



PDF Download
3746252.3761223.pdf
12 January 2026
Total Citations: 0
Total Downloads: 125

 Latest updates: <https://dl.acm.org/doi/10.1145/3746252.3761223>

RESEARCH-ARTICLE

Harnessing Commonsense: LLM-Driven Knowledge Integration for Fine-Grained Sentiment Analysis

KAI ZHANG, University of Science and Technology of China, Hefei, Anhui, China

YUPENG HAN, University of Science and Technology of China, Hefei, Anhui, China

Open Access Support provided by:

University of Science and Technology of China

Published: 10 November 2025

Citation in BibTeX format

CIKM '25: The 34th ACM International
Conference on Information and
Knowledge Management
November 10 - 14, 2025
Seoul, Republic of Korea

Conference Sponsors:

SIGWEB
SIGIR

Harnessing Commonsense: LLM-Driven Knowledge Integration for Fine-Grained Sentiment Analysis

Kai Zhang

State Key Laboratory of Cognitive Intelligence, University
of Science and Technology of China
Hefei, China
kkzhang08@ustc.edu.cn

Yupeng Han[†]

State Key Laboratory of Cognitive Intelligence, University
of Science and Technology of China
Hefei, China
yupenghan@mail.ustc.edu.cn

Abstract

Fine-grained sentiment analysis, which aims to identify sentiments associated with specific aspects within sentences, faces challenges in effectively incorporating commonsense knowledge. Recent advancements leveraging large language models (LLMs) as data generators show promise but are limited by the LLMs' lack of nuanced, domain-specific understanding and pose a significant risk of data leakage during inference, potentially leading to inflated performance metrics. To address these limitations, we propose LLM-Kit, a novel framework for commonsense-enhanced fine-grained sentiment analysis that integrates knowledge via LLM-guided graph construction, effectively mitigating data leakage risks. LLM-Kit operates in two key stages: (1) Commonsense Graph Construction (CGC): We design second-order rules and leverage LLMs for evaluation to ensure the accuracy of the generated graph and mitigate the risk of data leakage from LLMs. (2) Knowledge-integration Graph Representation Learning (KGRL): We extract knowledge that is aware of various aspects through Graph Representation Learning (GRL). To capture the underlying semantic nuances within the input sentence, we develop a Sentence Semantic Learning (SSL) module based on RoBERTa that explicitly encodes internal semantics. This module provides complementary information to the GCN, improving the model's ability to discern subtle sentiment variations related to different aspects. Comprehensive experiments on three public datasets affirm that LLM-Kit achieves comparable performance with state-of-the-art models.

CCS Concepts

• Information systems → Sentiment analysis; Retrieval tasks and goals.

Keywords

Web Semantic Mining; Knowledge Graph; Sentiment Analysis

ACM Reference Format:

Kai Zhang and Yupeng Han[†]. 2025. Harnessing Commonsense: LLM-Driven Knowledge Integration for Fine-Grained Sentiment Analysis. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM '25)*, November 10–14, 2025, Seoul, Republic of Korea. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3746252.3761223>

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) is a fine-grained task in natural language processing (NLP) that dissects sentences to determine the sentiment polarities of specific aspects. Depending on whether the distinct components are explicitly mentioned in the review sentences, this task can be divided into two subcategories: Aspect Term Sentiment Analysis (ATSA) [26] and Aspect Category Sentiment Analysis (ACSA) [26].

For instance, given a review sentence such as “*Although the sushi is delicious, the waiter is rude!*”, it encompasses two aspect terms, specifically “*sushi*” and “*waiter*”, along with two aspect categories, namely “*food#taste*” and “*restaurant#service*”. The objective of ATSA is discerning the sentiment associated with the aspect terms present within the review sentence. In contrast, ACSA targets the identification of the sentiment tendency linked to the aspect category. After elucidating the characteristics of these two tasks, we proceed with a brief overview of previously employed approaches. **For a more detailed comparison, please see Section 3.1.**

In addressing the ACSA conundrum, two primary research trajectories have been embraced within both the industrial and academic spheres [7, 17, 25, 34, 35, 39, 40]. In particular, deep learning architectures, such as attention-based models [25, 40] and pre-trained language models [17, 31, 41], advocate the learning of parameterized prediction functions through the optimization of meticulously designed model structures. These methods implicitly map the semantic relations between aspect category and its contextual words. Additionally, several LLMs-based [10, 45] and graph-based methods [10, 39, 45] explicitly encode the feature representation intrinsic to the syntactic graph and subsequently extract semantics predicated on such graphical data. Although existing methods demonstrate commendable performance, they often fall short due to two common problems that remain inadequately addressed.

The first problem is that commonsense knowledge, while important for ACSA, is often under-considered. Augmenting models via integration of knowledge graphs (KGs) has shown utility across numerous NLP applications, as KGs facilitate the association of principal terms with the actual entities, factual knowledge, and commonsense concepts. In the context of ACSA, the polarities of sentiment words may fluctuate across distinct aspect categories, a

[†] Corresponding authors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CIKM '25, Seoul, Republic of Korea

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-2040-6/2025/11
<https://doi.org/10.1145/3746252.3761223>

```

<sentence id="1358">
  <text> Although the sushi is delicious, the waiter is rude! </text>
  <aspectTerms>
    <aspectTerm term="sushi" polarity="positive" from="13" to="18"/>
    <aspectTerm term="waiter" polarity="negative" from="37" to="43"/>
  </aspectTerms>
  <aspectCategories>
    <aspectCategory category="Food#taste" polarity="positive"/>
    <aspectCategory category="Restaurant#service" polarity="negative"/>
  </aspectCategories>
</sentence>

```

Figure 1: ABSA is tasked with identifying the sentiment associated with aspect terms that emerge within sentences (e.g., “sushi”, “waiter”). On the other hand, ACSA is dedicated to recognizing the sentiment of aspect categories (categories which typically do not emerge in sentence text).

situation for which KGs are well-suited. For instance, the sentiment-bearing word “hot” can be a positive feature for aspect category “food#taste”, but a negative feature for “restaurant#environment”. Therefore, it is challenging to assign implicit polarities based solely on review text, especially in the absence of knowledge supervision. Fortunately, commonsense knowledge provides an insightful way to enhance context and assist models understand factual semantics (e.g., the word “hot” describes “food#taste” positively but “restaurant#environment” negatively). However, despite the efficiency of external commonsense KGs, ACSA has only made sporadic attempts to incorporate them.

The second problem is that LLM-based enhancement methods often suffer from data leakage issues. Large language models, such as ChatGPT and GPT-4, are versatile and can solve various tasks due to their powerful knowledge generation capabilities [16, 38]. Recent studies have directly exploited LLMs as knowledge generators to augment existing data, yielding surprisingly interesting results [5, 43]. However, because LLMs are trained on massive amounts of internet data, they may lack the domain-specific knowledge needed to perform certain tasks and may also cause data leakage during the data generation process [2]. Therefore, finding ways to use LLMs for data enhancement without causing data leakage is an urgent problem that needs to be addressed.

