JAIST Repository
https://dspace.jaist.ac.jp/

| | |
|---|---|
| Title | 属性に着目したアプローチに基づく属性単位の感情分析の教師なし領域適応 |
| Author(s) | 陸, 兵漢 |
| Citation | |
| Issue Date | 2026-03 |
| Type | Thesis or Dissertation |
| Text version | ETD |
| URL | https://hdl.handle.net/10119/20587 |
| Rights | |
| Description | Supervisor: 白井 清昭, 先端科学技術研究科, 博士 |

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

Doctoral Dissertation

# Aspect-Oriented Approaches to Unsupervised Domain Adaptation in Aspect-Based Sentiment Analysis

Binghan Lu

Supervisor Kiyoaki Shirai

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

March, 2026

**Abstract**

Aspect-Based Sentiment Analysis (ABSA) aims to identify the sentiment polarity associated with specific aspects mentioned in review texts, such as service, food, or price in restaurant reviews. Unlike conventional sentiment classification that focuses on overall opinion at the sentence or document level, ABSA requires fine-grained analysis to capture how evaluative expressions relate to individual aspects. By uncovering aspect-level opinions, ABSA enables more precise understanding of user attitudes, supporting applications such as product improvement, market monitoring, and decision-making based on detailed customer feedback. Despite the success of deep neural architectures and pre-trained language models, ABSA systems remain highly domain-dependent. When an ABSA model trained on a source domain such as restaurant, its performance often degrades on another different domain like laptop, because two different domains usually vary significantly in vocabulary, sentiment expressions, and aspect semantics.

To address this issue, domain adaptation has been extensively studied to enable knowledge transfer across domains. However, conventional approaches typically assume the availability of labeled target-domain data during training, which is often unrealistic in practical scenarios. Furthermore, they overlook the fact that aspect-level variation itself can serve as an additional dimension of domain shift. Different aspects within a single domain, such as service and price, follow distinct linguistic patterns and sentiment distributions. Consequently, achieving robust adaptation requires a framework capable of transferring sentiment–aspect knowledge not only across domains but also across aspects within a domain.

In this dissertation, we investigate two complementary adaptation scenarios designed to overcome these challenges, each reflecting a different level of data availability and generalization difficulty. The first, referred to as Unsupervised Domain Adaptation (Scenario A), assumes that labeled target data are not accessible during training, but unlabeled target data are available. In this scenario, the domain boundary is redefined from the conventional dataset-based perspective to an aspect-oriented one, in which each aspect (such as service, food, or price) is treated as a domain. The key challenge lies in transferring sentiment knowledge from labeled source aspect to unlabeled target aspect, where linguistic expressions and sentiment tendencies vary considerably. The second scenario is called Domain-Agnostic Adaptation (Scenario B), which removes the dependence on target-domain data entirely. Here, the model is trained only on labeled data from multiple

source domains, and generalize to unseen target domains. This formulation represents a more extreme yet realistic condition where no target data, even unlabeled, are available in advance. The focus of this setting is to build a system that can internalize aspect-level knowledge and apply it universally. To tackle these two scenarios, this dissertation proposes two complementary frameworks tailored to each setting.

In Scenario A, where unlabeled target-aspect data are available, each aspect is regarded as a distinct domain, and the goal is to transfer sentiment knowledge from labeled source domain to unlabeled target domain. The proposed framework is divided into two methods for training data construction: Pseudo-Label assignment for unlabeled target domain data and Cross-Aspect Review Generation (CARG). Specifically, a base classifier that fine-tuned on the labeled source domain is trained, and is used to predict the sentiment of unlabeled target domain. Reviews whose maximum class probability exceeds an empirically determined threshold are accepted as reliable pseudo-labeled instances. In the CARG stage, labeled target reviews are automatically generated by replacing sentiment words and keywords in sentences of the source domain. Finally, the labeled data constructed by these two methods are used to train a classifier for determining the polarity of the aspects in the target domain. Experiments on several aspect domains from restaurant and laptop dataset demonstrate improved accuracy and macro-F1 scores of polarity classification. Additional analyses are conducted on threshold selection for pseudo-labeling and sentiment word scoring, and qualitative examples of valid and invalid generations.

In Scenario B, where no target-domain data are available during training, this dissertation proposes an Aspect-Enhanced Prompting (AEP) framework for domain-agnostic adaptation in aspect sentiment classification. The proposed AEP framework is based on two generative language models: one generates a prompt from a given review, while the other follows the prompt and classifies the sentiment of an aspect. The first model extracts Aspect-Related Features (ARFs), which are words closely related to the aspect, from the review and incorporates them into the prompt in a domain-agnostic manner, thereby directing the second model to identify the sentiment accurately. Our framework incorporates an innovative rescoring mechanism and a cluster-based prompt expansion strategy. Both are intended to enhance the robustness of the generation of the prompt and the adaptability of the model to diverse domains. The results of experiments conducted on five datasets (Restaurant, Laptop, Device, Service, and Location) demonstrate that our method outperforms the baselines, including a state-of-the-art unsupervised domain adaptation method. The effectiveness of both the rescoring mechanism and the cluster-based prompt expansion is also validated through an

ablation study.

**KeyWords**: aspect-based sentiment analysis, unsupervised domain adaptation, multi-source domain adaptation, data argumentation, text generation model, prompt engineering

**Acknowledgments**

I would like to express my sincere gratitude to all those who have supported and guided me throughout my doctoral studies at the Japan Advanced Institute of Science and Technology (JAIST).

I would like to express my deepest gratitude to my supervisor, Professor Kiyoaki Shirai, for his invaluable guidance, patience, and constant encouragement throughout my doctoral studies. His insightful advice was essential to the completion of this dissertation.

I am sincerely grateful to my minor research supervisor, Professor Shogo Okada, for his support and suggestions. My heartfelt thanks also go to my second supervisor, Professor Naoya Inoue, and Associate Professor Natthawut Kertkeidkachorn for their constructive comments and assistance in improving the quality of this research.

I would like to thank all the members of the Shirai Laboratory for creating a friendly and collaborative research atmosphere.

