, MDE elucidates the meanings of aspect terms, preventing misinterpretations of specialized aspects. Additionally, the reasoning process strengthens the logical consistency between data and sentiment labels, thus mitigating the model's tendency to learn erroneous shortcuts.

We apply MDE to enhance several ABSA benchmark datasets and fine-tune various pretrained models, achieving significant performance improvements. In particular, for encoder-decoder models such as T5 and Flan-T5, transforming sentiment classification into sentiment prediction generation with reasoning notably enhances the model's comprehension of the relationship between intrinsic semantics and sentiment. Experimental results demonstrate that MDE surpasses state-of-the-art (SOTA) baselines, significantly improves the accuracy of neutral sentiment recognition, and exhibits superior performance in robustness tests.

Our contributions are summarized as follows:

- We propose MDE framework, enhancing both the breadth and depth of ABSA datasets through LLM-driven paraphrase generation and task-specific sentiment reasoning, improving data diversity and quality.
- We transform the training objective by shifting from sentiment classification to sentiment prediction generation with reasoning, which enhances models' understanding of the relationship between semantics and sentiment.
- Experiments show that MDE achieves significant performance improvements on ABSA, surpassing SOTA baselines and notably boosting neutral sentiment recognition accuracy.

## 2 Related Work

### 2.1 Aspect-based Sentiment Analysis

ABSA aims to analyze the sentiment towards specific aspects within a sentence. Initially, attention mechanisms are employed to capture relationships between context and target words (Tang et al., 2016; Wang et al., 2016; Cheng et al., 2017; Li et al., 2018; Gu et al., 2018; Fan et al., 2018). Subsequently, researchers incorporate syntactic information and use graph neural networks to model syntactic connections between words and target aspects (Huang and Carley, 2019; Sun et al., 2019; Wang et al., 2020; Zhang et al., 2022; Liang et al., 2022). More recently, pretrained language models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) are applied to ABSA with notable success (Song et al., 2019; Jiang et al., 2019; Wang et al., 2020, 2022b). Additionally, LLMs show significant advancements in sentiment analysis, particularly excelling in zero-shot scenarios (Fei et al., 2023; Wang et al., 2024b). These models leverage extensive linguistic and world knowledge but commonly used base encoders like BERT and RoBERTa exhibit significant sentiment recognition bias. We propose ABSA-specific data enhancements to mitigate the bias.

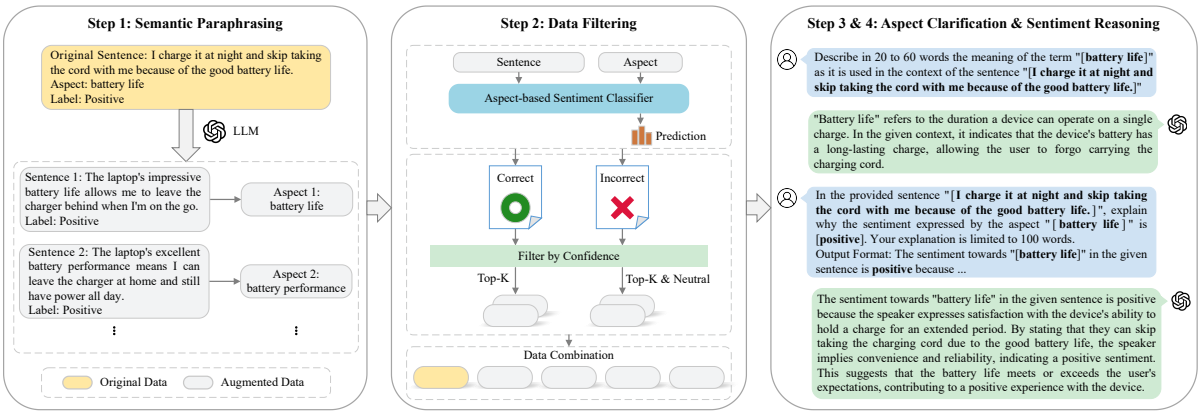


Figure 2: Overview of MDE framework. **Step 1** expands the data breadth, **Step 2** ensures data diversity and quality, and **Step 3** involves mining semantic depth.

### 2.2 Data Enhancemant for ABSA

Enhancing training datasets is an effective way to improve model performance. In ABSA, it has gained traction (Chen et al., 2022; Wang et al., 2022a; Hsu et al., 2021). However, traditional methods, such as token replacement, masked aspect prediction, and polarity reversal, often lack semantic diversity. Recent approaches leverage language models to generate more varied expressions. Ouyang et al. (2024) propose generating sentences with more explicit opinion words to enhance the understanding of implicit sentiment for specific aspects. Chang et al. (2024) use LLMs to create counterfactual data, which strengthens model robustness. Wang et al. (2023) harness the reasoning capabilities of LLMs to produce explanatory sentiment information as training data, aiming to reduce spurious correlations in ABSA. Additionally, Deng et al. (2023) and Wang et al. (2024a). generate new sentences using aspect-opinion-sentiment tuples to address cross-domain data scarcity. In contrast, our MDE approach takes a multifaceted approach, considering data diversity, quality, and logical consistency to mitigate model bias.

## 3 Methodology

### 3.1 Task Definition

Given a dataset $D_o = \{(x_i, y_i) \mid i \in [1, |D_o|]\}$ containing $|D_o|$ instances, each instances $x_i$ consists of a sentence $s_i$ and an aspect term $a_i$ that is a subsequence of $s_i$. Each $x_i$ has a sentiment label $y_i \in \{Positive, Negative, Neutral\}$. The goal of ABSA is to predict a sentiment polarity $\hat{y}_i$ towards the aspect $a_i$ given the input $x_i$.

### 3.2 Method Overview

The method consists of two phases: MDE and model training. Firstly, MDE construct enhanced dataset through four key steps: semantic paraphrasing, data filtering, aspect clarification, and sentiment reasoning. Then, a sentiment reasoning model is trained on the MDE dataset. This training incorporates the broader data coverage and deeper semantic insights provided by MDE, enabling the model to develop intrinsic logical reasoning capabilities. Following sections will elaborate on these components.

### 3.3 Multifaceted Data Enhancement

We first outline MDE and define some symbols. Let $D_o$ represent the original dataset. The first step is to generates $N$ new samples for each instance, resulting in the paraphrased dataset $D_p$. After filtering $D_p$, we obtain the filtered dataset $D_f$. The final step involves merging $D_o$ and $D_f$ to incorporate aspect clarification and sentiment reasoning. The final enhanced dataset $D_e = \{(x_i, c_i, r_i, y_i) \mid i \in [1, |D_e|]\}$, where $c_i$ is the aspect clarification and $r_i$ is the sentiment reasoning.

**Step 1: Semantic Paraphrasing.** Leveraging current LLM technologies enables substantial data generation. However, when using LLMs to construct ABSA data for specific domains, several issues arise: 1) Domain Shift: LLM-generated data may not consistently align with the original domain due to content uncontrollability; 2) Aspect Annotation Deviations: LLMs often diverge from ground-truth aspect labels (Wang et al., 2024b); 3) Sentiment Annotation Bias: LLMs frequently misclassify neutral data as positive or negative sentiment (Wang et al., 2024b).

According to these issues, directly using LLMs to generate sentences and annotate aspects and sentiments may lead to misalignment with the original data. Therefore, we use sentence-to-sentence semantic paraphrasing method to generate $N$ new sentences for each original sentence. This ensures that these sentences possess the same semantics but exhibit varied expressions, thus mitigating domain shift. By adding the original sentence $s_i$ and the original aspect term $a_i$ as reference examples in the prompt, utilizing LLMs to annotate aspects for sentences with similar semantics becomes straightforward and effective. Additionally, there is no need to re-label sentiment for the generated sentences, as their preserved semantics ensure consistent sentiment labels with the original samples. Overall, by employing LLMs for semantic paraphrasing and aspect annotation, we generate a substantial volume of paraphrased data that retains the same sentiment labels as the original samples, denoted as $D_p$.

**Step 2: Data Filtering.** Models trained on original dataset are good at recognizing positive and negative sentiments, but struggle with identifying neutral sentiments. We propose a Dual Confidence Filtering (DCF) algorithm to leverage the strengths and mitigate the weaknesses of data distribution.

As described in Algorithm 1, an ABSA classifier $f$ trained on the original dataset $D_o$ is used to classify the generated paraphrased samples, categorizing them into correctly classified set $P^{cc}$ and misclassified set $P^{mc}$. For $P^{mc}$, we only select those samples with neutral sentiment labels. $P^{cc}$ and $P^{mc}$ represent the strengths and weaknesses of the data, respectively. To further refine the data quality, we rank the samples in $P^{cc}$ by confidence score, retaining the top $K^{cc}$ samples. The confidence score is the probability value to the predicted sentiment. Similarly, for $P^{mc}$, we apply the same ranking method to preserve the top $K^{mc}$ samples. The combined dataset from these two subsets forms the filtered dataset $D_f$.

**Step 3: Aspect Clarification.** Aspect terms may be specialized domain-specific terms whose meanings are difficult to grasp based on limited context alone. We leverage LLMs to provide supplementary clarification on these terms, helping models better understand the underlying meanings and thus improving sentiment analysis accuracy.