To tackle the above problems, we attempt to unlocking the power of commonsense through LLMs and address the data leakage challenge in the process of generating commonsense knowledge graphs. In particular, we propose an LLM-enhanced Knowledge Integration (**LLM-Kit**) framework which contain two main stages: the first stage is commonsense graph construction (**CGC**) which aims to generate aspect-aware commonsense knowledge; the second stage is knowledge-integration graph representation learning (**KGRL**) which aims to fully leverage the semantic benefits of commonsense knowledge in ACSA. **In CGC stage**, we propose a two-order extraction strategy that includes an LLM-based scoring evaluator. This strategy transforms the general KGs into aspect-oriented knowledge graphs (Asp-KGs), which capture both commonsense knowledge and the specific semantics of an aspect in a particular domain. **In KGRL stage**, we design a GRL module to capture semantics between aspect categories and opinion terms, incorporating commonsense knowledge. Additionally, a sentence semantic

learning (SSL) module with pre-trained RoBERTa explores sentence structure semantics. To merge the semantics on divergent spaces, we created a knowledge fusion function that adaptively balances their contributions. Comprehensive experiments conducted on public and real-world datasets underscore the efficacy of the proposed methodology. The source code for this method is accessible on¹.

2 Related Work

Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment analysis task that can be changed into coarser-grained aspect category sentiment analysis (ACSA) [26, 51]. Traditional methods [6, 15] have engineered numerous rule-based models for the task, deploying machine learning algorithms in conjunction with rules to ascertain sentiment polarity concerning the aspects.

DNN-based Models. In recent years, most research work employed deep learning models to examine context words surrounding the aspect categories [12, 39, 41, 44, 50]. Specifically, several studies [8, 17, 25, 40] have devised various attention mechanisms to predict the aspect’s sentiment. For example, Fan et al. [8] defined a multi-grained network, which is responsible for linking words from the aspects and sentences. Li et al. [17] designed a target-specific attention model to integrate aspects into the review sentence. In addition, most of the existing GCN-based studies [9, 10, 32, 39, 48, 50] perform aspect term sentiment analysis (ATSA) by constructing syntactic dependency graph. Wang et al. [39] reshaped dependency trees and proposed a relational GAT model to encode syntactic features. Zhong et al. [50] proposed knowledge graph augmented network to incorporate external knowledge with syntactic information. Zhao et al. [48] designed a metric-free method to model the relations among the aspects and sentences.

Pre-trained Models. In addition, pre-trained language models (PLMs) have proved remarkable success in both ABSA and ACSA tasks [12, 20, 31, 33, 41, 46, 47]. Among them, Song et al. [31] devised an attentional network and BERT-SPC to learn features between aspect and context. Tian et al. [36] proposed a knowledge enhanced pre-training for sentiment analysis. Ke et al. [12] devised a continual learning pre-trained model to effectively encode specific semantics from multiple tasks. Bu et al. [3] designed a joint model to address ACSA task and Rating Prediction task synthetically. Even though current pre-trained models have come a long way, they still rely heavily on sentence-level internal semantics (i.e., semantics from the review text) and ignore some external commonsense knowledge that are crucial for ACSA task.

LLM-based Models. More recently, we have witnessed the rapid development of large language models (LLMs) in both ABSA and ACSA tasks [19, 22, 30, 49]. Among them, a retrieval-based example mining method for instructional learning in ABSA tasks to improve the performance by selecting effective examples was proposed [49]. Li et al. [19] designed a systemic iterative data augmentation framework (IterD), which leverages the powerful ability of LLMs to generate more high-quality labeled data. Shi et al. [30] proposed an instruction tuning and bi-layer sentiment representation-based RoBERTa model (ITGCN) for aspect-based sentiment analysis. LLMs have achieved great success in understanding the sentence

¹https://github.com/Hyp26cs/LLM_Kit

semantics with single aspect category, however, faced with serious challenges in processing sentence with mixed aspect categories.

3 Preliminaries

3.1 Differences between ABSA and ACSA

As mentioned in main section, ABSA and ACSA are two different tasks, although the two tasks are similar. Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment classification task that determines the sentiment of a single aspect in a given review. As shown in Figure 1, according to whether the aspects are explicitly mentioned in the review sentences, ABSA can be subdivided into Aspect-Term Sentiment Analysis (ATSA) and Aspect-Category Sentiment Analysis (ACSA) [26]. For example, as shown in Figure 1, given a review sentence “Although the sushi is delicious, the waiter is rude!”, it contains two aspect terms (i.e., “sushi” and “waiter”), and two aspect categories (i.e., “food#taste” and “restaurant#service”) corresponding to the aspect term. The ABSA task aims to identify aspectual aspect term sentiment that appears in the review sentence. When it comes to ACSA, however, the goal is to identify sentiment tendencies of the aspect category. Note that, the aspect categories do not appear in the review sentence.

3.2 Problem Statement

Given a review text $R = \{w_1, w_2, \dots, w_L\}$ consisting of a series of L words and a set of specific aspect categories $A = \{a_1, a_2, \dots, a_N\}$ which are pre-defined in the data corpus D . The goal of ACSA is to learn a model that can precisely predict the sentiment polarity (i.e., *positive*, *neutral* and *negative*) of review R relative to the mentioned aspect category a_i where $i \in \{1, 2, \dots, N\}$. Note that L denotes the length of review, R and N is the number of pre-defined aspect categories (e.g., “food#taste” and “food#portion”) in the dataset. As mentioned before, the external knowledge graph can solve existing problems and plays a crucial role in aspect category sentiment prediction. Therefore, in this paper, we further apply additional Asp-KGs (i.e., $\mathcal{G}_N$) as input of our proposed KGRL model to encode the aspect-aware semantic information for ACSA.

4 The LLM-Kit Framework

In this section, we propose a novel two-stage LLM-enhanced framework, comprising a CGC stage and a KGRL stage. Based on the training pipeline, we will describe these two core stages in detail. For better understanding, an illustration of our LLM-Kit framework is shown in Figure 2.

4.1 CGC stage: Asp-KGs Construction

Our method begins by re-structuring a common Knowledge Graph (e.g., Freebase) into aspect-oriented knowledge graphs (Asp-KGs). Specifically, it can produce first-order neighbors (internal semantic, which denotes semantic-bearing and aspect category related words) and second-order neighbors (external commonsense knowledge, which represents additional words added by constraints). In the following, we illustrate the details of constructing Asp-KGs.

Algorithm 1. Commonsense Graph Construction

Input: aspect categories $A = \{a_1, \dots, a_N\}$, sentences $S = \{w_1, \dots, w_M\}$, and common knowledge graph $\mathcal{G}_c$

- 1: Take aspect category a_i as central node;
- 2: **for** $i \leftarrow 1$ **to** N **do**
- 3: **for** $j \leftarrow 1$ **to** M **do**
- 4: Calculate relational score $r_{i,j}$ by LLMs
- 5: Sort $r_{i,j}$ in descending order and select Top P nodes as first-order neighbors $\{n_1, n_2, \dots, n_P\}$
- 6: **end for**
- 7: **for** $m \leftarrow 1$ **to** P **do**
- 8: Search neighbor nodes of n_m in graph $\mathcal{G}_c$
- 9: Count number of occurrences of above nodes
- 10: Select nodes, i.e., $\{n_n^1, n_n^2, \dots, n_n^Q\}$, with the number index $> 3 \& < 10$ as neighbor nodes for n_m (i.e., nodes for central node a_i)
- 11: **end for**
- 12: Gather first-order nodes and second-order nodes as the Asp-KG $\mathcal{G}_i$ for aspect category a_i
- 13: **end for**

Output: aspect-oriented KGs $\{\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_N\}$.