Lastly, I would like to thank to my parents and my cousin for their unconditional love and unwavering support throughout my academic journey.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Research Background

In recent years, e-commerce platforms and review sharing websites in domains such as restaurant and consumer electronic have experienced rapid growth, with users and customers actively posting their opinions on a wide range of products and services. The increasing volume of user-generated reviews offers significant potential for understanding customer attitudes, while at the same time presenting methodological challenges for analysis. As a result, considerable research has been directed toward the analysis of opinions expressed in such reviews. Within this research background, sentiment analysis has emerged as a key task, aiming to categorize user opinions into sentiment polarity such as positive, negative, and neutral.

Sentiment analysis, has been extensively investigated in the field of natural language processing (NLP) [34, 46]. Its primary objective is to determine the sentiments or opinions expressed within a given text. Traditional approaches mainly target document-level sentiment classification, where an overall sentiment polarity is assigned to a multi-sentence review [47, 45, 38, 61]. Although such document-level sentiment analysis can provide a general understanding of user attitudes, it often ignores the fact that a single review may contain multiple opinions on various aspects, which may even be conflicting.

To overcome this limitation, subsequent research has extended to sentiment analysis to the sentence level, where each sentence in a review is assigned an independent sentiment label. Sentence-level sentiment classification has been widely studied using both traditional machine learning models and neural approaches, ranging from early dependency-based methods [42] to benchmark datasets such as the Stanford Sentiment Treebank [57] and

neural architectures like convolutional networks [26]. Despite these significant advances, sentence-level sentiment analysis still cannot fully capture the diversity of opinions in a single sentence. For instance, considering a sentence like "The laptop design is good, but the battery life is unsatisfactory," the sentiment towards *design* is positive, while the sentiment toward *battery life* is negative. This limitation has motivated the development of Aspect-Based Sentiment Analysis (ABSA), a fine-grained task that aims to classify the sentiment polarity expressed toward a particular aspect within a sentence [49].

ABSA is typically formulated as a combination of two subtasks: aspect term extraction (ATE) and aspect sentiment classification (ASC) [84]. ATE identifies and extracts explicit aspects or opinion targets mentioned in a review, while ASC determines the sentiment polarity expressed toward each identified aspect. Despite many NLP methods and algorithms for ABSA have undergone remarkable advances, many significant challenges remain. One of the most critical issues is the reliance on large-scale annotated data for each aspect or domain. Since the vocabulary and sentiment expressions differ greatly across domains, models trained on one domain often experience severe performance deterioration when applied to an out-of-distribution domain. This deterioration can be attributed to the fact that most NLP algorithms assume that the training and test data are drawn from the same underlying distribution [52]. Consequently, there remains a considerable gap between current ABSA research and its real-world deployment, where annotated data for new domains or aspects is often scarce or entirely unavailable.

Domain Adaptation (DA) addresses the domain shift problem by enabling NLP algorithms to adapt to out-of-distribution domain. Traditional DA research in NLP has primarily focused on supervised approaches, in which a small amount of labeled data from the target domain is combined with a large quantity of labeled data from the source domain [3, 44]. Although such approaches improve transferability from source to target domains, acquiring labeled data in the target domain requires substantial manual effort. To alleviate this issue, Unsupervised Domain Adaptation (UDA) has been extensively studied, where unlabeled data from the target domain is leveraged to enhance generalization [52]. However, most UDA methods assume that the target domain is at least known and accessible during training, even if its labels are unavailable. This assumption does not always hold in real-world scenarios. For example, when a new product or service is released, customer reviews for it have not yet been accumulated, meaning that even unlabeled data from the target domain may be unavailable. In such cases, a company may still wish to automatically analyze the opinions of pioneer customers, but the model's generalization ability can be severely limited when confronted

with entirely unseen domains. A more challenging but relatively underexplored setting of DA is thus to adapt models to any possible unknown target domains that are not available during training, a scenario often referred to as domain generalization.

Another important challenge is the imbalance of sentiment labels across domains. In many benchmark datasets, positive instances substantially outnumber neutral or negative ones, which makes it difficult for models to learn robust decision boundaries for the underrepresented classes, especially the neutral category. The neutral class is often ambiguous and context-dependent, leading to annotation inconsistencies and degraded classification accuracy. Addressing such imbalance is critical for building reliable ABSA systems that perform consistently across diverse domains.

These limitations highlight the need for approaches that can generalize across domains and aspects, even under conditions of limited or missing supervision in the target domain. To address this gap, this dissertation explores domain adaptation for ABSA under two complementary scenarios: (1) UDA, where unlabeled target-domain data is available, and (2) domain-agnostic adaptation, where the target domain is entirely unknown during training.

## 1.2 Research Goals

The overall goal of this dissertation is to address the domain shift problem in the ASC by systematically investigating how models can adapt to different levels of target-domain availability. In real-world applications, sentiment analysis systems may encounter two representative scenarios:

- **Scenario A (Unsupervised Domain Adaptation, UDA):** The target domain is available in the form of unlabeled data. This setting allows models to exploit domain-specific distributions without relying on any annotated labels. It reflects scenario where sufficient customer feedback has accumulated, but manual labeling is infeasible.
- **Scenario B (Domain-Agnostic Adaptation):** The target domain is completely unknown during training. In this case, models must generalize to unseen domains without any target-domain supervision, a setting that reflects real-world scenario such as the release of a new product or service with no prior reviews.

By jointly exploring these two complementary scenarios, this dissertation provides a unified perspective on domain adaptation for ABSA, spanning

both data-limited and data-absent scenarios. Our contributions can be summarized as follows.

- We consider two representative scenarios in domain adaptation for ABSA. We address the domain shift problem in ABSA by explicitly studying two realistic settings: (1) Unsupervised Domain Adaptation, where unlabeled target-domain data is available; and (2) Domain-Agnostic Adaptation, where the target domain is entirely unseen during training. To the best of our knowledge, the latter setting has not been actively studied in terms of ABSA.

- For the first scenario, we develop aspect-oriented strategies that use unlabeled target data through automatic labeling, cross-aspect review generation, and filtering mechanisms. These methods leverage domain-specific information without requiring any annotated target domain data, and further mitigate class imbalance with loss re-weighting techniques.

- For the second scenario, we propose Aspect-Enhanced Prompting, which integrates aspect-related features into prompt-based generative sentiment classification. We further enhance robustness by prompt rescoring module and expanding them through clustering-based strategies.

- We conduct extensive experiments across multiple benchmark datasets in both scenarios, demonstrating that the two proposed methods consistently outperform strong baselines. In addition, we conduct detailed analysis and ablation studies for each scenarios.