**Step 4: Sentiment Reasoning.** Models are prone to learning erroneous shortcuts during training. We utilize LLMs to generate detailed explanations for sentiment polarity judgments. These explanations establish logical connections between data instances and their sentiment labels, thereby enhancing the model's sentiment reasoning capabilities. To ensure consistency between the reasoning and sentiment labels, the labels are incorporated into the prompt as input to the LLMs.

**Algorithm 1** Dual Confidence Filtering

**Input**: Original dataset $D_o$, ABSA model $f$
**Parameter**: $N$, $K^{cc}$, $K^{mc}$
**Output**: Filtered dataset $D_f$

Initialize empty set $D_f$
**for** $x \in D_o$ **do**
  // Obtain $N$ paraphrased samples
  $P = \text{Paraphrase}(x, N)$
  Initialize empty set $P^{cc}$, $P^{mc}$
  **for** $p \in P$ **do**
    // Obtain the prediction and the confidence
    $(\hat{y}, confidence) = f(p)$
    $P^{cc} \leftarrow (p)$, **if** $\hat{y} = y$
    $P^{mc} \leftarrow (p)$, **if** $\hat{y} \neq y$ and $y$ is neutral
  **end for**
  // Sort $P^{cc}$ and $P^{mc}$ based on $confidence$
  $P^{cc} = \text{SortbyConfidence}(P^{cc})$
  $P^{cc} = \text{SortbyConfidence}(P^{mc})$
  // Select top $K$ samples
  $D_f \leftarrow \text{SelectTop}(P^{cc}, K^{cc})$
  $D_f \leftarrow \text{SelectTop}(P^{mc}, K^{mc})$
**end for**
**return** Filtered dataset $D_f$

The aspect clarification and sentiment reasoning are achieved through multi-turn dialogues with LLMs. The dialogue process is depicted in **Step 3 & 4** of Fig. 2. For a given sample $x_i$, the LLM first clarifies the meaning of its aspect terms within the context. Next, the LLM analyze the reasons for the sentiment polarity of the aspect terms being $y_i$, with the resulting explanation denoted as $r_i$. Merging the original dataset $D_o$ and the filtered dataset $D_f$, and performing aspect clarification and sentiment reasoning, we obtain the enhanced dataset $D_e = \{(x_i, c_i, r_i, y_i)\}$.

### 3.4 Model Training

We fine-tune generative T5 and Flan-T5. The training objective is redefined from traditional sentiment classification to sentiment prediction generation with reasoning. The encoder's input consists of the sentence $s_i$, the aspect term $a_i$, and the aspect clarification $c_i$. The decoder's output includes both the sentiment polarity prediction for the aspect and the corresponding explanation for this prediction. The model is trained by minimizing the following loss function:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\log P(g_{i,t}|\hat{g}_{i,<t}, s_i, a_i, c_i) \quad (1)$$

where $g_i = [y_i; r_i]$ represents the target sequence of the output, $g_{i,t}$ denotes the true token at position $t$, $\hat{g}_{i,<t}$ represents the generated sequence at positions less than $t$, and $P(g_{i,t}|\hat{g}_{i,<t}, s_i, a_i, c_i)$ is the probability of generating token $g_{i,t}$ given $\hat{g}_{i,<t}$, $s_i$, $a_i$, and $c_i$.

## 4 Experimental Setup

### 4.1 Datasets.

We evaluate MDE on four ABSA datasets: Rest14 and Lap14 from (Pontiki et al., 2014), Rest15 from (Pontiki et al., 2015), and Rest16 (Pontiki et al., 2016) from (Pontiki et al., 2016). For instances with multiple aspects, each aspect is treated as a separate single-aspect data instance. Detailed statistics of the datasets are provided in Table 1. See the Appendix D for details on MDE data.

### 4.2 Implement Details.

We use LLM GPT-3.5 [2] in MDE. All prompts used are provided in Appendix G. The models T5-base and Flan-T5-base are fine-tuned using the transformers library [3]. The fine-tuning process involve training for 10 epochs using the AdamW optimizer with a learning rate of 1e-4. We set $N = 10$, and select parameters $K^{cc}$ and $K^{mc}$ from the range [0, 3]. We choose the values for $K^{cc}$ and $K^{mc}$ based on the highest F1 score and then run the experiments three times, reporting the average results as the main results. All experiments are implemented in PyTorch and conducted on an A5000 GPU with 24GB of memory. The Accuracy (Acc) and Macro-F1 score (F1) are used as the evaluation metrics.

### 4.3 Compared Baselines

#### 4.3.1 BERT-Based Baselines:

BERT (Devlin et al., 2019) processes sentence-aspect pairs to learn aspect-aware representations. BERT-PT (Xu et al., 2019) further trains BERT on domain-specific data. BERT-RSC (Wang et al., 2023) induces LLMs to generate explanations for aspect sentiment. BERT-CEIB (Chang et al., 2024) uses counterfactual data to reduce spurious correlations. BERT-RGAT (Wang et al., 2020) employs relational graph attention for syntactic dependencies. BERT-DualGCN (Li et al., 2021b) integrates syntactic and semantic knowledge. BERT-SenticGCN (Liang et al., 2022) adds affective knowledge into the dependency graph.

[2]https://openai.com (version: gpt-3.5-turbo-1106)
[3]https://github.com/huggingface/transformers

| Dataset | Split | Pos. | Neu. | Neg. | Total |
|---|---|---|---|---|---|
| Rest14 | Train | 2164 | 633 | 805 | 3602 |
| | Test | 728 | 196 | 196 | 1120 |
| Lap14 | Train | 987 | 460 | 866 | 2313 |
| | Test | 341 | 169 | 128 | 638 |
| Rest15 | Train | 912 | 36 | 256 | 1204 |
| | Test | 326 | 34 | 182 | 542 |
| Rest16 | Train | 1240 | 69 | 439 | 1748 |
| | Test | 469 | 30 | 117 | 616 |

Table 1: Statistics of the ABSA datasets.

#### 4.3.2 T5-Based Baselines:

T5 and Flan-T5 predicts sentiment labels from sentences and aspect terms. T5-C$^3$DA (Wang et al., 2022a) uses contrastive data augmentation by altering aspect terms and sentiment polarity. T5-ESA (Ouyang et al., 2024) generates augmentations with distinct opinion words for aspect terms.

## 5 Experimental Results and Analysis

### 5.1 Main Results

Table 2 is the main results of different methods on four datasets. MDE achieves the best performance across all datasets. Overall, T5-based methods outperform those based on BERT. MDE enhances sentiment prediction by providing logically reasoned data, which requires the model to generate both sentiment predictions and the reasoning behind them. Therefore, MDE data is trained on generative models to realize its full potential. Both T5 and Flan-T5 trained on MDE data exceed previous methods. Compared to data enhanced methods such as RSC, CEIB, C$^3$DA, and ESA, MDE still achieves superior performance due to its comprehensive enhancement of both the breadth and depth of the data.

### 5.2 Performance Breakdown by Sentiment

Table 3 shows the F1 scores for T5 across different sentiment classes on the original and MDE-enhanced datasets[4]. Training with MDE data significantly improves neutral sentiment recognition while maintaining high performance for positive and negative sentiments. Neutral sentiment samples are sparse in Rest15 and Rest16 datasets, with 9.77% and 13.16% F1 improvements on them.

Fig. 3 compares the confusion matrices of vanilla T5 and T5-MDE on the Rest14 test set. MDE no-

[4]Subsequent experments are based on T5, with Flan-T5 results in the Appendix H.

| Method | Rest14 | | Lap14 | | Rest15 | | Rest16 | |
|---|---|---|---|---|---|---|---|---|
| | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 |
| BERT | 84.46 | 76.81 | 78.37 | 73.21 | 82.66 | 65.91 | 90.58 | 75.41 |
| BERT-PT$^\flat$ | 84.95 | 76.96 | 78.07 | 75.08 | - | - | - | - |
| BERT-RSC$^\flat$ | 84.66 | 76.18 | 78.68 | 75.19 | 82.63 | 65.97 | 90.12 | 73.69 |
| BERT-CEIB$^\flat$ | 87.77 | 82.08 | 82.92 | 79.50 | 86.16 | 72.97 | 92.86 | 81.08 |
| BERT-RGAT | 85.18 | 78.38 | 78.21 | 73.27 | 82.84 | 69.33 | 90.91 | 75.76 |
| BERT-DualGCN$^\flat$ | 87.13 | 81.16 | 81.80 | 78.10 | - | - | - | - |
| BERT-SenticGCN$^\flat$ | 86.92 | 81.03 | 82.12 | 79.05 | 85.32 | 71.28 | 91.07 | 79.56 |
| T5 | 87.59 | 80.60 | 81.03 | 76.48 | 87.45 | 74.42 | 93.83 | 78.41 |
| T5-C$^3$DA$^\natural$ | 86.93 | 81.23 | 80.61 | 77.11 | - | - | - | - |
| T5-ESA$^\natural$ | 88.29 | 81.74 | 82.44 | 79.34 | - | - | - | - |
| Flan-T5 | 87.41 | 79.91 | 82.76 | 78.96 | 86.90 | 74.18 | 94.15 | 77.00 |
| T5-MDE | 90.18 | 85.78 | 83.54 | 81.37 | **89.85** | **78.90** | 93.34 | 82.24 |
| Flan-T5-MDE | **90.27** | **85.96** | **84.80** | **82.01** | 88.38 | 76.04 | **94.48** | **84.34** |