4.1.1 First-order Internal Aspect-aware Graph. It is observed that the aspect category of each data set has a strong correlation with some specific description words. Therefore, we aim to use LLMs to assist in the knowledge generation process. However, existing research indicates that directly using large models to generate knowledge often leads to data leakage issues, significantly impacting the fairness performance of subsequent models [2, 42].

To address this problem, we innovatively use LLMs as evaluator instead of generator in the knowledge generation process. Specifically, we first employ prompt learning to design prompts that guide the large model, turning it into an estimator. The prompt design is “You are a well-trained sentiment classifier with deep commonsense knowledge. Please evaluate the sentiment relevance scores of words w_j and aspect category a_i . The scores range from 0 to 10, where 0 means irrelevant and 10 means extremely relevant.”.

Then, the LLM-guided evaluators calculate the relational score² that provides the importance of word w_j for aspect category a_i . The process is briefly defined as follows:

$$r_{i,j} = \text{score}(a_i, w_j) = \text{LLM}(a_i, w_j). \quad (1)$$

Next, we sort $r_{i,j}$ in descending order and extract top P most relevant words (i.e., $\{n_1, n_2, \dots, n_P\}$) as the first-order neighbor nodes of i -th aspect category a_i . Since those nodes are extracted from the data corpus, they are the most relevant internal semantics for the specific aspect category.

4.1.2 Second-order External Commonsense Graph. The external commonsense information mainly comes from the common knowledge graph ($\mathcal{G}_c$). We retrieve the second-order nodes by using the retrieved first-order nodes as the root entity, as described in

²We tested dozens of prompts and found that different prompts had little impact on the final relational results.

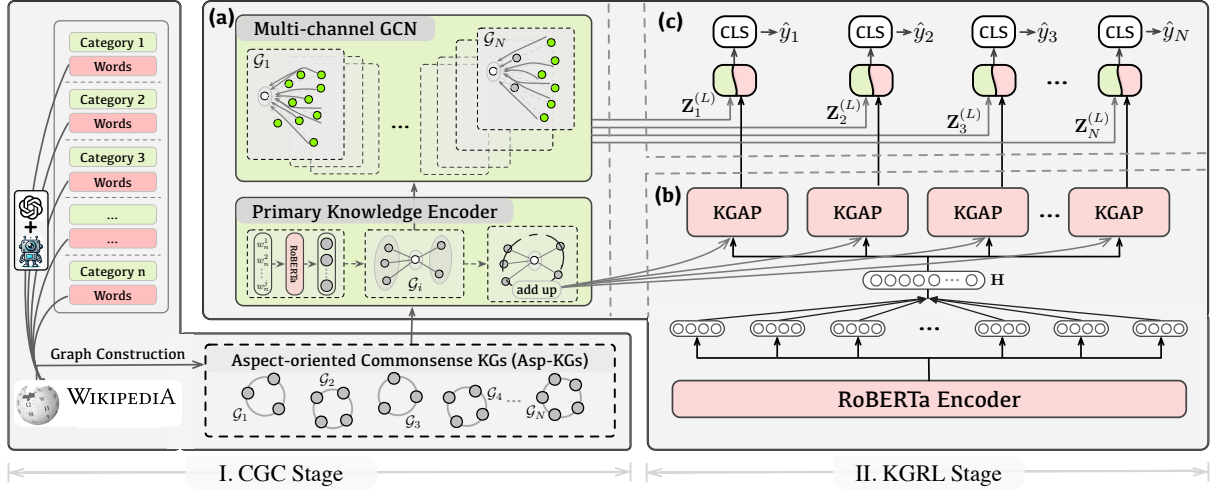


Figure 2: The LLM-Kit framework. Generally, it can be divided into two stages: I. the CGC stage and II. the KRGL stage.

Algorithm 1, steps 7~11. In this phase, we take the above nodes, i.e., $\{n_1, n_2, \dots, n_p\}$ generated by LLMs, as root entity and filter out related adjacent entities based on frequency. To achieve this, we gave an index for each node to count the number of their occurrences. Besides, to better control the scale of Asp-KGs and reduce the introduction of too much irrelevant knowledge, we select the appropriate second-order neighbors $\{n_n^1, n_n^2, \dots, n_n^Q\}$ for each node by limiting the size of the index. Note that, the value of the index size is a hyperparameter, we can manually change it to control the size of the map. Here, we chose 3 and 10 to get the best-performing commonsense KG.

Through the Algorithm 1, we obtain internal words as the first-order neighbor nodes and external entities as the second-order neighbor nodes. Finally, we gather them together to form Asp-KGs, i.e., $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_N$. There are at least two advantages with such an aspect-oriented knowledge graph. First, each aspect category has its own Asp-KG and thus can better learn specific representations from the entire semantic space. Second, if a review text contains more than one aspect category, the Asp-KG will aggregate the nodes at the particular aspect, which provides commonsense knowledge.

4.2 KGRL stage: Commonsense Learning

KGRL is made up of three main parts, as shown in Figure2: a graph representation learning module (GRL), a sentence semantic learning module (SSL), and a knowledge fusion and sentiment prediction module. In the following, we describe each of them in detail.

4.3 Graph Representation Learning (GRL)

Primary Knowledge Encoder. To employ the constructed Asp-KGs (i.e., $\mathcal{G}_N$), we first need to generate an initial embedding of aspect category's neighbor nodes (words). Here, we train a primary knowledge encoder to learn external semantics on aspect categories. Given a central node (aspect category) $\{w_a^1, \dots, w_a^I\}$ and its neighbor nodes $\{w_n^1, \dots, w_n^J\}$ in Asp-KG, the primary knowledge encoder

takes them as input and encodes each node via a pre-trained language model, and then splices them together to form the primary representation of the aspect category.

In this paper, we adopt RoBERTa [24] as the upstream feature encoder and freeze most of its parameters to get each node's initial semantic embedding. Formally, the calculation is briefly described as follows:

$$\{z_a^1, \dots, z_a^I\} = \text{RoBERTa}\{w_a^1, \dots, w_a^I\}, i \in [1, I] \quad (2)$$

$$\{z_n^1, \dots, z_n^J\} = \text{RoBERTa}\{w_n^1, \dots, w_n^J\}, j \in [1, J] \quad (3)$$

where I is the number of words in aspect category and J is the number of aspect category's neighbor words. Note that, aspect-category may contains multiple words, here, we take them as one central node by adding up their embeddings:

$$Z_a = [(z_a^1 \oplus z_a^2 \oplus \dots \oplus z_a^I) \oplus z_n^1 \oplus z_n^2 \oplus \dots \oplus z_n^J], \quad (4)$$

where $\oplus$ denotes the addition of word embeddings. Z_a is the primary representation of aspect category. Through the above calculations, we generate the primary semantic representation of each aspect category (i.e., $Z_a^1, Z_a^2, \dots, Z_a^N$, N is the number of aspect categories) from its original embedding while simultaneously preserving the full semantic information of the neighbor nodes.