## 1.3 Research Questions

Based on the above goals, this dissertation is guided by one main research question.

**Major Research Question:** How can ASC models be effectively adapted to domain shift under different levels of target domain availability: (1) with unlabeled target data, and (2) with no target-domain access?

To address this question, the study is organized into the following sub-questions:

- **RQ1:** How can we construct or augment effective training data when labeled target-domain data is unavailable?

- **RQ2:** How can we extract and integrate aspect-related information or features to guide sentiment classification in a way that remains useful?

- **RQ3:** How can we mitigate noise introduced by pseudo-labels, generated sentences, or irrelevant features?

## 1.4 Chapter Organization

The rest of this dissertation is organized into four parts as follows:

- **Chapter 2** presents a comprehensive review of the literature. It introduces the fundamental concepts and tasks of ABSA, with a particular emphasis on ASC. Then it introduces prior work on domain adaptation for ABSA, including approaches to UDA and multi-source domain adaptation.

- **Chapter 3** investigates Scenario A, where unlabeled target-domain data is available during training. It formalizes the problem setting and presents the proposed framework, which integrates automatic labeling with cross-aspect review generation to leverage unlabeled target data. The chapter also explains how Focal Loss is employed to mitigate class imbalance. Finally, the evaluation of the proposed method is described, including the experimental setup, reporting results on benchmark datasets, and providing detailed analysis.

- **Chapter 4** investigates Scenario B, where the target domain is entirely unavailable during training. It introduces the Aspect-Enhanced Prompting (AEP) framework, which leverages Aspect-Related Features (ARFs) to construct prompts for guiding sentiment classification in a domain-agnostic manner. The chapter further describes the prompt rescoring and cluster-based prompt expansion modules. At last, the chapter describes extensive evaluations, including ablation studies, parameter sensitivity analysis, and error analysis.

- **Chapter 5** concludes the dissertation by summarizing the main contributions, and answering the research questions. It also outlines promising directions for future work.

# Chapter 2

# Literature Review

This chapter provides an overview of prior research relevant to this dissertation. It is organized into two main parts. The first part introduces Aspect-Based Sentiment Analysis, the central task studied in this dissertation. This part reviews the definition of opinions and aspects, outlines the main subtasks of ABSA, and introduces the evolution of methods for Aspect Sentiment Classification, ranging from early neural network models to pre-trained language models and recent generative or prompt-based approaches. The second part focuses on domain adaptation techniques as they are apply to ABSA in this study. This part reviews research on unsupervised domain adaptation and multi-source domain adaptation, highlighting both classical feature-based methods and more recent generative and prompt-based paradigms. Together, these reviews establish the research context and identify the problems that motivate the contributions of this dissertation.

## 2.1 Aspect-Based Sentiment Analysis

Aspect-Based Sentiment Analysis is a fine-grained sentiment analysis task that aims to analyze the sentiment of opinions at the aspect level [84]. Unlike traditional sentiment analysis, which focuses on document-level or sentence-level sentiment classification, ABSA provides a fine-grained view by associating sentiment polarity with specific aspects or attributes mentioned in the sentence.

Pang and Lee described sentiment analysis or opinion mining generally as the task of analyzing opinions, namely, "a sentiment is basically an opinion that a person expresses towards an aspect, entity, person, event, feature, object, or a certain target" [46]. To study this problem systematically, several studies have attempted to give a formal definition of what constitutes an

"opinion." Such formalization is important because it provides a clear mathematical framework for representing the elements involved in ABSA, and it also allows different methods to be compared under a common notation. In early work, the term *feature* was more commonly used to denote product attributes [22], whereas in recent years, the term *aspect* has been more widely adopted in the literature than *feature*.

In particular, Liu defined an opinion as a quintuple:

$$(e_i, a_{ij}, s_{ijkl}, h_k, t_l), \tag{2.1}$$

where $e_i$ is an entity (e.g., a restaurant), $a_{ij}$ is the $j$-th aspect of $e_i$ (e.g., service), $s_{ijkl}$ represents the sentiment expressed by an opinion holder $h_k$ at time $t_l$ toward $a_{ij}$, which can be positive, negative, or neutral [33]. For instance, Table 2.1 illustrates this quintuple representation with a concrete example. In the review "The service was excellent." the opinion can be represented as ($e_i$ = *restaurant*, $a_{ij}$ = *service*, $s_{ijkl}$ = *positive*, $h_k$ = *reviewer*, $t_l$ = *time of review*). This formalization highlights the fine-grained characteristics of ABSA. Rather than coarse document-level or sentence-level sentiment classification, it explicitly links a sentiment polarity to a particular aspect of an entity.

Table 2.1: Quintuple representation of "The service was excellent." in ABSA.

| **Component** | **Example Value** |
|---|---|
| Entity ($e_i$) | Restaurant |
| Aspect ($a_{ij}$) | Service |
| Sentiment ($s_{ijkl}$) | Positive |
| Opinion Holder ($h_k$) | Reviewer |
| Time ($t_l$) | Time of review |

### 2.1.1 Subtasks of ABSA

ABSA can be decomposed into several subtasks, each of which corresponds to a different linguistic unit of the opinion quintuple. These subtasks are closely related but conceptually distinct, and together they form the foundation of ABSA. The most commonly discussed subtasks include aspect term extraction (ATE), opinion term extraction (OTE), aspect category detection (ACD), and aspect sentiment classification (ASC).

- **Aspect Term Extraction:** ATE is the task of identifying and extracting explicit aspect expressions in a given sentence. An aspect

term usually appears as a noun or noun phrase that refers to a specific component, attribute, or feature of an entity. For example, in the review "The battery lasts long," the word "battery" is an aspect term that points to a concrete feature of the entity "laptop." Formally, ATE corresponds to locating $a_{ij}$ in the opinion quintuple $(e_i, a_{ij}, s_{ijkl}, h_k, t_l)$, where $a_{ij}$ is explicitly mentioned in the sentence.

Researches on ATE have evolved into supervised, semi-supervised, and unsupervised methods. Early supervised approaches formulated ATE as a token-level sequence labeling problem and employed CRF [79] or CNN-based [72] models, often enhanced by domain-specific representation. To reduce the reliance on large labeled data, semi-supervised methods introduced data augmentation approaches by generating pseudo-labeled data for training. More recently, unsupervised ATE has been explored through neural network based methods such as Attention-based Aspect Extraction model [20], POS-based extraction with contrastive attention [64], and self-supervised contrastive learning techniques [55]. Those methods enable aspect term extraction without labeled data.