Table 2: Experimental results of MDE and baseline models. Results marked with $^\natural$ are from (Ouyang et al., 2024), and those marked with $^\flat$ are from the original papers. All other results are from our own implementations. The highest scores are highlighted in bold, and the previous highest scores are underlined.

| Dataset | Pos. | Neu. | Neg. | Overall |
|---|---|---|---|---|
| Rest14 | 93.55 | 62.39 | 85.85 | 80.60 |
| +MDE | **94.06** | **75.83** | **86.54** | **85.78** |
| Lap14 | **91.42** | 62.46 | 75.56 | 76.48 |
| +MDE | 90.54 | **72.30** | **81.52** | **81.37** |
| Rest15 | 92.61 | 44.78 | 85.88 | 74.42 |
| +MDE | **93.37** | **54.55** | **88.77** | **78.90** |
| Rest16 | **97.37** | 50.00 | **87.87** | 78.41 |
| +MDE | 96.73 | **63.16** | 86.84 | **82.24** |

Table 3: Breakdown of F1 Performance for T5 on original and MDE-enhanced datasets.

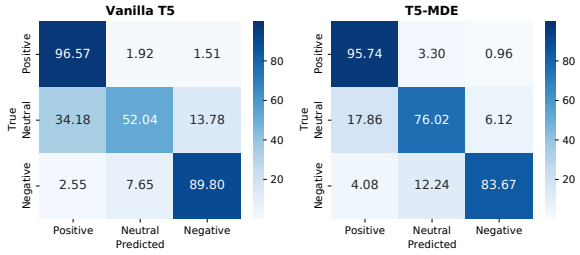


Figure 3: Confusion matrices of Rest14 test set.

tably boosts neutral sentiment accuracy but introduces a trade-off, with a slight increase in misclassifying positive and negative sentiments as neutral. This shift is likely attributed to the increased susceptibility of neutral sentiment labeling to subjectivity compared to positive or negative sentiments. Mild positive or negative instances may be categorized as neutral, and defining what constitutes "mild" is challenging and subjective.

### 5.3 Generalization of MDE.

MDE expands data in three key areas: paraphrase-filtered (PF) data, aspect clarification (AC) data, and sentiment reasoning (SR) data. To validate the generalization of MDE, we applied partially or fully MDE data across different models, as shown in Table 4. BERT-RGAT constructs a syntactic dependency tree based on the input, making it unable to incorporate additional data as input, so it only utilizes PF data. BERT, which are not suited for generative tasks, cannot use SR data. MDE data improves performance across models.

Notably, T5's performance slightly declines with AC and SR data, but improves significantly when the full MDE dataset is used. This suggests that greater data diversity boosts model performance, and integrating AC and SR data within a more varied dataset enhances the model's understanding and reasoning. These findings highlight MDE's effectiveness in balancing data breadth and depth.

### 5.4 MDE Effectiveness Analysis

**DCF vs Random.** In the DCF algorithm, top $K^{cc}$ correctly classified and $K^{mc}$ misclassified samples are selected based on confidence for augmentation. Comparing with random (RD) selection, where $K$ augmented samples are randomly selected for each original sample, Table 5 shows the result. When $K = K^{cc}$ and $K^{mc} \in [1, 3]$, the scale of data expansion is similar, but DCF proves more effective for model training.

**MDE Data Efficiency.** To evaluate MDE data efficiency, we randomly select 10%, 30%, and 50%

| Method | Rest14 | | Lap14 | |
|---|---|---|---|---|
| | Acc | F1 | Acc | F1 |
| BERT-RGAT | 85.18 | 78.38 | 78.21 | 73.27 |
| + PF | **85.71** | **78.94** | **79.15** | **75.31** |
| BERT | 84.46 | 76.81 | 78.37 | 73.21 |
| + PF | 85.27 | 77.18 | 79.62 | 75.74 |
| + AC | 85.09 | 76.75 | 79.62 | 74.58 |
| + PF & AC | **86.43** | **78.89** | **81.35** | **77.42** |
| T5 | 87.59 | 80.60 | 81.03 | 76.48 |
| + PF | 88.04 | 83.45 | 83.07 | 79.87 |
| + AC | 86.96 | 80.14 | 82.76 | 79.16 |
| + SR | 87.14 | 80.18 | 82.45 | 78.91 |
| + AC & SR | 88.66 | 82.58 | 81.35 | 78.22 |
| + MDE | **90.18** | **85.78** | **83.54** | **81.37** |

Table 4: Experimental results of different models using partial or full MDE data.

| Method | Rest14 | | Lap14 | |
|---|---|---|---|---|
| | Scale | F1 | Scale | F1 |
| T5 | 1.00 | 82.58 | 1.00 | 78.22 |
| RD (1) | 2.00 | 84.16 | 2.00 | 80.13 |
| DCF (1) | 2.08 | **85.76** | 2.09 | **81.30** |
| RD (2) | 3.00 | 84.29 | 3.00 | 81.12 |
| DCF (2) | 3.01 | **85.36** | 3.14 | **82.01** |
| RD (3) | 4.00 | 84.57 | 4.00 | 80.08 |
| DCF (3) | 4.15 | **85.78** | 3.97 | **81.85** |

Table 5: Comparison of DCF and random selection. The numbers in parentheses is the values of $K$ and $K^{cc}$.

| Method | Rest14 | | Lap14 | |
|---|---|---|---|---|
| | Scale | F1 | Scale | F1 |
| 10% $D_o$ | 0.10 | 73.38 | 0.10 | 71.62 |
| MDE (1) | 0.11 | 73.67 | 0.11 | 77.13 |
| MDE (2) | 0.22 | 76.90 | 0.22 | **77.18** |
| MDE (3) | 0.30 | **78.73** | 0.30 | **77.18** |
| 30% $D_o$ | 0.30 | 75.77 | 0.30 | 76.06 |
| MDE (1) | 0.38 | 79.50 | 0.33 | 79.49 |
| MDE (2) | 0.66 | 82.39 | 0.61 | 79.83 |
| MDE (3) | 0.94 | **83.15** | 0.95 | **81.17** |
| 50% $D_o$ | 0.50 | 79.30 | 0.50 | 79.62 |
| MDE (1) | 0.60 | 79.29 | 0.54 | 81.10 |
| MDE (2) | 1.12 | 83.64 | 1.11 | **81.43** |
| MDE (3) | 1.47 | **84.02** | 1.49 | 81.02 |
| full $D_o$ | 1.00 | 82.58 | 1.00 | 78.22 |
| MDE (1) | 1.20 | 84.07 | 1.09 | **81.35** |
| MDE (2) | 2.13 | 84.59 | 2.03 | 80.90 |
| MDE (3) | 3.15 | **85.65** | 3.08 | 81.17 |

Table 6: Comparison of T5 performance using original data ($D_o$) versus MDE data (from $D_p$). The numbers in parentheses is the values of $K^{cc}$, and the underlined values indicate data sizes close to the original data.

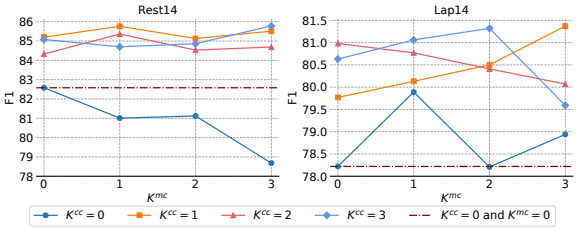


Figure 4: Results of T5 for different $K^{cc}$ and $K^{mc}$.

of the original data to construct the corresponding MDE data. In the main experiment, MDE data includes two sources, $D_o$ and $D_p$. Here, we use only $D_p$ to compare the training performance of the new data against the original data. As shown in Table 6, MDE data (only from $D_p$) consistently outperforms the original data. Notably, with 30% and 50% original data, when the MDE data scale is approximately 1x, the model performance surpasses that of using the full original data, demonstrating the efficiency of MDE data.

**Effect of $K^{cc}$ and $K^{mc}$.** We conduct experiments with $K^{cc}$ and $K^{mc}$ values ranging from 0 to 3. The results, shown in Fig. 4, indicate significant performance differences across various combinations of $K^{cc}$ and $K^{mc}$. Configurations with $K^{cc} \neq 0$ consistently outperform those with those with $K^{cc} = 0$. On the Rest14 dataset, setting $K^{cc} = 0$ results in consistently poor performance, regardless of $K^{mc}$. Solely increasing misclassified data may cause the model to forget previously learned information, so including correctly classified data is essential to preserve its strengths. Overall, the model performs optimally when $K^{cc}$ is set to 2 or 3 and $K^{mc}$ to 2.