Multi-channel GCN (MGCN). Under the graphical structure, different Asp-KGs may convey varying degrees of semantic information about the various aspect categories, thus, we developed a Multi-channel GCN to capture these distinct semantic features. Generally, the MGCN is a slight variant of the GCN [14] architecture, which is able to integrate node features and graphical structures in a complex knowledge graph with rich information.

Specifically, we have N aspect categories in the dataset, and we generate N Asp-KGs (i.e., $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_N$) through Algorithm 1. As the process of learning reviews' semantics is relatively consistent across all aspect-categories, we mainly introduce one aspect category's semantic aggregation for space saving. Formally, given an Asp-KG $\mathcal{G} = (A, X)$ for a specific aspect category, where $A \in \mathbb{R}^{J \times J}$ is adjacency matrix and $X \in \mathbb{R}^{J \times d}$ is the node feature matrix, we can utilize the following GCN layers to iterate l times (thus propagate

neighbors' semantics in l hops) to generates the output encoding for central node representation as $Z^{(l)} \in \mathbb{R}^{J \times d}$. Note that, $A_{mn} = 1$ represents there is an edge between nodes m and n ($m, n \in [1, J]$), otherwise, $A_{mn} = 0$. We define the aggregation process as:

$$\begin{aligned} Z_i^{(l)} &= GCN(A_i, X_i), \\ &= \tanh \left(\tilde{D}_i^{-\frac{1}{2}} \tilde{A}_i \tilde{D}_i^{-\frac{1}{2}} Z_i^{(l-1)} W_i^{(l)} \right), \end{aligned} \quad (5)$$

where $W_i^{(l)}$ is learnable weight matrix of l -th layer in GCN and A_i is the i -th Asp-KG corresponding to the i -th aspect category. $\tanh$ is the activation function and the initial value of $Z_i^{(l)}$ is $Z_i^{(0)} = X$. Besides, we have $\tilde{A}_i = A_i + I$ and $\tilde{D}_i$ is the diagonal degree matrix of $\tilde{A}_i$. We denote the last layer output embedding as $Z_i^{(L)}$. As a result, we can obtain the embedding of each central node from the corresponding Asp-KG with a single GCN, capturing the desired semantics in a unified feature space.

In the above, we have described how each aspect category feature is extracted from a single GCN (i.e., one channel to aggregate all nodes in each Asp-KG). In reality, however, the feature spaces of different Asp-KGs are not entirely relevant. For example, the feature distribution of Asp-KG directed to "Service" differs significantly from that aimed at "Food". Using the same GCN to learn these features may hurt the performance. To obtain independent external knowledge of each aspect category, we augment the classic GCN into a Multi-channel GCN, which uses different channels for the aggregation of each Asp-KG. Specifically, the input of MGCN is every Asp-KG, and the output is the feature of each aspect category. The operation incorporates the individual influence among a slate of neighbor nodes via a given channel and thus characterizes the importance of each node in Asp-KG:

$$\begin{aligned} (Z_1^{(L)}, \dots, Z_N^{(L)}) &= MGCN(A_1, \dots, A_N, X_1, \dots, X_N), \\ &= \begin{cases} GCN(A_1, X_1), \\ \dots \dots \dots \\ GCN(A_N, X_N), \end{cases} \end{aligned} \quad (6)$$

Here, N is the channel number, which is in line with the number of aspect categories. For each GCN channel, the process described by Equation 5 is followed. As depicted in Figure 2 (a), the MGCN assigns an independent GCN submodule to each Asp-KG to ensure that the feature representation learned from external KGs is more inclined to a particular aspect category. So far, we can retrieve each category's aspect-aware features from the external knowledge graph. In what follows, we introduce how to encode internal semantics from the sequential review sentences.

4.4 Sentence Semantic Learning (SSL)

RoBERTa Encoder. As shown in Figure 2 (b), we use last hidden state of the pre-trained RoBERTa to present words in the review text. Specifically, let $H \in \mathbb{R}^{d \times L}$ be the matrix consisting of token embedding vectors that produces:

$$H = \{h_1, \dots, h_L\} = RoBERTa\{w_1, \dots, w_L\}, \quad (7)$$

where d is the size of hidden layers and L is the length of review text. We discard the <CLS> token embedding, thus the number of output features are consistent with the input words. Note that,

Table 1: Statistics of datasets used in this paper.

Dataset	#Pos.	#Neg.	#Neu.	Ave. # Length	Ave. # Aspects
Restaurant	1,982	881	138	12.6	1.5
Laptop	1,614	901	149	13.2	1.3
ASAP	133, 721	27, 425	52, 225	319	5.8

Table 2: Statistics of the constructed knowledge graphs.

KGs	#Asp-KGs	#Nodes	#Relations	Ave. # Nodes
Res.-KGs	12	621	1127	51.8
Laptop-KGs	10	380	789	38.0
ASAP-KGs	18	527	509	29.3

all parameters were shared between the two RoBERTa encoders shown in Equations 2, 3, and 7.

Knowledge-guided Attention Pooling (KGAP). Since the semantics of the aspect categorie is scattered throughout the overall review, such ambiguous information could be detrimental if the learned feature is ill-fitted. Therefore, we develop a knowledge-guided attention-pooling layer to dynamically aggregate the related token embeddings under the direction of the prior knowledge representations (i.e., Z_a^i) for each aspect category. Formally, the calculation process is:

$$p_i = \tanh(W_i * H), \quad (8)$$

$$\alpha_i = \text{softmax}(\gamma * Z_a^i + (1 - \gamma) * \omega_i) * p_i, \quad (9)$$

$$R_i = \tanh(W_i^p * H * \alpha_i^T), \quad (10)$$

where $p_i \in \mathbb{R}^{d \times L}$, $W_i \in \mathbb{R}^{d \times d}$ and $i \in [1, N]$. $H \in \mathbb{R}^{d \times L}$ is the embedding matrix of review text. $Z_a^i \in \mathbb{R}^d$ is the embedding of each aspect category which obtains from equation 4. $\omega_i \in \mathbb{R}^d$ is a learnable vector. γ is a factor to balance the influence of external aspect-aware knowledge. $\alpha_i \in \mathbb{R}^L$ is a vector that includes attention weights of all tokens that can selectively attend the regions of the aspect category related tokens. $R_i \in \mathbb{R}^d$ is the attentive representation of review text with respect to the i -th aspect category.

4.5 Knowledge Fusion and Sentiment Prediction

In the following, we explain a new knowledge fusion operation that aims to balance the knowledge between external Asp-KGs and internal semantics efficiently. To be specific, before applying the learned features into sentiment prediction layer, the external representation $Z_i^{(L)}$ and internal representation R_i is first linearly projected to the same subspace:

$$F_i = Z_i^{(L)} * W_i^f * R_i, \quad (11)$$

$$\hat{y}_i = \text{softmax}(W_i^g * F_i + b_i^g). \quad (12)$$

where F_i is the final fusion representation, $W_i^f \in \mathbb{R}^{d \times d}$ is the weight matrix that maps external and internal features to the same latent semantic space. Later, we use a traditional loss function, which is defined as:

$$\mathcal{L}(\hat{y}, y) = - \sum_{j=1}^C y_j^i \log(\hat{y}_i^j), \quad (13)$$

where y is the ground truth and $\hat{y}$ is the predicted probability of y . C is the number of labels (i.e., 3 in our task).