- **Opinion Term Extraction:** OTE focuses on extracting the words or phrases that carry sentiment toward a given aspect. Unlike aspect terms, which are usually nouns, opinion terms are often adjectives, adverbs, or verbs that express sentiment. For example, in "The service was excellent," the word "excellent" is an opinion term that conveys positive sentiment. Within the quintuple framework, OTE helps to identify the linguistic expressions that correspond to $s_{ijkl}$, i.e., the sentiment component.

  Since opinion expressions are meaningful only when linked to the aspects they describe, researches on OTE has centered on modeling this dependency, giving rise to two main task formulations: aspect opinion co-extraction (AOCE) and target-oriented opinion word extraction (TOWE). AOCE extracts aspect terms and opinion terms jointly, typically using sequence labeling models with dual label sets. Because of the dependency between aspect terms and opinion terms, many AOCE approaches focus on modeling their interactions through dependency-tree based methods [66] or attention mechanisms [67]. In contrast, TOWE assumes that an aspect term is given and aims to extract its associated opinion expressions. Methods for TOWE often incorporate the aspect into the sentence representation, such as through aspect-fused LSTM encodings [9] or syntactic distance features [50], to better

differentiate which opinion words correspond to the given aspect.

- **Aspect Category Detection:** ACD, sometimes referred to as aspect category classification, is the task of mapping aspects to predefined categories. Unlike ATE, which identifies aspect terms as they appear in text, ACD assigns aspects to higher-level semantic categories such as *Service*, *Food*, or *Price* in the restaurant domain. A key challenge in ACD is handling implicit aspects, which are not directly mentioned but can be inferred from context. The importance of implicit aspect detection was also highlighted in the SemEval-2014 shared task on ABSA [49]. For instance, the sentence "This dish was too expensive" contains no explicit mention of the word "price," yet the opinion clearly targets the *Price* category. In terms of the opinion quintuple, ACD corresponds to generalizing $a_{ij}$ into a broader semantic class.

  Studies on ACD have followed two main directions: supervised and unsupervised aspect category detection. Supervised ACD is typically formulated as a multi-label classification problem, where early approaches such as RepLearn relied on learned word representations and feed-forward architectures [86], while later models employed attention mechanisms [41]. In contrast, unsupervised ACD is typically implemented through a two-step process: first extracting candidate aspect terms and then clustering or mapping them to predefined categories [20].

- **Aspect Sentiment Classification:** ASC is the task of determining the sentiment polarity associated with an aspect, once the aspect has been identified or assumed. For instance, in the sentence "The service was excellent," the aspect "service" is linked to the sentiment label *positive*. Formally, ASC assigns a value to $s_{ijkl}$ in the quintuple $(e_i, a_{ij}, s_{ijkl}, h_k, t_l)$, selecting from a predefined set of polarity classes such as positive, negative, or neutral. Unlike document-level or sentence-level sentiment analysis, which yield a single sentiment label for a large text span, ASC requires the model to differentiate between sentiments directed at different aspects within the same sentence.

These subtasks represent the single-task formulation of ABSA, where each task focuses on one component of the opinion quintuple. Beyond these fundamental tasks, ABSA also has various compound tasks, such as joint aspect-opinion pair extraction or end-to-end ABSA that integrate multiple subtasks into a unified prediction problem. However, such compound settings fall outside the scope of this dissertation. Our work concentrates on ASC, which classifies the sentiment polarity associated with an identified aspect. Furthermore, ASC has received the most attention, as it directly determines the

sentiment orientation of aspects and serves as the core task in many ABSA applications. In this dissertation, we specifically address the task of ASC in challenging scenarios, focusing on unsupervised domain adaptation and domain-agnostic adaptation. Next, we will primarily introduce the latest research progress related to ASC.

### 2.1.2 Aspect Sentiment Classification

Aspect Sentiment Classification is the most widely studied subtask in ABSA, as it directly assigns sentiment polarity to aspects within a sentence. The research on ASC has evolved substantially over the past decades, progressing from deep neural networks to approaches based on pre-trained language models and recent to generative models.

#### 2.1.2.1 Neural Network based Methods

Neural networks significantly advanced ASC by enabling models to automatically learn representations of aspects and their surrounding contexts, reducing the need for manual feature engineering. Among neural architectures, Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have been the most widely studied.

CNN-based approaches were applied to ASC because of their efficiency in capturing local n-gram features. However, standard CNNs are limited in their ability to model sequential dependencies and aspect-specific context, which are essential for distinguishing sentiment toward different aspects in the same sentence. To address this, Xue and Li proposed the Gated Convolutional Network, which integrates a gating mechanism into CNN layers [75]. Their Gated Tanh-ReLU Units selectively output sentiment features conditioned on the given aspect.

LSTM-based models have become more suitable for ASC because of their ability to capture long-term dependencies in text. One early baseline LSTM approach was the Target-Dependent LSTM (TD-LSTM) model [60]. This model was motivated by the observation that standard LSTM representations fail to account for the semantic relatedness between a target aspect and its surrounding context words. In many cases, different context words exert different influences on the sentiment polarity toward a specific aspect, making it essential to explicitly integrate target information into the modeling process. To address this, TD-LSTM employs two separate LSTMs to encode the left and right contexts of the target, thereby modeling sentiment with respect to the aspect's position in the sentence. Building on this, the ATAE-LSTM model [68] introduced aspect embeddings directly into the in-

put and attention mechanism, allowing the network to selectively focus on words that are most relevant to a given aspect. This combination of LSTM and attention significantly improved performance, as it addressed the challenge of mixed sentiments toward different aspects in the same sentence.

Overall, neural network approaches such as CNNs, Recurrent Neural Networks (RNNs), LSTMs, and their attention-based extensions have become the foundation of modern ASC. They effectively capture both local patterns and sequential dependencies, while attention mechanisms enhance the model's ability to focus on aspect-relevant sentiment expressions. Despite their success, these models often require large labeled datasets and remain sensitive to domain shifts, which has motivated the later development of pre-trained language models.