Furthermore, we investigate the impact of correctly classified data (CC Data), misclassified data (MC Data), and both types (PF Data) on the recognition of different sentiments. As shown in Table 7, performance for positive sentiment remains relatively high, while neutral sentiment exhibits noticeable fluctuations. Adding only CC or MC data does not always result in consistent improvement. The use of PF data, combining both types, better balances strengths and weaknesses, effectively mitigating the model's sentiment recognition bias.

### 5.5 MDE Data Quality Analysis

We leverage LLMs to paraphrase the original sentences and use them as references to annotate aspect terms in the new sentences. Since the new sentences may use different expressions, the aspect terms in these sentences may not exactly match those in the original data (e.g., "*cord*," "*power cord*," "*charger*"). This introduces greater data

| Method | Rest14 | | | | Lap14 | | | |
|---|---|---|---|---|---|---|---|---|
| | Pos. | Neu. | Neg. | Overall | Pos. | Neu. | Neg. | Overall |
| T5 ($K^{cc}=0, K^{mc}=0$) | 94.16 | 68.22 | 85.35 | 82.58 | 89.49 | 66.24 | 78.93 | 78.22 |
| w/ MC Data ($K^{cc}=0$) | 92.06 | 62.80 | 81.17 | 78.68 | 88.96 | 68.42 | 79.43 | 78.94 |
| w/ CC Data ($K^{mc}=0$) | 94.55 | 74.02 | **86.68** | 85.08 | **90.55** | 68.75 | 80.00 | 79.77 |
| w/ PF Data ($K^{cc}\neq0, K^{mc}\neq0$) | **94.96** | **75.83** | 86.54 | **85.78** | 90.54 | **72.30** | **81.25** | **81.37** |

Table 7: Experimental results under different filtering settings, reporting the F1 for each category and the overall macro-F1.

| Metric | Rest14 | Lap14 | Rest15 | Rest16 |
|---|---|---|---|---|
| $P_{sub}$ (%) | 90.46 | 90.44 | 91.64 | 91.95 |
| $P_{co}$ (%) | 71.27 | 70.33 | 70.27 | 68.54 |

Table 8: Results of $P_{sub}$ and $P_{co}$ in different MDE-enhanced dataset.

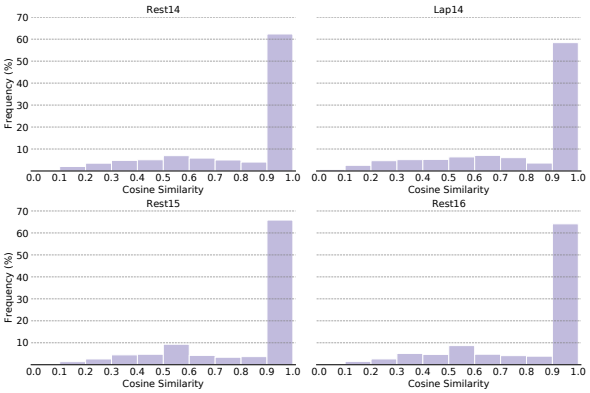


Figure 5: Distribution of similarity scores between new and original aspect terms in different datasets

| Method | Rest14-ARTS | | Lap14-ARTS | |
|---|---|---|---|---|
| | F1 | Drop | F1 | Drop |
| BERT-RGAT | 60.10 | -21.25 | 55.68 | -18.38 |
| BERT-CEIB | 73.97 | -8.06 | 65.51 | -12.02 |
| T5-MDE | **79.65** | **-6.13** | 75.49 | -5.88 |
| Flan-T5-MDE | 79.46 | -6.50 | **76.82** | **-5.19** |

Table 9: Robustness results on ARTS test set.

diversity. Additionally, a few aspect terms may not strictly be substrings of the sentence, such as "*widely used hardware*" in the sentence "*It now utilizes hardware that is widely used in the industry.*" While this may not meet the substring requirement, it is semantically reasonable and helps improve model robustness.

Given these considerations, we evaluate the validity of aspect terms using three metrics: 1) **aspect term substring ratio** ($P_{sub}$): the correctness of aspect term formatting; 2) **co-occurrence ratio** of new and original aspect terms ($P_{co}$): the lexical similarity between the two; 3) **semantic similarity** of new and original aspect terms: the cosine similarity based on their embeddings[5]. Table 8 shows results for $P_{sub}$ and $P_{co}$, demonstrating high accuracy and lexical similarity. Fig. 5 displays the distribution of similarity scores between new and original aspect terms, with most values concentrated between 0.9 and 1. These results indicate that the new aspect terms are highly accurate and retain strong similarity to the original terms.

[5]https://huggingface.co/sentence-transformers/

### 5.6 Robustness Analysis.

Rest14-ARTS and Lap14-ARTS (Xing et al., 2020) are adversarial datasets designed to test ABSA models' robustness by manipulating sentiment for target and non-target aspects. As shown in Table 9, the experimental results indicate that T5-MDE and Flan-T5-MDE significantly outperform other methods. Compared to the main results in Table 2, our models exhibit smaller performance drops, highlighting their superior robustness.

## 6 Conclusion and Future Work

In this paper, we propose the MDE framework to address sentiment recognition bias in ABSA models. MDE enhances data diversity and semantic depth through four key steps: semantic paraphrasing, data filtering, aspect clarification, and sentiment reasoning. By training generative models with MDE data, we improve the logical coherence of both sentiment predictions and explanations.

In the future, extending the MDE framework to other sentiment analysis tasks is a promising direction. Ensuring high-quality and diverse training data is crucial for developing more reliable and robust sentiment analysis models. Additionally, further refining the MDE framework to build high-quality demonstrations for in-context learning offers an efficient approach to address various tasks without extensive training.

## Limitations

### Dependence on LLMs

MDE relies on LLMs for semantic paraphrasing, knowledge enhancement, and explainable sentiment analysis. The quality of the enhanced data is directly tied to the performance and capabilities of the underlying LLMs. Since the performance of different LLMs varies, investigating whether using multiple LLMs to generate more diverse data can further enhance the robustness of ABSA models is a potential area for future research. Additionally, the performance of LLMs is often closely related to the design of prompts. Exploring various prompt designs to push the boundaries of LLMs is another worthy focus.

### Sentiment Recognition Bias of LLMs

LLMs exhibit significant sentiment recognition biases in ABSA tasks compared to human-annotated data. Given the enormous computational resources required to train large models, retraining them is often impractical. Addressing sentiment recognition bias through prompt optimization, example-based prompting, or parameter-efficient fine-tuning methods is a promising area of investigation. Moreover, exploring whether similar bias issues exist in other tasks can serve as a direction for future research.

## References


Mingshan Chang, Min Yang, Qingshan Jiang, and Ruifeng Xu. 2024. Counterfactual-enhanced information bottleneck for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17736–17744.

David Z. Chen, Adam Faulkner, and Sahil Badyal. 2022. Unsupervised data augmentation for aspect based sentiment analysis. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6746–6751, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Jiajun Cheng, Shenglin Zhao, Jiani Zhang, Irwin King, Xin Zhang, and Hui Wang. 2017. Aspect-level sentiment classification with heat (hierarchical attention) network. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, page 97–106, New York, USA. Association for Computing Machinery.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Yue Deng, Wenxuan Zhang, Sinno Jialin Pan, and Lidong Bing. 2023. Bidirectional generative framework for cross-domain aspect-based sentiment analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12272–12285, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3433–3442, Brussels, Belgium. Association for Computational Linguistics.

Hao Fei, Bobo Li, Qian Liu, Lidong Bing, Fei Li, and Tat-Seng Chua. 2023. Reasoning implicit sentiment with chain-of-thought prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1171–1182, Toronto, Canada. Association for Computational Linguistics.

Shuqin Gu, Lipeng Zhang, Yuexian Hou, and Yin Song. 2018. A position-aware bidirectional attention network for aspect-level sentiment analysis. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 774–784, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ting-Wei Hsu, Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021. Semantics-preserved data augmentation for aspect-based sentiment analysis. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4417–4422, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Binxuan Huang and Kathleen Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5469–5477, Hong Kong, China. Association for Computational Linguistics.

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 the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6280–6285, Hong Kong, China. Association for Computational Linguistics.

Chengxi Li, Feiyu Gao, Jiajun Bu, Lu Xu, Xiang Chen, Yu Gu, Zirui Shao, Qi Zheng, Ningyu Zhang, Yongpan Wang, and Zhi Yu. 2021a. Sentiprompt: Sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis. *Preprint*, arXiv:2109.08306.

Lishuang Li, Yang Liu, and AnQiao Zhou. 2018. Hierarchical attention based position-aware network for aspect-level sentiment analysis. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 181–189, Brussels, Belgium. Association for Computational Linguistics.

Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard Hovy. 2021b. Dual graph convolutional networks for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6319–6329, Online. Association for Computational Linguistics.

Zhengyan Li, Yicheng Zou, Chong Zhang, Qi Zhang, and Zhongyu Wei. 2021c. Learning implicit sentiment in aspect-based sentiment analysis with supervised contrastive pre-training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 246–256, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Bin Liang, Hang Su, Lin Gui, Erik Cambria, and Ruifeng Xu. 2022. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowledge-Based Systems*, 235:107643.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Preprint*, arXiv:1907.11692.