5 Experimental Setups

5.1 Datasets

Restaurant & Laptop Dataset: The series of SemEval datasets consists of user reviews from Amazon e-commerce websites widely used and pushed forward related research. This paper uses the SemEval-2016 task-5 dataset, and the statistics of processed datasets are shown in Table 1. Note that the common KGs used in those two datasets is Freebase [1]. **ASAP Dataset:** The public review dataset ASAP [3] is collected from a popular O2O e-commerce platform in China, which allows users to publish fine-grained reviews to restaurants they have visited. As Table 1 shows, ASAP dataset contains more words and aspect categories in each review, so it is more challenging to predict the sentiment behind those long review texts. Besides, the statistics of the constructed KGs are shown in Table 2.

5.2 Details of Baseline Methods

To thoroughly assess the performance of our model, we have selected key baseline methods from both the ACSA and ABSA tasks. In the following, we provide a detailed introduction to each of these methods.

5.2.1 The ACSA Models.

- **AAL-SS** [52]: a novel model for aspect category sentiment classification which contains an aspect lexicon based and an aspect supervision based algorithm.
- **Vanilla BERT** [4]: is a powerful pre-trained language model which takes aspect and sentence as joint input.
- **CapsNet-BERT** [11]: It is a capsule network to model the complicated relationship between aspects and contexts, which combines the strengths of the pre-trained model.
- **Joint Model** [3]: It's a joint learning model to address ACSA and Rating Prediction in a multitask learning manner.
- **AFML** [21]: It's a framework that constructs aspect-aware and aspect-contrastive representations from external knowledge to match the target aspect with aspects in the training set

5.2.2 The ABSA Models.

- **ASTE** [29]: introduces a sentiment triplet extraction task that answers what is the aspect, how is its sentiment and why is the sentiment in one shot by coupling together aspect extraction aspect term sentiment classification and opinion term extraction in a two-stage framework..
- **KGAN** [50]: proposes a knowledge graph augmented network (KGAN), which aims to effectively incorporate external knowledge with explicitly syntactic and contextual information.
- **DSSK-GAN** [23]: introduces a novel DSSK-GAN model, which utilizes both dynamic and static external knowledge to aid in modeling the relationship between aspect words and their associated sentiment words.

5.3 Motivation for Baseline Selection

It is common knowledge that there are numerous methodologies in ABSA domain; however, given the particulars of the ACSA task and how it differs from ABSA, we are unable to compare all of the methodologies. We explain the reasons for the selection of the baselines and why some ABSA models are not applicable to ACSA task in this section.

5.3.1 DNN-based and Pre-trained Models. As described in Section 2, each of these models uses different model architectures (e.g., LSTM, Attention and Transformer) to semantically encode the entire sentence. Therefore, these approaches are applicable regardless of whether the target of sentiment analysis is aspect term or aspect category. To this end, we select the optimal models (e.g., [18] and [3]) in both ABSA and ACSA tasks as the baseline methods for experimental comparison. It is worth noting that LLM-Kit is distinct from these methods because it simultaneously probes both external commonsense knowledge and internal semantics. Specifically, unlike pre-trained models [12, 41, 44], LLM-Kit has a remarkable ability to encode commonsense knowledge through Asp-KGs, which are purpose-built for aspect categories, in addition to using PLMs to encode the whole sentence.

5.3.2 GNN-based Models. As we described in Related Work, most GNN-based methods [28, 32, 39, 45] performed aspect based sentiment analysis (ABSA) by constructing syntactic graph for the aspect. However, those methods are limited to exploiting aspect categories' sentiment since generating syntactic graph for the aspect category is impossible (i.e., aspect category do not appear in the review sentence). For this reason, it is difficult to use ABSA's graph-based models on ACSA task.

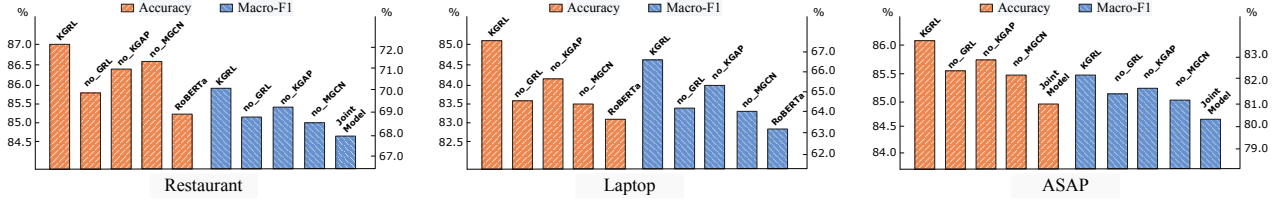
Meantime, it is worth noting that LLM-Kit differs from the previous graph-based methods in that we compose the commonsense knowledge graph construction process concerning aspect categories. As a result, LLM-Kit is able to explicitly encode the mutual influences from external knowledge graphs and handle aspect-aware semantics. More importantly, unlike existing data augmentation methods that rely on large models, LLM-Kit innovatively employs large models as evaluators. This strategy significantly mitigates the risk of data leakage associated with LLMs.

5.4 Training Setup

Most of the baselines are implemented in Pytorch based on the public code, following their parameter settings (learning rate, batch size, etc). For our LLM-Kit model, we use Adam [13] as the optimizer, with 0.9 of β_1 , and 0.999 of β_2 . For parameters in RoBERTa, the learning rate is initially set as $2e-5$. Besides, the maximum length of the review text is 512. The dimension of the word embedding is 768 and the training batch size is 16. To avoid overfitting and stabilize the training, we set the dropout rate as 0.1. The value of P is set as 30, 25 and 20 for restaurant, laptop and ASAP, respectively. The key parameter settings for baselines are set as default. We use two metrics (i.e., Accuracy and Macro-F1) for the model evaluation, which are widely used in ACSA. We conduct experiments on each baseline with parameters in the corresponding papers, and the results are generally consistent with those in the paper. All of the

Table 3: Performance on three public datasets. The underlined number in each column is the best result among the baselines.

Baseline Methods	Restaurant.		Laptop.		ASAP.	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
Aspect-category Baselines						
(1) AAL-SS	84.35%	67.14%	78.29%	60.00%	82.64%	77.86%
(2) Vanilla BERT	84.52%	67.19%	82.69%	62.86%	84.09%	79.18%
(3) CapsNet-BERT	83.37%	67.38%	83.10%	63.04%	83.74%	78.92%
(4) Joint Model	85.33%	68.02%	83.02%	63.10%	85.15%	80.75%
(5) AFML	<u>86.28%</u>	<u>68.13%</u>	<u>83.71%</u>	<u>64.20%</u>	84.77%	79.13%
Aspect-based Baselines						
(1) ASTE [29]	82.22%	63.23%	79.94%	59.03%	80.04%	75.40%
(2) KGAN [50]	83.32%	64.12%	81.30%	60.67%	82.78%	77.10%
(3) DSSK-GAN [23]	85.17%	67.88%	82.94%	62.99%	84.77%	78.75%
KGRL + Freebase	86.94%	69.69%	85.17%	66.84%	86.11%	81.94%
KGRL + AspKG	87.38%	70.33%	85.89%	66.88%	86.76%	82.38%
LLM-Kit (AspKG)	88.04%	70.16%	86.27%	67.30%	87.49%	82.72%


Figure 3: Ablation results from variants of KGRL. The left Y-axis denotes the value of Accuracy and the right Y-axis indicates the value of Macro-F1. no_GRL is a variant of KGRL that removes the GRL module from KGRL directly.

models are trained on a Linux server with 64 Intel(R) CPUs and 4 Tesla V100 32GB GPUs.