#### 2.1.2.2 Pre-trained Language Models based Methods

The emergence of large pre-trained language models (PLMs) has revolutionized natural language processing in many tasks, including ASC. Unlike earlier neural models that relied on task-specific training from scratch, PLMs such as BERT [8] provide contextualized word representations learned from massive corpora. These contextual embeddings capture both syntactic and semantic information, making them highly effective for fine-tuning on ABSA datasets. An early study by Hoang et al. [21] demonstrated that directly fine-tuning BERT for aspect-level sentiment classification substantially outperformed LSTM and CNN based baselines on SemEval datasets. Li et al. systematically apply BERT to ABSA, proposed an end-to-end framework that jointly modeled aspect extraction and sentiment classification, which was one of the first attempts to systematically apply BERT to ABSA [31]. Their results on the SemEval benchmark datasets demonstrated substantial improvements over LSTM- and CNN-based baselines, establishing new state-of-the-art results on SemEval datasets. This work highlighted the advantage of contextualized embeddings in context-sensitive sentiment expressions.

Beyond these early applications of PLMs to ABSA, subsequent research explored how to better adapt PLMs to the aspect-level setting. Song et al. proposed the Attentional Encoder Network (AEN), which replaced recurrent components with self-attention mechanisms and leveraged BERT embeddings for input representation [58]. This design reduced the reliance on recurrent structures such as LSTMs, resulting in a more efficient architecture that was better aligned with transformer-based models.

Xu et al. introduced Position-aware BERT (PBERT), which incorporated explicit position embeddings of aspect terms into the transformer architecture [74]. While standard BERT representations do not encode the rela-

tive distance between aspects and their contexts, PBERT enhanced aspect-context interactions by modeling positional information more explicitly. This extension alleviated a major limitation of vanilla BERT in ABSA, namely its difficulty in distinguishing multiple aspects within the same sentence.

In summary, pre-trained language models have brought a significant improvement in ASC performance by providing rich contextual embeddings and reducing the reliance on handcrafted features. Nevertheless, they still require labeled data in the target domain for effective fine-tuning, and their generalization across unseen domains remains limited. Furthermore, they suffer from performance degradation in cross-domain transfer, highlighting the need for adaptation techniques. These limitations motivate the exploration of generative and prompt-based paradigms, which aim to reduce data dependence and improve cross-domain adaptability.

#### 2.1.2.3 Graph-based Methods

In addition to sequential neural models and pre-trained language models, another significant direction in ASC focuses on graph representation methods. Graph-based Methods address a key limitation of sequence encoders: aspect-opinion relationships often follow syntactic structure rather than word order. Graph-based ASC models therefore represent a sentence as a dependency graph and apply Graph Neural Networks (GNNs) to propagate relevant information between tokens.

Zhang et al. proposed the earliest graph-based models Graph Convolutional Network (GCN) [82] for ABSA. GCN builds a Graph Convolutional Network on top of the dependency tree of a sentence, enabling the model to exploit syntactic relations and long-distance dependencies that are inaccessible to sequential encoders. By designing an aspect-specific graph representation, the model selectively aggregates information from syntactically relevant context words while filtering out irrelevant ones. Experiments on multiple benchmark datasets confirmed that GCN effectively captures syntactic constraints and long-range sentiment indicators, yielding comparable performance compared to state-of-the-art attention-based and CNN-based models.

Following this direction, Huang et al. introduced a Syntax-aware Graph Attention Network (SAGAT) for ASC [24]. Instead of using GCNs with uniform neighborhood aggregation, SAGAT incorporates graph attention mechanisms to dynamically weight different dependency edges. This allows the model to more precisely model the interactions between aspect terms and their syntactically related words. Furthermore, pre-trained contextual embeddings from BERT are integrated into the graph structure, and subword-

level representations are aligned with the dependency tree. By combining syntactic structure with pre-trained language model knowledge, SAGAT obtains more accurate context representations and demonstrates improvements over prior sequential and graph-based approaches.

To further advance graph-based sentiment modeling, Li et al. proposed Dual Graph Convolutional Networks (DualGCN) [30], addressing two notable limitations of dependency-based ASC models: inaccuracies in dependency parsing and the informal, syntactically irregular nature of user-generated reviews. DualGCN leverages the complementary strengths of syntactic structure and semantic correlation by constructing two parallel graphs. The SynGCN module encodes dependency-based syntactic relations while enriching them with additional syntactic knowledge to mitigate parsing errors. In contrast, the SemGCN module captures semantic associations between words using a self-attention mechanism that learns soft relational structures beyond the dependency tree. Through joint learning over syntactic and semantic graphs, DualGCN effectively integrates structural and contextual information. Experiments on several datasets demonstrate that DualGCN achieves state-of-the-art performance, confirming the benefit of combining dual graph structures for robust aspect-level sentiment classification.

#### 2.1.2.4 Generative language models and prompt-based Methods

With the rapid progress of large-scale sequence-to-sequence models, researchers have begun to reformulate ABSA as a generation or prompt-based task. Unlike traditional discriminative approaches that design task-specific classifiers, generative and prompt-based methods leverage the pre-training objectives of language models and unify multiple ABSA subtasks under a single formulation. This paradigm shift has shown strong potential in improving generalization across tasks and domains, especially in low-resource and cross-domain scenarios.

Zhang et al. first proposed a unified generative framework for ABSA, named as Towards Generative ABSA [83]. In this approach, various ABSA subtasks (e.g., aspect extraction, opinion extraction, and sentiment classification) are converted into sequence generation problems. They designed two modeling paradigms, annotation-style and extraction-style generation, where the model produces outputs such as aspect-opinion-sentiment triplets in natural language form. Experiments on four ABSA tasks across multiple benchmark datasets demonstrated that this generative reformulation achieved new state-of-the-art results, validating the generality of a unified generative framework.

Based on this idea, Yan et al. proposed a more comprehensive Unified

Generative Framework for ABSA [76]. They redefined all seven ABSA subtasks into a generative sequence format by mixing pointer indexes with sentiment class indexes, enabling a single BART-based encoder-decoder model to solve aspect term extraction, opinion term extraction, aspect sentiment classification, and their joint variants simultaneously. This framework offered a real end-to-end solution for the full ABSA pipeline, avoiding the need for separate models for different subtasks. Extensive experiments showed substantial performance gains across multiple datasets, establishing generative modeling as a strong and scalable alternative to classification-based methods.