Jihong Ouyang, Zhiyao Yang, Silong Liang, Bing Wang, Yimeng Wang, and Ximing Li. 2024. Aspect-based sentiment analysis with explicit sentiment augmentations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):18842–18850.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Targeted sentiment classification with attentional encoder network. *Preprint*, arXiv:1902.09314.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019. Aspect-level sentiment analysis via convolution over dependency tree. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5679–5688, Hong Kong, China. Association for Computational Linguistics.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective LSTMs for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307, Osaka, Japan. The COLING 2016 Organizing Committee.

Bing Wang, Liang Ding, Qihuang Zhong, Ximing Li, and Dacheng Tao. 2022a. A contrastive cross-channel data augmentation framework for aspect-based sentiment analysis. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6691–6704, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Haining Wang, Kang He, Bobo Li, Lei Chen, Fei Li, Xu Han, Chong Teng, and Donghong Ji. 2024a. Refining and synthesis: A simple yet effective data augmentation framework for cross-domain aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10318–10329, Bangkok, Thailand. Association for Computational Linguistics.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238, Online. Association for Computational Linguistics.

Qianlong Wang, Keyang Ding, Bin Liang, Min Yang, and Ruifeng Xu. 2023. Reducing spurious correlations in aspect-based sentiment analysis with explanation from large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2930–2941, Singapore. Association for Computational Linguistics.

Tianlu Wang, Rohit Sridhar, Diyi Yang, and Xuezhi Wang. 2022b. Identifying and mitigating spurious correlations for improving robustness in NLP models. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1719–1729, Seattle, United States. Association for Computational Linguistics.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas. Association for Computational Linguistics.

Zengzhi Wang, Qiming Xie, Yi Feng, Zixiang Ding, Zinong Yang, and Rui Xia. 2024b. Is chatgpt a good sentiment analyzer? a preliminary study. *Preprint*, arXiv:2304.04339.

Xiaoyu Xing, Zhijing Jin, Di Jin, Bingning Wang, Qi Zhang, and Xuanjing Huang. 2020. Tasty burgers, soggy fries: Probing aspect robustness in aspect-based sentiment analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3594–3605, Online. Association for Computational Linguistics.

Hongling Xu, Qianlong Wang, Yice Zhang, Min Yang, Xi Zeng, Bing Qin, and Ruifeng Xu. 2024. Improving in-context learning with prediction feedback for sentiment analysis. *Preprint*, arXiv:2406.02911.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.

Heng Yang and Ke Li. 2024. Modeling aspect sentiment coherency via local sentiment aggregation. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 182–195, St. Julian's, Malta. Association for Computational Linguistics.

Zheng Zhang, Zili Zhou, and Yanna Wang. 2022. SSEGCN: Syntactic and semantic enhanced graph convolutional network for aspect-based sentiment analysis. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4916–4925, Seattle, United States. Association for Computational Linguistics.

# Appendix

## A Case Study

| | |
|---|---|
| | **Case 1** |
| **Input** | **Sentence**: The unfortunate lady next to us thought she had ordered a salad a (including asking for salad dressing) and was instead given a quesedilla.<br>**Clarification**: A quesadilla is a Mexican dish consisting of a tortilla filled with cheese, and often other ingredients, then folded in half and grilled. In this context, it refers to a mix-up in the lady's order, receiving a quesadilla instead of a salad. |
| **Lable** | Neutral |
| **Prediction** | The sentiment towards "quesadilla" in the given sentence is **neutral** because it simply describes a mistake in the lady's order, without expressing any positive or negative emotion. The term "quesadilla" is used in a factual manner to describe the substitution, without conveying any specific feelings or opinions about the dish itself. Therefore, the sentiment expressed by the aspect term "quesadilla" remains neutral in this context. |
| | **Case 2** |
| **Input** | **Sentence**: The guac is fresh, yet lacking flavor, we like to add our fresh salsa into it.<br>**Clarification**: Fresh salsa refers to a type of salsa made with freshly chopped ingredients such as tomatoes, onions, peppers, and cilantro. It is typically uncooked and known for its vibrant, crisp flavors that complement dishes like guacamole. |
| **Lable** | Neutral |
| **Prediction** | The sentiment towards "fresh salsa" in the given sentence is **neutral** because the term "fresh" does not inherently convey positive or negative emotions. It simply refers to the type of salsa used, without expressing any particular sentiment towards it. The use of "fresh" suggests a positive aspect, but the overall sentiment is neutral as it does not express any particularly positive or negative feelings towards the fresh salsa itself. |

Figure 6: Examples with neutral sentiment that are easily mispredicted. Flan-T5 incorrectly predicts them as negative and positive respectively, while Flan-T5-MDE correctly identifies them all.

In the Fig. 6, we present two examples where Flan-T5 incorrectly predicts sentiment, highlighting the cues that are prone to misinterpretation. For instance, the word "*unfortunate*" in the first case tend to bias the sentiment towards the negative. In the second case, the term "*fresh*" commonly associated with positive sentiment regarding ingredient quality. Flan-T5-MDE effectively recognizes these aspect terms as neutral, providing clear reasoning that demonstrates the model's ability to accurately identify targets and deliver coherent explanations.

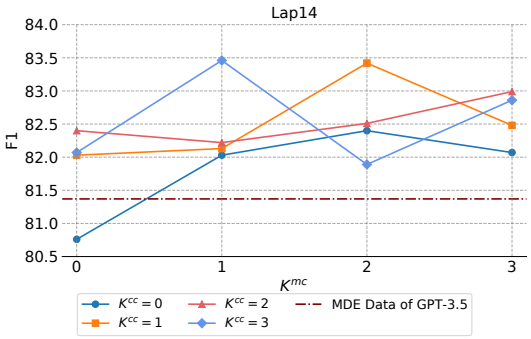


Figure 7: Results of T5 for different $K^{cc}$ and $K^{mc}$ on the Lap14 dataset constructed using GPT-4o.

| Method | Pos. | Neu. | Neg. | Overall |
|---|---|---|---|---|
| | Rest14 | | | |
| Llama-2-7B | 87.90 | 37.89 | 67.46 | 64.42 |
| Llama-3-8B | 90.60 | 44.38 | 78.59 | 71.19 |
| GPT-3.5 | 91.42 | 40.79 | 78.38 | 70.20 |
| GPT-4o | 93.90 | 47.48 | 82.93 | 74.77 |
| Flan-T5-MDE | **94.96** | **74.49** | **88.42** | **88.96** |
| | Lap14 | | | |
| Llama-2-7B | 84.43 | 48.07 | 62.80 | 65.10 |
| Llama-3-8B | 87.06 | 55.10 | 76.82 | 72.99 |
| GPT-3.5 | 88.52 | 55.51 | 79.60 | 74.54 |
| GPT-4o | 88.61 | 48.74 | 77.99 | 71.78 |
| Flan-T5-MDE | **91.78** | **72.29** | **81.46** | **82.01** |

Table 10: Zero-shot experiment results. The table reports the F1 score for each class and the overall macro-F1 score. Flan-T5-MDE is our fine-tuned model.

Additionally, we notice an inconsistency in the explanation of the second example, initially stating, "*fresh does not inherently convey positive or negative emotion*", while later suggesting "*The use of fresh implies positivity*". These two statements convey contradictory meanings, yet the model still accurately identifies the sentiment as neutral.

## B MDE with GPT-4o

MDE data used in the main experiments is constructed with GPT-3.5. Since both Step 1 and Step 3 of MDE rely on LLMs, we further validate MDE using the updated model, GPT-4o [6]. To manage API costs, we conduct validation on the Lap14 dataset. Fig. 7 presents the results under various $K^{cc}$ and $K^{mc}$ combinations, with the best result from the GPT-3.5-based MDE data as the baseline. MDE data constructed with GPT-4o generally outperforms that from GPT-3.5.

Table 16 compares the MDE-enhanced Lap14 data constructed by both models. GPT-4o generates longer clarifications and reasonings. We also evaluate aspect term substring ratio ($P_{sub}$), co-occurrence ratio ($P_{co}$), and semantic similarity between new and original aspects. $P_{sub}$ for GPT-4o is 94.01%, higher than GPT-3.5's 90.46%, indicating better substring accuracy. $P_{co}$ is 65.38%, slightly lower than GPT-3.5's 70.33%, while the cosine similarity between 0.9 and 1, is 58.19%, nearly identical to GPT-3.5's 58.42%. These results suggest that GPT-4o uses semantically similar but more diversified expressions, leading to improved model performance.

## C Zero-shot Experiments

We conduct zero-shot experiments with Llama-2-7B, Llama-3-8B, GPT-3.5, and GPT-4o on Rest14 and Lap14 datasets. The specific prompts used are detailed in Table 18. The experimental results are presented in Table 10, where we report the F1 scores for the three sentiment categories as well as the overall F1 score. For comparison, we also include the results of Flan-T5-MDE. All three models exhibit similar sentiment recognition biases: their performance in classifying positive sentiment is the best, nearly reaching the level of our fine-tuned method, but their performance in recognizing neutral sentiment is significantly lower than the other categories. The experimental results suggest that directly leveraging LLMs for data synthesis and sentiment annotation is not feasible.