6 Experimental Results

6.1 Overall Results and Discussion

The main results, presented in Table 3, show that our proposed LLM-Kit outperforms all baselines across various metrics, including ACSA and ABSA methods. Furthermore, several notable phenomena emerge from these results:

First, it is clearly apparent that our model outperforms all baselines across all datasets by a significant margin. Specifically, LLM-Kit achieves remarkable relative improvement over the strongest aspect-category baseline w.r.t. accuracy and macro-F1, respectively. It proves the superiority of LLM-Kit in the ACSA, as previous models did not fully utilize commonsense information. Second, LLM-Kit outperforms the best ABSA model (i.e., DSSK-GAN) in both of the accuracy and macro-F1 score. The reason for this phenomenon is that previous ABSA models typically focus on the syntactic structure within the sentence, overlooking the commonsense information that is crucial for ACSA. Finally, we also evaluated the two core stages of LLM-Kit. As shown in the table, both the Asp-KG we constructed and the downstream KGRL module achieved excellent performance.

The above observations make it clear that the performance of the baselines is limited without the guidance of commonsense

knowledge. In contrast, our LLM-Kit model enhances the interaction between internal semantics and external knowledge and thus outperforms all the baseline methods.

6.2 Ablation Analysis

Without Graph Represent Learning (no_GRL). To evaluate the efficacy of the graph represent learning (GRL) module, we remove it from KGRL directly. As shown in Figure 3, three datasets perform worse without the GRL. Thus, the graph representation learning effectively captures the rich mutual influences between aspect category and neighbors, confirming the necessity of emphasizing external knowledge from Asp-KGs. **Without Multi-channel GCN (no_MGCN).** In no_MGCN module, we swap out the MGCN for a regular GCN; thus, the model learns graph embeddings from Asp-KGs via a parameter-shared GCN. In Figure 3, we see that KGRL’s performance drastically decreases when MGCN is not used to acquire specific knowledge for each Asp-KG. **Without Knowledge-guided Attention Pooling (no_K-GAP).** We also study the influence of KGAP module. We remove KGAP so that the words are aggregated without knowledge supervision. As shown in Figure 3, the performance of no_KGAP becomes worse compared with KGRL. This is possible because commonsense knowledge supervision is vital for aggregating the aspect categories with related words. With the above experiments, we find that each sub-module in KGRL is effective in learning aspect-aware semantics and is indispensable for LLM-Kit to achieve excellent performance.

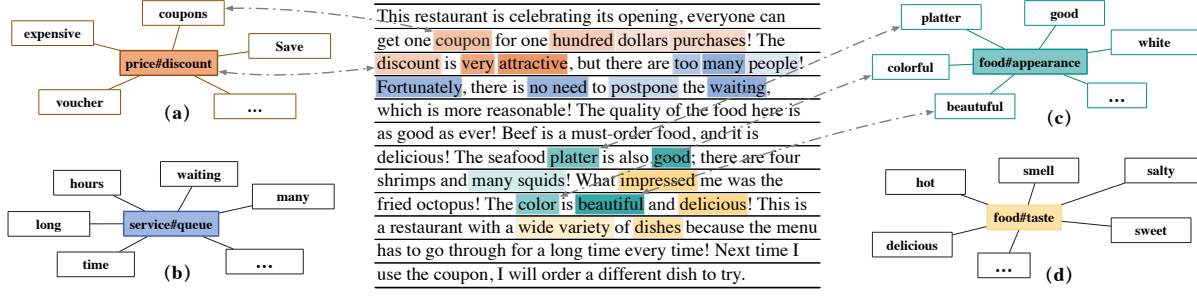


Figure 4: The visualization of first-order nodes from Asp-KGs with respect to four aspect categories (i.e., the central nodes in (a), (b), (c) and (d)). Besides, we also visualize the attention scores of sentence. The color intensity reflects the attention’s magnitude.

Model	Restaurant		Laptop		ASAP	
	Accuracy (%)	Macro-F1 (%)	Accuracy (%)	Macro-F1 (%)	Accuracy (%)	Macro-F1 (%)
LLaMA	73.48	45.34	68.32	41.03	51.58	39.56
Qwen	81.69	53.82	71.63	46.73	54.73	30.22
Baichuan	42.53	36.54	32.81	24.98	23.34	16.52
Deepseek	76.60	43.80	71.60	39.88	57.78	32.91
Mistral	77.75	65.76	74.53	66.01	61.18	40.59

Table 4: The comparative analysis with other LLM methods.

6.3 Neighbor Nodes from Asp-KGs

In order to intuitively show the knowledge extracted by LLM-Kit, we show the top-5 first-order nodes on the ASAP dataset with the highest relational attention scores ($r_{i,j}$ in Algorithm 1 and Equation 1) to each aspect category in Figure 4 (a)~(d), respectively. We take the aspect category “price#discount” as the example for illustrating the selected neighbor words. Figure 4 (a) demonstrates that almost all neighbor nodes are highly correlated with aspect category “price#discount”. For example, the neighbor nodes “voucher”, “coupons” and “save” contain rich commonsense knowledge that can directly express a positive tendency towards discounts. Consequently, associating these words with the aspect category will improve the semantic information of “price#discount”. Similar phenomena in other aspect categories (i.e., Figure 4 (b), (c), and (d)) show that the Asp-KG construction algorithm performs well on an aspect category and can extract the most relevant neighbor information for each aspect category.

6.4 Effect Verification of Asp-KGs

To show the effectiveness of Asp-KGs, we randomly sampled a review text and extracted word attention scores from the vector α_i as calculated in Equation 9. As shown in Figure 4, the review contains four aspect categories (e.g., “price#taste”). The review contains four colors, each representing a word related to a specific aspect category. For instance, red highlights words the model focuses on when predicting the sentiment label for the aspect category “price#discount”. The following observations can be made from the visualization results:

First, we can observe that the model pays attention to different words in a small scope of review text when predicting the sentiment

of varying aspect categories. The reason may be that the aspect category’s representation is calculated from the Asp-KG, thus including the overall semantics. In contrast, the attention scores are calculated precisely from the sentence tokens with the guidance of external knowledge. With these attention scores, the model may select context words most related to the aspect category. On the other hand, our proposed model assigns more significant weight to the words that co-occurred in aspect category KGs. It makes sense because the external KGs help integrate commonsense information into aspect category representation, thus learning better aspect-aware semantics for ACSA.