More recently, prompt-based approaches have been explored to adapt pre-trained language models to ABSA in a knowledge-aware manner. Li et al. proposed SentiPrompt, a sentiment knowledge enhanced prompt-tuning method [28]. They injected domain knowledge about aspects, opinions, and sentiment polarity into natural language prompts, and explicitly modeled relations among terms via consistency and polarity judgment templates. By incorporating knowledge-enhanced prompts, SentiPrompt achieved notable improvements on triplet extraction, pair extraction, and aspect-sentiment classification. This marked a paradigm shift from discriminative classification to generation and prompting, which directly motivates the approaches developed in this dissertation.

In summary, the research on ASC has progressed from neural architectures to pre-trained and generative models, steadily improving performance on benchmark datasets. However, a major limitation is that most of these models assume that training and test data come from the same domain, which restricts their applicability in real-world scenarios. This challenge, commonly known as *domain shift*, motivates the exploration of domain adaptation techniques for ABSA, which will be reviewed in the following section.

## 2.2 Domain Adaptation for ABSA

In real-world applications, sentiment classification models are often required to operate across domains where labeled data are scarce or unavailable. For example, a model trained on restaurant reviews may need to be applied to laptop or service reviews, where the vocabulary and sentiment expressions differ significantly. This mismatch between the source and target domains is commonly referred to as domain shift, and it presents a major challenge in ABSA. To address this, domain adaptation techniques have been widely studied, particularly in the settings of unsupervised domain adaptation (UDA) and multi-source domain adaptation (MSDA). This section provides a review of these two research areas with a focus on their applications in ABSA.

### 2.2.1 Unsupervised Domain Adaptation

UDA assumes that labeled data are available in the source domain, while only unlabeled data exist in the target domain. Approaches to UDA can be broadly divided into two categories: feature-based methods and generative data augmentation methods [52].

#### 2.2.1.1 Feature-based Methods

Feature-based methods aim to reduce domain discrepancy by learning domain-invariant representations. Two major strategies have been developed: feature augmentation and feature generalization. Feature augmentation approaches rely on pivot features to construct an aligned feature space, identifying shared lexical or semantic cues across domains [3, 44]. Deep learning has further advanced these methods by employing adversarial learning [10] or statistical measures [36] to align source and target feature distributions.

Feature generalization approaches attempt to learn robust latent representations that transfer across domains. Early studies employed autoencoders, such as the stacked denoising autoencoder (SDA) [14] and its efficient variant, the marginalized SDA (mSDA) [7], to extract domain-invariant features by reconstructing noisy inputs. While autoencoder-based models were effective, they were limited by lack of explicit linguistic knowledge. Recent advances therefore leverage PLMs such as BERT [8], which naturally encode rich linguistic and contextual knowledge [19, 25]. A notable example is Source-Free ABSA [85], which combines feature-based adaptation and pseudo-labeling in a source-free setting, represents a more practical setting where the source data cannot be accessed due to privacy or storage constraints..

#### 2.2.1.2 Generative Data Augmentation Methods

Generative Data Augmentation methods attempt to mitigate domain shift by synthesizing target-like training examples. The key idea is to expand the training distribution through interpolation, paraphrasing, or text generation, thereby bridging the gap between source and target data. For instance, Mixup interpolation, originally proposed for image classification, has been adapted to NLP by interpolating word or sentence embeddings, yielding improvements in cross-domain sentiment classification [17]. Other studies have employed review generation models to produce synthetic reviews resembling target-domain text, which improves robustness [80]. More recently, Refined and Synthesis Data Augmentation (RSDA) introduced a two-stage framework, first filtering pseudo-labeled target data with natural language

inference, and then synthesizing diverse labeled samples through label composition and paraphrasing [65]. These approaches highlight the growing role of generative methods in cross-domain ABSA.

### 2.2.2 Multi-Source Domain Adaptation

While UDA typically considers a single source domain, MSDA extends the setting by allowing labeled data from multiple source domains, with the target domain remaining unlabeled. MSDA poses additional challenges: source domains often vary in distribution, and naively combining them may result in negative transfer. The goal of MSDA is therefore to exploit complementary information across multiple sources while ensuring robust generalization to the target domain.

#### 2.2.2.1 Domain Divergence and Distance-based Methods

A central concept in MSDA is measuring domain divergence. Accurately estimating the similarity between domains enables selective transfer from the most relevant sources. For example, the DistanceNet-Bandit model incorporates distance-based metrics into its loss function and dynamically selects the most informative source domains using a multi-armed bandit controller [16]. This model demonstrated that dynamic source selection could outperform static combinations of multiple domains.

#### 2.2.2.2 Mixture-of-Experts Approaches

Another prominent line of research adopts the mixture-of-experts (MoE) paradigm [18], where each source domain is treated as an expert. The model learns to combine predictions from domain-specific experts to better approximate the target distribution. Transformer-based MSDA models have shown that large PLMs such as BERT are inherently robust to domain variation, and MoE strategies further enhance performance by adaptively weighting domain-specific experts [70]. However, effective ensemble of experts remains challenging, as transformer-based experts often produce homogeneous predictions across different domains.

#### 2.2.2.3 Data Selection and Curriculum Learning

An alternative direction leverages data selection strategies to fine-tune pretrained models for domain adaptation. Ma et al. proposed a BERT-based method that incorporates domain classification and curriculum learning to select training samples most relevant to the target domain [37]. Although this

approach is effective, it requires a small validation set from the target domain to tune hyperparameters, which limits its applicability in fully unsupervised MSDA.

#### 2.2.2.4 Recent Advances

Recent work has proposed more sophisticated mechanisms for cross-domain alignment. Feature Structure Matching (FSM) introduces dynamic parameter fusion and feature structure constraints to improve robustness across domains [29]. FaiMA specifically targets ABSA by integrating linguistic, domain, and sentiment features into a graph attention network, coupled with contrastive learning, to enhance cross-domain transfer in MSDA [77].

## 2.3 Prompt-based Learning

Recent advances in large language models (LLMs) have led to a paradigm shift in NLP, moving from the traditional "pre-train and fine-tune" strategy toward the "pre-train, prompt, and predict" framework [35]. In this new paradigm, downstream tasks are reformulated to align with the input and output style used during language model pre-training, allowing models to perform inference by conditioning on carefully designed textual prompts rather than learning new task-specific parameters. Prompt-based learning thus provides a unified and efficient interface, enabling LLMs to generalize across tasks and domains through linguistic instructions. This section provides an overview of recent studies that apply prompt learning and prompt generation methods to domain adaptation and ABSA.