## D MDE Data Details

### D.1 Confidence Distribution.

We train an ABSA classifier using the original dataset and subsequently applied it to classify the paraphrased data. We compute the confidence scores for both correctly and incorrectly classified instances, segmented into various intervals as illustrated in Table 11. Remarkably, irrespective of the correctness of classification, the confidence scores predominantly reside in high intervals, indicating a strong conviction in the classifier's predictions. This high level of confidence suggests that the paraphrased sentences exhibit a high degree of semantic similarity to the original training data, which aligns with our expectations. The paraphrased sentences maintain the same sentiment polarity towards the same targets without significant deviation from the original domain.

Moreover, the concentration of confidence

[6] version: gpt-4o-mini-2024-07-18

| | Interval | #Correct | #Incorrect |
|---|---|---|---|
| **Rest14** | [0, 0.9) | 832 | 871 |
| | [0.9, 0.99) | 1296 | 901 |
| | [0.99, 1] | 29190 | 2930 |
| | Total | 31318 | 4702 |
| **Lap14** | [0, 0.9) | 390 | 334 |
| | [0.9, 0.99) | 421 | 347 |
| | [0.99, 1] | 19786 | 1852 |
| | Total | 20597 | 2533 |
| **Rest15** | [0, 0.9) | 50 | 90 |
| | [0.9, 0.99) | 87 | 73 |
| | [0.99, 1] | 11243 | 497 |
| | Total | 11380 | 660 |
| **Rest16** | [0, 0.9) | 104 | 198 |
| | [0.9, 0.99) | 198 | 210 |
| | [0.99, 1] | 15885 | 885 |
| | Total | 16187 | 1293 |

Table 11: Statistics of paraphrased data across varying confidence intervals.

scores in high intervals underscores our rationale for adopting a Top-K filtering strategy. A threshold-based filtering approach would be impractical due to the challenge of selecting an optimal threshold that balances the dataset volume and quality. The observed distribution of confidence scores substantiates the feasibility and efficacy of Top-K filtering.

### D.2 Data Details in Different $K^{cc}$ and $K^{mc}$.

During the data filtering stage, we introduce two hyperparameters, $K^{cc}$ and $K^{mc}$. $K^{cc}$ represents the number of samples selected from the set of correctly classified instances, while $K^{mc}$ denotes the number of samples chosen from the set of incorrectly classified neutral instances. In our experiments, we vary these parameters from 0 to 3. Table 16 and 17 provide a detailed breakdown of the number of instances for each class, along with the total number of instances. Additionally, we measure the average length of sentence, aspect clarifications and sentiment reasonings. The original review sentences are very short and contain limited information, the clarifications enrich the content with prior knowledge. The lengths of clarifications and reasonings remain relatively stable across different datasets, indicating that the information generated by LLMs is both consistent and effective.

### D.3 Experiments of Different $K^{cc}$ and $K^{mc}$

We conducted experiments using different combinations of $K^{cc}$ and $K^{mc}$ on both T5 and Flan-T5. The results of T5 for the Rest14 and Lap14 datasets are presented in the main content. The rest of experimental results are shown in Fig. 8 and 9. When $K^{cc}$ is set to 0, the model's performance may slightly lag behind the baseline without enhanced data. However, when both $K^{cc}$ and $K^{mc}$ are greater than 0, the models consistently outperform the baseline.

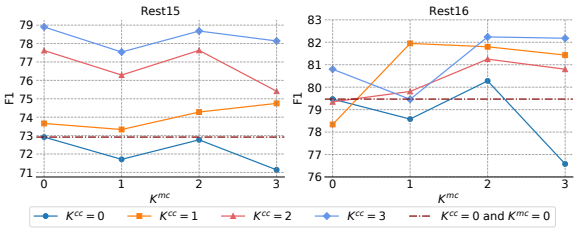


Figure 8: Results of T5 for different $K^{cc}$ and $K^{mc}$.

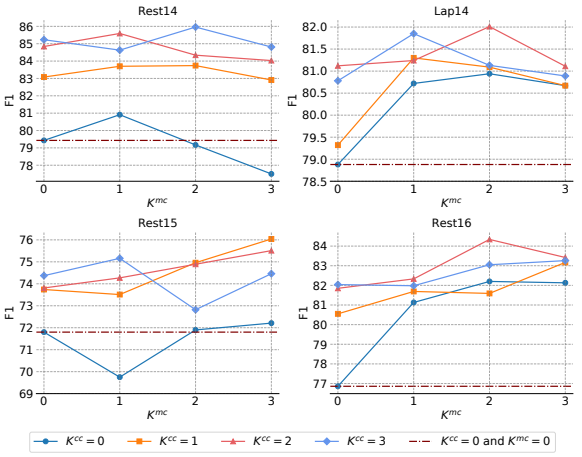


Figure 9: Results of Flan-T5 for different $K^{cc}$ and $K^{mc}$.

| Method | Rest14 | | Lap14 | |
|---|---|---|---|---|
| | $\text{All}_{F1}$ | $\text{ISA}_{F1}$ | $\text{ALL}_{F1}$ | $\text{ISA}_{F1}$ |
| Flan-T5 | 79.91 | 69.16 | 78.96 | 73.81 |
| Flan-T5-THOR | 82.98 | 71.70 | 79.75 | 67.63 |
| Flan-T5-ESA | 83.79 | 73.76 | 81.78 | 77.91 |
| Flan-T5-MDE | **85.96** | **73.84** | **82.01** | **80.23** |

Table 12: Experimental results of implicit sentiment analysis.

## E Implicit Sentiment Analysis (ISA).

In the Rest14 and Lap14 datasets, aspect terms lacking explicit sentiment expression are marked as implicit sentiments (Li et al., 2021c). We compare our approach with THOR (Fei et al., 2023) and ESA, both based on Flan-T5. THOR employs chain-of-thought prompting to offer additional insights for implicit sentiment analysis. As shown in Table 12, MDE achieves the highest F1 scores for both overall and implicit sentiment data. These re-

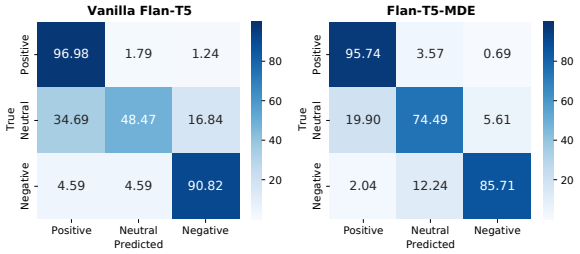


Figure 10: Confusion matrices of Rest14 test set.

| Dataset | Pos. | Neu. | Neg. | Overall |
|---|---|---|---|---|
| Rest14 | 93.45 | 60.70 | 85.58 | 79.91 |
| +MDE | **94.96** | **74.49** | **88.42** | **85.96** |
| Lap14 | 91.01 | 65.74 | 80.13 | 78.96 |
| +MDE | **91.78** | **72.79** | **81.46** | **82.01** |
| Rest15 | 91.12 | 46.67 | 83.76 | 74.18 |
| +MDE | **92.75** | **50.00** | **85.38** | **76.04** |
| Rest16 | 97.47 | 54.55 | **90.83** | 80.95 |
| +MDE | **97.56** | **66.67** | 88.79 | **84.34** |

Table 13: Breakdown of F1 Performance for Flan-T5 on original and MDE-enhanced datasets.

| Method | Rest14 | | Lap14 | |
|---|---|---|---|---|
| | Acc | F1 | Acc | F1 |
| Flan-T5 | 87.41 | 79.91 | 82.76 | 78.96 |
| + PF | 87.95 | 81.20 | 82.76 | 79.21 |
| + AC | 89.20 | 83.04 | 81.82 | 77.54 |
| + SR | 88.04 | 81.69 | 83.07 | 79.13 |
| + AC & SR | 87.14 | 79.43 | 82.92 | 78.88 |
| + MDE | **90.27** | **85.96** | **84.80** | **82.01** |

Table 14: Experimental results of Flan-T5 using partial or full MDE data.

sults underscore that a diverse training dataset can effectively capture various sentiment expressions.

## F Implement Details for BERT

We replicate BERT to solve ABSA using the bert-base-uncased. The input to the model is formatted as {`[CLS]` $s_i$ $c_i$ `[SEP]` $a_i$}, where $s_i$ represents the input sentence, $c_i$ denotes the aspect clarification, and $a_i$ corresponds to the aspect term. It is important to note that in the vanilla BERT implementation, $c_i$ is not included in the input sequence. The model is fine-tuned for 10 epochs using the AdamW optimizer, with a learning rate set to 1e-5.

## G Prompt Templates

In evaluating the ABSA performance of LLMs and various stages of the MDA method, we utilize LLMs. Table 18 lists the prompts used in each step. Our prompt design is generally divided into two parts: task description and output format. In the sentiment reasoning step of MDE framework, we include sentiment labels in the prompt to ensure that the generated explanatory information aligns with the sentiment labels.