6.5 Comparative analysis of LLMs

We list some core results for LLMs in Table 4. In LLM-Kit, RoBERTa was chosen over models like GPT-4 [27] or LLaMA [37] due to its strong performance in sentence-level sentiment and semantic encoding tasks, making it a good fit for our Sentence Semantic Learning (SSL) module. Additionally, RoBERTa integrates effectively with our multi-channel graph convolutional network (GCN) to capture aspect-specific sentiment features.

While models like GPT-4 and LLaMA excel in generation tasks, they are not yet optimized for fine-grained aspect-based sentiment analysis and often incur higher computational costs. Hence, RoBERTa serves as a practical, effective choice for sentence encoding in Aspect Category Sentiment Analysis (ACSA). We plan to further explore comparisons with other LLMs in future work.

6.6 Error Analysis of LLMs

In order to study the reasons why LLMs generate some errors in ACSA, we carefully select a typical sentence from the laptop dataset as shown in Table 5 **Sentence One**. Specifically, LLMs inevitably misrecognize the sentiment involved in some aspect categories due to the complexity of sentence semantics. For example, both satisfaction and dissatisfaction with the computer involved in the given sentence make it difficult for LLMs to understand the correct sentiment. In addition, without the integration of multiple domain knowledge like our LLM-Kit, LLMs cannot generate the corresponding aspect category according to commonsense information.

6.7 More Experiments

To further explore the shortcomings of LLMs in the ACSA task, we conducted some related experiments. Sentiment element definitions

Error Analysis	LLMs	LLM-Kit
Sentence One: I will give advise any prospective buyers this is not great laptop, It's good for web browsing and word processing. Aspect Categories: Laptop#General (Neu); Laptop#Miscellaneous (Pos)	Neg/None (XX)	Neu/Pos(✓✓)
Sentence Two: People are frustrating to work with, the product itself is very cheaply made, and the accessories are less than satisfactory. Aspect Categories: Laptop#Features (Neg); Laptop#Quality (Neg); Support#Quality (Neg)	None/Neg/None	Neg/Neg/Neg
Sentence Three: The system constantly overheats, the battery life is too short,the case is coming apart, and my core applications that I use every day in my work as a graphic artist run poorly. Aspect Categories: LAPTOP#OPERATION_PERFORMANCE (Neg); LAPTOP#QUALITY (Neg); BATTERY#OPERATION_PERFORMANCE (Neg)	Neg/None/Neg	Neg/Neg/Neg

Table 5: Case Study of our LLM-Kit compared with LLMs.The colored words in brackets represent the ground truth sentiment polarity of the corresponding aspects. We denote positive, neutral and negative sentiment as Pos, Neu and Neg, respectively. Additionally, None means that LLMs can not detect the sentiment of the corresponding aspect category.

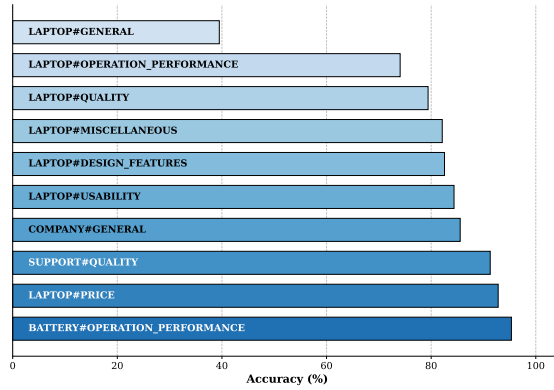


Figure 5: Accuracy of different aspect Categories conducted by GPT-4.

include the permitted label space, e.g. allowed sentiment polarities and aspect categories. The output format describes the expected structure of model responses, allowing us to decode the responses into our desired format.

As mentioned in the previous section, we carefully selected some typical examples for the error analysis of LLMs in ACSA. Due to the page limit of the main body, we present more examples in Table 5. Specifically, LLMs cannot analyze some specific aspect categories from the given review, which does not contain directly related aspect terms. For example, without the integration of multiple domain knowledge like our LLM-Kit, LLMs are unable to generate the aspect categories “Laptop#Design_Features” and “Support#Quality” from the “cheaply made” and “accessories”. In addition, LLMs also lack the ability to push further into deeper aspect categories from the ones they can already recognize. As the second example shows, the aspect categories “Laptop#Operation_Performance” and “Battery#Operation_Performance” naturally introduce the aspect category “Laptop#Quality”.

6.8 Case Study

In addition, we also conducted experiments to study the performance of LLMs in different aspect categories. As shown in Figure 5, LLMs exhibit an imbalance in different aspect categories for ACSA. Taking the Laptop dataset as an example, LLMs achieve comparable performance in some aspect categories with our LLM-Kit. However, the general accuracy is only 55.35% which is much less than our LLM-Kit on Laptop dataset. It is well known that a single review tends to contain multiple aspect categories. LLMs can perform well in sentences including a single aspect category, while still have a long way to go in processing the sentence with mixed aspect categories. In addition, LLMs objectively lack the ability to detect the implied aspect category, which is one of the significant reasons for the low accuracies.

7 Conclusion

In this paper, we proposed a LLM-Kit framework, which enhanced knowledge integration through two stages: commonsense graph construction (CGC) and knowledge-integration graph representation learning (KGRL). In the CGC stage, a two-order extraction strategy with an LLM-based scoring evaluator transformed general knowledge graphs into aspect-oriented ones (Asp-KGs), capturing both commonsense and domain-specific knowledge. In the KGRL stage, a graph representation learning module and a sentence semantic learning module (using RoBERTa) were designed to integrate aspect categories, opinion terms, and sentence structures. A knowledge fusion function then balanced their contributions. Experiments on three public and real-world datasets demonstrated our framework’s effectiveness.

Acknowledgments

This research was partially supported by the China National Natural Science Foundation (2406303), the Anhui Provincial Natural Science Foundation (No. 2308085QF229), the Anhui Province Science and Technology Innovation Project (202423k09020010), the Fundamental Research Funds for Central Universities (No. WK2150110034).

GenAI Usage Disclosure

We express our gratitude to the large language models that assisted in the polishing of this paper. Their contributions have been instrumental in refining the text, ensuring clarity and precision in the presentation of our research.