### 2.3.1 Prompt-based Learning for ABSA

Recent studies have extended prompt-based learning to ABSA in order to unify multiple sub-tasks within a single generative framework. Gao et al. proposed LEGO-ABSA, a task-assemblable unified generative model that formulates various ABSA tasks, such as aspect extraction, opinion extraction, and sentiment classification within the same text-to-text paradigm [11]. Instead of generating the output as an unstructured sequence, LEGO-ABSA decomposes it into multiple controllable elements through task-specific prompts.

Wang et al. proposed UnifiedABSA, a general-purpose ABSA framework based on multi-task instruction tuning [69]. This model jointly learns multiple ABSA tasks with shared parameters and task indicators, enabling inter-task dependency modeling and data-efficient transfer. Instead of training separate models for each task, UnifiedABSA captures common linguistic

patterns through instruction-based prompts, significantly outperforming dedicated task-specific models on 11 ABSA tasks across two benchmark datasets. More recently, Yin et al. proposed a method called SynPrompt, which is a syntax-aware enhanced prompt engineering approach for ABSA [78]. SynPrompt incorporates syntactic dependency structures to capture the relationships between aspects and their associated opinion expressions. By mining key syntactic paths from dependency trees, the model enhances the contextual alignment between aspect and sentiment terms.

### 2.3.2 Prompt-based Learning for Domain Adaptation

Another emerging direction is the integration of prompt-based learning into the domain adaptation task. Recent research in Example-Based Prompt Learning has shown significant potential in improving the performance of domain adaptation, especially in scenarios where the target domain is unseen during training [27]. Instead of relying solely on parametric representations, models in this framework use specific examples from the training data to make predictions. Notably, recent work demonstrates that by tuning soft prompts, large models like T5 can be adapted to various tasks with far fewer parameters while maintaining competitive performance compared to fully fine-tuned models.

Gao et al. proposed LM-BFF (better few-shot fine-tuning of language models), which improves few-shot fine-tuning by using both manually designed and automatically searched prompts [12]. The framework converted each input into a cloze-style prompt, where label words were carefully selected to represent different classes, and applied a calibration procedure to mitigate verbalizer bias. Through this simple yet effective design, LM-BFF demonstrated that properly constructed prompts and label mappings could significantly enhance few-shot generalization and stability across domains.

Prompt learning for on-the-fly Any-Domain Adaptation (PADA) introduced a dynamic framework in which example-based prompts are generated for each target instance, facilitating adaptation to unseen domains [2]. Instead of pre-defining static templates, PADA dynamically constructs prompts conditioned on both source-domain examples and target-domain representations. This mechanism enables the model to perform instance-level adaptation, leveraging multi-source knowledge to infer task-relevant prompts even without target-domain supervision.

Prompt Tuning with Domain Knowledge (PTDK) further extended this idea by combining hard prompts and trainable soft prompts to integrate structured domain knowledge [59]. By embedding domain-specific concepts or sentiment lexicons into prompt representations, PTDK bridged discrete

symbolic knowledge and continuous prompt tuning. This hybrid design enhanced the interpretability and robustness of the adaptation process, showing the potential of combining structured knowledge with lightweight prompt-based learning.

## 2.4 Limitations of Existing Approaches on UDA and MSDA

Despite significant progress in UDA and MSDA, existing approaches typically assume some level of target domain awareness. Most existing approaches rely on target-domain data, either unlabeled reviews, or validation datasets to guide adaptation or construct domain-specific intermediate representations. These methods also typically define domain at the category level, treating each dataset (e.g., restaurant, laptop) as a distinct domain.

In contrast, this dissertation considers two domain adaptation scenarios that differ fundamentally from prior work. First, although the first scenario resembles the standard UDA setting, the notion of domain is defined at the aspect level rather than the category level. Each aspect constitutes its own domain, and adaptation is performed across aspects, not across categories. This distinction is crucial because aspect-level domain shifts are fine-grained and have not been the primary focus of prior UDA research.

Second, we further introduce a domain-agnostic scenario, which departs completely from conventional UDA or MSDA assumptions. In this setting, no target-domain data, neither labeled or unlabeled are available during training time. The model is trained exclusively on source-domain aspects and is required to generalize directly to the entirely unseen aspects or domains at inference time.

These two scenarios demonstrate that our work not only revisits domain adaptation from an aspect-oriented perspective but also extends it to a previously unexplored domain-agnostic setting, which is not addressed by existing UDA or MSDA methods.

# Chapter 3

# Aspect-oriented Unsupervised Domain Adaptation

## 3.1 Problem Setting

In this chapter, we address the task of ASC, which is a foundamental subtask of ABSA. We focus the scenario that data from the target domain is available during the model training time. Specifically, in addition to a source domain dataset $D_S = (x_i^S, y_i^S)$ consisting of labeled review texts, we can also access a target domain dataset $D_T = x_j^T$, which only contains unlabeled aspect-level review texts. The goal is to learn a sentiment classifier trained from labeled source data and unlabeled target data, which can generalize well to the target domain.

Formally, given labeled source domain data and unlabeled target domain data, the objective is to minimize the performance gap between the source and target domains. This setting corresponds to the conventional unsupervised domain adaptation framework, which assumes that a sufficient amount of unlabeled target domain data is available during training. In practice, this setting is realistic when at least a moderate quantity of target domain data can be collected in advance.

Unlike conventional domain adaptation for ABSA, where the different genres of the target products or services are considered as the domains, this thesis proposes a novel method of unsupervised domain adaptation for ABSA where aspects are regarded as distinct domains. Consequently, the knowledge transfer from the labeled data of one aspect to another is explored. Supposing that there is no labeled data for a certain aspect, the goal is to train a polarity classifier for the target aspect using labeled data of a different aspect. For example, a model to classify the polarity of the aspect “service” in restaurant

reviews is derived from labeled training data of the aspect "food".

## 3.2 Overview of the Proposed Approach

An overview of the proposed method is presented in Fig.3.1. In this framework, it is assumed that a set of labeled reviews from a source domain and a set of unlabeled reviews from a target domain are available. Here, the labeled data refers to review sentences annotated with ground-truth polarity labels toward the corresponding aspect. In Fig. 3.1, the symbols (S) and (T) denote the source and target domains, respectively.