## H Other Experimental Results of Flan-T5

Due to page limitations, some Flan-T5 experimental results are not included in the main content. They exhibit similar characteristics to the T5 model, leading to the same conclusions. These results are listed here for reference and to demonstrate the generalizability of MDE.

- Performance breakdown by sentiment: Table 13 presents Flan-T5's performance across different sentiment classes before and after applying MDE data. Fig. 10 shows the confusion matrix results for Flan-T5 on the Rest14 test set.
- Flan-T5-MDE ablation study: Table 14 shows the impact of using different subsets of MDE data on Flan-T5. Full MDE data ensures diversity, accuracy, and logical consistency, effectively unlocking the model's potential for optimal performance.
- Impact of correctly classified (CC) and misclassified (MC) data: Table 15 shows model performance when using only CC, only MC, or both types of data.

| Method | Rest14 | | | | Lap14 | | | |
|---|---|---|---|---|---|---|---|---|
| | Pos. | Neu. | Neg. | Overall | Pos. | Neu. | Neg. | Overall |
| Flan-T5 ($K^{cc}=0, K^{mc}=0$) | 93.52 | 60.38 | 84.39 | 79.43 | 91.30 | 55.85 | 79.87 | 78.88 |
| w/ MC Data ($K^{cc}=0$) | 93.53 | 61.58 | 82.41 | 79.17 | 90.52 | 71.26 | 81.02 | 80.94 |
| w/ CC Data ($K^{mc}=0$) | 94.60 | 73.63 | 83.67 | 85.23 | 91.51 | 70.95 | 80.91 | 81.12 |
| w/ PF Data ($K^{cc}\neq 0, K^{mc}\neq 0$) | **94.96** | **74.49** | **88.42** | **85.96** | **91.78** | **72.29** | **81.46** | **82.01** |

Table 15: Experimental results under different filtering settings, reporting the F1 for each category and the overall macro-F1.

| | Pos. | Neu. | Neg. | Total | Scale | Len. of S | Len. of C | Len. of R |
|---|---|---|---|---|---|---|---|---|
| | **Lap14-GPT-3.5** | | | | | | | |
| $K^{cc}=0, K^{mc}=0$ | 987 | 460 | 866 | 2313 | 1.00 | 19.31 | 36.25 | 70.67 |
| $K^{cc}=0, K^{mc}=1$ | 987 | 753 | 866 | 2606 | 1.13 | 19.29 | 36.37 | 70.20 |
| $K^{cc}=0, K^{mc}=2$ | 987 | 1003 | 866 | 2856 | 1.23 | 19.30 | 36.50 | 69.86 |
| $K^{cc}=0, K^{mc}=3$ | 987 | 1214 | 866 | 3064 | 1.32 | 19.28 | 36.58 | 69.56 |
| $K^{cc}=1, K^{mc}=0$ | 1965 | 853 | 1714 | 4532 | 1.96 | 18.47 | 36.28 | 69.83 |
| $K^{cc}=1, K^{mc}=1$ | 1965 | 1146 | 1714 | 4825 | 2.09 | 18.51 | 36.34 | 69.63 |
| $K^{cc}=1, K^{mc}=2$ | 1965 | 1396 | 1714 | 5075 | 2.19 | 18.55 | 36.28 | 69.83 |
| $K^{cc}=1, K^{mc}=3$ | 1965 | 1714 | 1714 | 5286 | 2.29 | 18.57 | 36.47 | 69.30 |
| $K^{cc}=2, K^{mc}=0$ | 2939 | 1222 | 2558 | 6719 | 2.90 | 18.16 | 36.43 | 69.63 |
| $K^{cc}=2, K^{mc}=1$ | 2939 | 1515 | 2558 | 7012 | 3.03 | 18.20 | 36.47 | 69.50 |
| $K^{cc}=2, K^{mc}=2$ | 2939 | 1765 | 2558 | 7262 | 3.14 | 18.24 | 36.52 | 69.39 |
| $K^{cc}=2, K^{mc}=3$ | 2939 | 1976 | 2558 | 7473 | 3.23 | 18.26 | 36.55 | 69.28 |
| $K^{cc}=3, K^{mc}=0$ | 3911 | 1582 | 3398 | 8891 | 3.84 | 18.04 | 36.52 | 69.47 |
| $K^{cc}=3, K^{mc}=1$ | 3911 | 1875 | 3398 | 9184 | 3.97 | 18.08 | 36.55 | 69.37 |
| $K^{cc}=3, K^{mc}=2$ | 3911 | 2125 | 3398 | 9434 | 4.08 | 18.11 | 36.58 | 69.29 |
| $K^{cc}=3, K^{mc}=3$ | 3911 | 2396 | 3398 | 9645 | 4.17 | 18.13 | 36.61 | 69.20 |
| | **Lap14-GPT-4o** | | | | | | | |
| $K^{cc}=0, K^{mc}=0$ | 987 | 460 | 866 | 2313 | 1.00 | 19.31 | 45.68 | 83.07 |
| $K^{cc}=0, K^{mc}=1$ | 987 | 710 | 866 | 2563 | 1.11 | 19.36 | 45.72 | 82.98 |
| $K^{cc}=0, K^{mc}=2$ | 987 | 894 | 866 | 2747 | 1.19 | 19.39 | 45.79 | 82.94 |
| $K^{cc}=0, K^{mc}=3$ | 987 | 1042 | 866 | 2895 | 1.25 | 19.40 | 45.82 | 82.92 |
| $K^{cc}=1, K^{mc}=0$ | 1968 | 896 | 1717 | 4581 | 1.98 | 18.86 | 45.51 | 83.26 |
| $K^{cc}=1, K^{mc}=1$ | 1968 | 1146 | 1717 | 4831 | 2.09 | 18.91 | 45.55 | 83.20 |
| $K^{cc}=1, K^{mc}=2$ | 1968 | 1330 | 1717 | 5015 | 2.17 | 18.95 | 45.59 | 83.17 |
| $K^{cc}=1, K^{mc}=3$ | 1968 | 1478 | 1717 | 5163 | 2.23 | 18.96 | 45.61 | 83.15 |
| $K^{cc}=2, K^{mc}=0$ | 2947 | 1324 | 2553 | 6824 | 2.95 | 18.71 | 45.46 | 83.30 |
| $K^{cc}=2, K^{mc}=1$ | 2947 | 1574 | 2553 | 7074 | 3.06 | 18.76 | 45.48 | 83.26 |
| $K^{cc}=2, K^{mc}=2$ | 2947 | 1758 | 2553 | 7258 | 3.14 | 18.78 | 45.51 | 83.24 |
| $K^{cc}=2, K^{mc}=3$ | 2947 | 1906 | 2553 | 7406 | 3.20 | 18.80 | 45.53 | 83.22 |
| $K^{cc}=3, K^{mc}=0$ | 3921 | 1582 | 3379 | 9038 | 3.91 | 18.64 | 45.42 | 83.34 |
| $K^{cc}=3, K^{mc}=1$ | 3921 | 1988 | 3379 | 9288 | 4.02 | 18.67 | 45.44 | 83.31 |
| $K^{cc}=3, K^{mc}=2$ | 3921 | 2172 | 3379 | 9472 | 4.10 | 18.69 | 45.46 | 83.29 |
| $K^{cc}=3, K^{mc}=3$ | 3921 | 2320 | 3379 | 9620 | 4.16 | 18.71 | 45.44 | 83.28 |

Table 16: Details of MDE-enhanced Lap14 constructed using GPT-3.5 and GPT-4o under different $K^{cc}$ and $K^{mc}$ conditions. "Scale" indicates the ratio by which the original dataset is expanded. "Len. of S", "Len. of C" and "Len. of R" refers to the average length of, sentence, aspect clarification and sentiment reasonings respectively.