References

- [1] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. 1247–1250.
- [2] Jaydeep Borkar. 2023. What can we learn from Data Leakage and Unlearning for Law. *arXiv preprint arXiv:2307.10476* (2023).
- [3] Jiahao Bu, Lei Ren, Shuang Zheng, Yang Yang, Jingang Wang, Fuzheng Zhang, and Wei Wu. 2021. ASAP: A Chinese Review Dataset Towards Aspect Category Sentiment Analysis and Rating Prediction. *arXiv preprint arXiv:2103.06605* (2021).
- [4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
- [5] Qin Ding, Luo Zhao, and Wenhan Xia Junjie Hu Anh Tuan Luu Shafiq Joty Li, Guizhen Chen. 2024. Data Augmentation using Large Language Models: Data Perspectives, Learning Paradigms and Challenges. *arXiv preprint arXiv:2403.02990* (2024).
- [6] Xiaowen Ding and Bing Liu. 2007. The utility of linguistic rules in opinion mining. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 811–812.
- [7] Li Dong, Furu Wei, Chuankai Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 2: Short papers)*. 49–54.
- [8] 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*. 3433–3442.
- [9] Tiquan Gu, Hui Zhao, Zhenzhen He, Min Li, and Di Ying. 2023. Integrating external knowledge into aspect-based sentiment analysis using graph neural network. *Knowledge-Based Systems* 259 (2023), 110025.
- [10] Binxuan Huang and Kathleen M Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. *arXiv preprint arXiv:1909.02606* (2019).
- [11] 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)*. 6280–6285.
- [12] Zixuan Ke, Hu Xu, and Bing Liu. 2021. Adapting BERT for Continual Learning of a Sequence of Aspect Sentiment Classification Tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 4746–4755.
- [13] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [14] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [15] Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. Nrcanada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*. 437–442.
- [16] Chen Li, Jiang Ji, and Jianxin Li. 2024. LLM-based Multi-Level Knowledge Generation for Few-shot Knowledge Graph Completion. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence Main Track*. 2135–2143.
- [17] Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. *arXiv preprint arXiv:1805.01086* (2018).
- [18] Yuncong Li, Cunxiang Yin, Sheng-hua Zhong, and Xu Pan. 2020. Multi-instance multi-label learning networks for aspect-category sentiment analysis. *arXiv preprint arXiv:2010.02656* (2020).
- [19] Zhong Li, Liu Ke Zhu, Dacheng Bo Du, and Tao. 2024. Iterative Data Augmentation with Large Language Models for Aspect-based Sentiment Analysis. *arXiv preprint arXiv:2407.00341* (2024).
- [20] Zhengyan Li, Yicheng Zou, Chong Zhang, Qi Zhang, and Zhongyu Wei. 2021. 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*. 246–256.
- [21] Bin Liang, Xiang Li, Lin Gui, Yonghao Fu, Yulan He, Min Yang, and Ruifeng Xu. 2023. Few-shot aspect category sentiment analysis via meta-learning. *ACM Transactions on Information Systems* 41, 1 (2023), 1–31.
- [22] Shaopeng Tang Lin Li and Renwei Wu. 2024. Majority Rules Guided Aspect-Category Based Sentiment Analysis via Label Prior Knowledge. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. 10952–10957.
- [23] Hongtao Liu, Xin Li, Wanying Lu, Kefei Cheng, and Xueyan Liu. 2024. Graph Augmentation Networks Based on Dynamic Sentiment Knowledge and Static External Knowledge Graphs for aspect-based sentiment analysis. *Expert Systems with Applications* 251 (2024), 123981.
- [24] 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. *arXiv preprint arXiv:1907.11692* (2019).
- [25] Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893* (2017).
- [26] Sajad Movahedi, Erfan Ghadery, Hesham Faily, and Azadeh Shakery. 2019. Aspect category detection via topic-attention network. *arXiv preprint arXiv:1901.01183* (2019).
- [27] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, and Barret Zoph. 2024. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL] <https://arxiv.org/abs/2303.08774>
- [28] Shiguan Pang, Yun Xue, Zehao Yan, Weihao Huang, and Jinhui Feng. 2021. Dynamic and multi-channel graph convolutional networks for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 2627–2636.
- [29] Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 8600–8607.
- [30] Ming hu Shi, Shi Ren, and Satoshi Nakagawa. 2024. Aspect based sentiment analysis with instruction tuning and external knowledge enhanced dependency graph. *Applied Intelligence* (2024).
- [31] Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314* (2019).
- [32] 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)*. 5679–5688.
- [33] Xinjie Sun, Kai Zhang, Qi Liu, Meikai Bao, and Yanjiang Chen. 2024. Harnessing domain insights: A prompt knowledge tuning method for aspect-based sentiment analysis. *Knowledge-Based Systems* 298 (2024), 111975.
- [34] Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2015. Effective LSTMs for target-dependent sentiment classification. *arXiv preprint arXiv:1512.01100* (2015).
- [35] Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect level sentiment classification with deep memory network. *arXiv preprint arXiv:1605.08900* (2016).
- [36] Hao Tian, Can Gao, Xinyan Xiao, Hao Liu, Bolei He, Hua Wu, Haifeng Wang, and Feng Wu. 2020. SKEP: Sentiment Knowledge Enhanced Pre-training for Sentiment Analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 4067–4076.
- [37] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971 [cs.CL] <https://arxiv.org/abs/2302.13971>
- [38] Huang Wan, Quan Cai, and Shi Wei Bi. 2024. Knowledge Fusion of Large Language Models. *arXiv preprint arXiv:2401.10491* (2024).
- [39] Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. *arXiv preprint arXiv:2004.12362* (2020).
- [40] 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*. 606–615.
- [41] Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. *arXiv preprint arXiv:1904.02232* (2019).
- [42] Kun Li Yan, Dong Xu, Ren Zhang, and Cheng. 2024. On Protecting the Data Privacy of Large Language Models (LLMs): A Survey. *arXiv preprint arXiv:2403.05156* (2024).
- [43] Nuo Xu Ye, Zhou Wang, Tao Gui Zhang, and Huang. 2024. LLM-DA: Data Augmentation via Large Language Models for Few-Shot Named Entity Recognition. *arXiv preprint arXiv:2402.14568* (2024).
- [44] Biqing Zeng, Heng Yang, Ruyang Xu, Wu Zhou, and Xuli Han. 2019. Lcf: A local context focus mechanism for aspect-based sentiment classification. *Applied Sciences* 9, 16 (2019), 3389.
- [45] Chen Zhang, Qiuchi Li, and Dawei Song. 2019. Aspect-based sentiment classification with aspect-specific graph convolutional networks. *arXiv preprint arXiv:1909.03477* (2019).

- [46] Kai Zhang, Qi Liu, Hao Qian, Biao Xiang, Qing Cui, Jun Zhou, and Enhong Chen. 2021. Eatn: An efficient adaptive transfer network for aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering* 35, 1 (2021), 377–389.
- [47] Kai Zhang, Kun Zhang, Mengdi Zhang, Hongke Zhao, Qi Liu, Wei Wu, and Enhong Chen. 2022. Incorporating dynamic semantics into pre-trained language model for aspect-based sentiment analysis. *arXiv preprint arXiv:2203.16369* (2022).
- [48] Shiman Zhao, Yutao Xie, Wei Chen, Tengjiao Wang, Jiahui Yao, and Jiabin Zheng. 2024. Metric-Free Learning Network with Dual Relations Propagation for Few-Shot Aspect Category Sentiment Analysis. *Transactions of the Association for Computational Linguistics* 12 (2024), 100–119.
- [49] Wang Zheng and Xuejie Zhang Yu. 2024. Instruction Tuning with Retrieval-based Examples Ranking for Aspect-based Sentiment Analysis. *arXiv preprint arXiv:2405.18035* (2024).
- [50] Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, Hua Jin, and Dacheng Tao. 2023. Knowledge graph augmented network towards multiview representation learning for aspect-based sentiment analysis. *IEEE Transactions on knowledge and data engineering* 35, 10 (2023), 10098–10111.
- [51] Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2015. Representation learning for aspect category detection in online reviews. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29.
- [52] Peisong Zhu, Zhuang Chen, Haojie Zheng, and Tieyun Qian. 2019. Aspect aware learning for aspect category sentiment analysis. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 13, 6 (2019), 1–21.