| | Pos. | Neu. | Neg. | Total | Scale | Len. of S | Len. of C | Len. of R |
|---|---|---|---|---|---|---|---|---|
| | | | | | **Rest14** | | | |
| $K^{cc}=0, K^{mc}=0$ | 2164 | 633 | 805 | 3602 | 1.00 | 17.57 | 35.08 | 72.41 |
| $K^{cc}=0, K^{mc}=1$ | 2164 | 1098 | 805 | 4067 | 1.13 | 17.64 | 35.11 | 71.99 |
| $K^{cc}=0, K^{mc}=2$ | 2164 | 1513 | 805 | 4482 | 1.24 | 17.70 | 35.15 | 71.66 |
| $K^{cc}=0, K^{mc}=3$ | 2164 | 1886 | 805 | 4855 | 1.35 | 17.77 | 35.24 | 71.46 |
| $K^{cc}=1, K^{mc}=0$ | 4307 | 1138 | 1585 | 7030 | 1.95 | 17.57 | 35.17 | 71.97 |
| $K^{cc}=1, K^{mc}=1$ | 4307 | 1603 | 1585 | 7495 | 2.08 | 17.61 | 35.18 | 71.76 |
| $K^{cc}=1, K^{mc}=2$ | 4307 | 2018 | 1585 | 7910 | 2.20 | 17.65 | 35.20 | 71.59 |
| $K^{cc}=1, K^{mc}=3$ | 4307 | 2391 | 1585 | 8283 | 2.30 | 17.69 | 35.25 | 71.48 |
| $K^{cc}=2, K^{mc}=0$ | 6364 | 1600 | 2352 | 10386 | 2.88 | 17.69 | 35.23 | 71.86 |
| $K^{cc}=2, K^{mc}=1$ | 6364 | 2065 | 2352 | 10851 | 3.01 | 17.63 | 35.24 | 71.72 |
| $K^{cc}=2, K^{mc}=2$ | 6364 | 2480 | 2352 | 11266 | 3.13 | 17.65 | 35.25 | 71.60 |
| $K^{cc}=2, K^{mc}=3$ | 6364 | 2853 | 2352 | 11639 | 3.23 | 17.68 | 35.28 | 71.52 |
| $K^{cc}=3, K^{mc}=0$ | 8553 | 2033 | 3108 | 13694 | 3.80 | 17.59 | 35.22 | 71.92 |
| $K^{cc}=3, K^{mc}=1$ | 8553 | 2498 | 3108 | 14159 | 3.93 | 17.61 | 35.23 | 71.81 |
| $K^{cc}=3, K^{mc}=2$ | 8553 | 2913 | 3108 | 14574 | 4.05 | 17.63 | 35.24 | 71.71 |
| $K^{cc}=3, K^{mc}=3$ | 8553 | 3286 | 3108 | 14947 | 4.15 | 17.66 | 35.26 | 71.65 |
| | | | | | **Rest15** | | | |
| $K^{cc}=0, K^{mc}=0$ | 912 | 36 | 256 | 1204 | 1.00 | 16.99 | 35.09 | 71.64 |
| $K^{cc}=0, K^{mc}=1$ | 912 | 69 | 256 | 1237 | 1.03 | 16.95 | 35.10 | 71.54 |
| $K^{cc}=0, K^{mc}=2$ | 912 | 98 | 256 | 1266 | 1.05 | 16.93 | 35.09 | 71.47 |
| $K^{cc}=0, K^{mc}=3$ | 912 | 124 | 256 | 1292 | 1.07 | 16.89 | 35.09 | 71.40 |
| $K^{cc}=1, K^{mc}=0$ | 1818 | 65 | 506 | 2389 | 1.98 | 16.38 | 34.81 | 72.03 |
| $K^{cc}=1, K^{mc}=1$ | 1818 | 98 | 506 | 2422 | 2.03 | 16.37 | 34.82 | 71.97 |
| $K^{cc}=1, K^{mc}=2$ | 1818 | 127 | 506 | 2451 | 2.36 | 16.37 | 34.82 | 71.93 |
| $K^{cc}=1, K^{mc}=3$ | 1818 | 153 | 506 | 2477 | 2.06 | 16.35 | 34.82 | 71.89 |
| $K^{cc}=2, K^{mc}=0$ | 2721 | 87 | 754 | 3562 | 2.96 | 16.16 | 34.74 | 72.03 |
| $K^{cc}=2, K^{mc}=1$ | 2721 | 120 | 754 | 3595 | 2.99 | 16.15 | 34.75 | 71.99 |
| $K^{cc}=2, K^{mc}=2$ | 2721 | 149 | 754 | 3624 | 3.01 | 16.15 | 34.75 | 71.97 |
| $K^{cc}=2, K^{mc}=3$ | 2721 | 175 | 754 | 3650 | 3.03 | 16.14 | 34.75 | 71.94 |
| $K^{cc}=3, K^{mc}=0$ | 3624 | 108 | 999 | 4731 | 3.93 | 16.04 | 34.69 | 72.06 |
| $K^{cc}=3, K^{mc}=1$ | 3624 | 141 | 999 | 4764 | 3.96 | 16.04 | 34.69 | 72.03 |
| $K^{cc}=3, K^{mc}=2$ | 3624 | 170 | 999 | 4793 | 3.98 | 16.04 | 34.69 | 72.01 |
| $K^{cc}=3, K^{mc}=3$ | 3624 | 196 | 999 | 4819 | 4.00 | 16.04 | 34.63 | 71.91 |
| | | | | | **Rest16** | | | |
| $K^{cc}=0, K^{mc}=0$ | 1240 | 69 | 439 | 1748 | 1.00 | 17.34 | 34.87 | 71.09 |
| $K^{cc}=0, K^{mc}=1$ | 1240 | 124 | 439 | 1803 | 1.03 | 17.26 | 34.93 | 70.88 |
| $K^{cc}=0, K^{mc}=2$ | 1240 | 174 | 439 | 1853 | 1.06 | 17.19 | 34.92 | 70.77 |
| $K^{cc}=0, K^{mc}=3$ | 1240 | 220 | 439 | 1899 | 1.09 | 17.12 | 34.94 | 70.74 |
| $K^{cc}=1, K^{mc}=0$ | 2473 | 124 | 855 | 3452 | 1.97 | 16.55 | 34.76 | 71.32 |
| $K^{cc}=1, K^{mc}=1$ | 2473 | 179 | 855 | 3507 | 2.01 | 16.52 | 34.76 | 71.25 |
| $K^{cc}=1, K^{mc}=2$ | 2473 | 224 | 855 | 3557 | 2.03 | 16.49 | 34.76 | 71.19 |
| $K^{cc}=1, K^{mc}=3$ | 2473 | 275 | 855 | 3603 | 2.06 | 16.47 | 34.77 | 71.17 |
| $K^{cc}=2, K^{mc}=0$ | 3702 | 173 | 1265 | 5140 | 2.94 | 16.27 | 34.68 | 71.4 |
| $K^{cc}=2, K^{mc}=1$ | 3702 | 228 | 1265 | 5195 | 2.97 | 16.25 | 34.68 | 71.35 |
| $K^{cc}=2, K^{mc}=2$ | 3702 | 278 | 1265 | 5245 | 3.00 | 16.24 | 34.68 | 71.31 |
| $K^{cc}=2, K^{mc}=3$ | 3702 | 324 | 1265 | 5291 | 3.03 | 16.22 | 34.69 | 71.29 |
| $K^{cc}=3, K^{mc}=0$ | 4972 | 220 | 1666 | 6813 | 3.90 | 16.13 | 34.68 | 71.53 |
| $K^{cc}=3, K^{mc}=1$ | 4972 | 275 | 1666 | 6868 | 3.93 | 16.12 | 34.68 | 71.49 |
| $K^{cc}=3, K^{mc}=2$ | 4972 | 325 | 1666 | 6918 | 3.96 | 16.11 | 34.68 | 71.46 |
| $K^{cc}=3, K^{mc}=3$ | 4972 | 371 | 1666 | 6964 | 3.98 | 16.10 | 34.69 | 71.45 |

Table 17: Details of MDE-enhanced Rest14, Rest15 and Rest16 constructed using GPT-3.5 under different $K^{cc}$ and $K^{mc}$ conditions. "Scale" indicates the ratio by which the original dataset is expanded. "Len. of S", "Len. of C" and "Len. of R" refers to the average length of, sentence, aspect clarification and sentiment reasonings respectively.

| Task | Prompt |
|---|---|
| Evaluation of ABSA | Analyze the sentiment polarity towards a specified aspect within a given sentence.<br>Input Components:<br>The sentence to be analyzed.<br>The specific aspect within the sentence that the sentiment analysis should focus on.<br>Formatted Output:<br>At the end of the output, provide a formatted result as follows: Final Result: The sentiment towards the aspect is [positive, negative, or neutral].<br>Sentence: {**sentence**}<br>Aspect: {**aspect**} |
| Semantic Paraphrasing | Generate 10 review sentences about {**domain**} that convey a similar meaning to the provided review sentence: "{**sentence**}". Each sentence should capture the essence of the original review while presenting it in a different way.<br>Output Format:<br>Sentence i: {sentence} |
| Aspect Annotation | Provide aspect annotations for ten sentences that convey a similar meaning to the given source sentence. The source sentence includes an annotated aspect term. Your task is to identify and annotate the aspect term within each of the ten sentences, ensuring that the aspect term is a subsequence within its sentence and that carries a similar meaning to the source aspect term.<br>Just output the aspect of each sentence in the following format:<br>Aspect i: {aspect for sentence i}<br>Input:<br>Source Sentence: {**sentence**}<br>Source Aspect: {**aspect**}<br>Sentence {**i**}: {**new sentence**$_i$}<br>(Note: Ten paraphrased sentences are listed here.) |
| Aspect Clarification & Sentiment Reasoning | **Turn 1**<br>Describe in 20 to 60 words the meaning of the term "{**aspect**}" as it is used in the context of the sentence "{**sentence**}". |
| | **Turn 2**<br>In the provided sentence "{**sentence**}", explain why the sentiment expressed by the aspect term "{**aspect**}" is {**label**}. Your explanation is limited to 100 words.<br>Output Format:<br>The sentiment towards "{**aspect**}" in the given sentence is {**label**} because ... |

Table 18: The prompt templates used in LLMs have placeholders marked by {**bold text**} that need to be replaced. Aspect Clarification and Sentiment Reasoning is a multi-turn dialogue process.