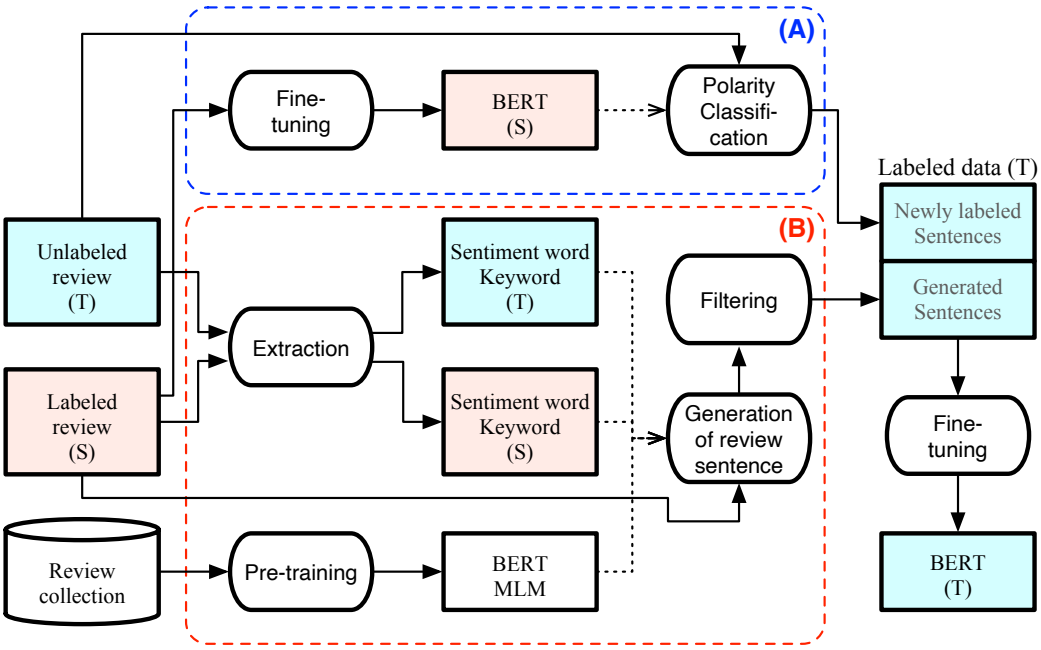


Figure 3.1: Overview of proposed method

The construction of labeled data for the target domain is achieved through two complementary strategies. The first strategy is the automatic labeling of review texts in the target domain, as illustrated by the dotted frame (A) in Fig. 3.1. The second strategy is Cross-Aspect Review Generation (CARG), which generates new labeled reviews for the target domain by modifying labeled reviews from the source domain, as shown in the dotted frame (B) in Fig. 3.1.

The details of these two processes are described in Subsections 3.3.1 and 3.3.2, respectively. Once the labeled data for the target aspect has been constructed by combining the results of these two strategies, a polarity classifier

for the target domain is trained using the constructed dataset, as explained in Subsection 3.3.3.

## 3.3 Proposed Method

### 3.3.1 Automatic Labeling

The first step of the proposed framework is the automatic labeling of unlabeled review texts in the target domain. To this end, a pre-trained BERT model is fine-tuned on the labeled data of the source domain, resulting in a polarity classifier denoted as BERT(S) in Fig. 3.1. This classifier is then applied to the unlabeled review sentences of the target domain to assign polarity labels.

Since the classifier BERT(S) is trained on a different domain, its predictions are not always reliable. To mitigate the risk of introducing noisy labels into the constructed training set, a confidence-based filtering mechanism is adopted. Specifically, for each unlabeled target-domain sentence, the probability of the predicted polarity is computed. Only those sentences whose prediction probability exceeds a predefined threshold $T_p$ are retained, while the others are discarded. This ensures that the automatically labeled target data maintains a reasonable level of reliability and can subsequently be utilized for classifier training.

Table 3.1 shows several examples of automatic labeling based on the predicted polarity probabilities. In this experiment, the dataset of the Service aspect from the restaurant domain (details are provided in Section 3.4.1) was used to fine-tune a pre-trained BERT model, denoted as BERT(S). When applying BERT(S) to a review sentence from the Food aspect, namely "The sauce is delicious and the crust is perfect," the predicted probabilities for the four sentiment classes were as follows: negative = 0.0062, neutral = 0.0516, conflict = 0.0042, and positive = 0.9380. Since the highest probability corresponded to the positive class and exceeded the predefined threshold of 0.8, the sentence was assigned a pseudo label of "positive." In this case, the gold label was also "positive," indicating that the pseudo label was correctly assigned.

In contrast, when the review sentence "It's true, this place is not cheap" from the Price aspect was evaluated, the predicted probabilities were negative = 0.5107, neutral = 0.0144, conflict = 0.4697, and positive = 0.0052. Although the negative class had the highest probability, the value did not exceed the threshold of 0.8. Therefore, the confidence level was regarded as insufficient, and no pseudo label was assigned to this review.

Another example can be found in the review sentence "People dress in suits or evening gowns as well as shirts jeans" from the Ambience aspect. Here, the predicted probability for the positive class was 0.9933, which exceeded the threshold, and thus the pseudo label "positive" was assigned. However, the gold label of this review was "neutral," showing that in this case the pseudo labeling led to an incorrect assignment.

The choice of the threshold was determined empirically. Through preliminary experiments with different thresholds, we selected the one that provided a reasonable balance between labeling accuracy and the number of reviews to which pseudo labels could be assigned. The detailed settings and data set-specific thresholds are described in section 3.4.1.

Table 3.1: Examples of automatic labeling

| Domain | | Review sentence | Predicted probability | | | | Gold label | Pseudo label (threshold = 0.8) |
|---|---|---|---|---|---|---|---|---|
| Source | Target | | negative | neutral | conflict | positive | | |
| Service | Food | The sauce is delicious and the crust is perfect | 0.0062 | 0.0516 | 0.0042 | **0.9380** | positive | positive |
| Service | Price | It's true, this place is not cheap. | **0.5107** | 0.0144 | 0.4697 | 0.0052 | positive | not assigned |
| Service | Price | Could be pricey without a prix fixe meal. | **0.9904** | 0.0012 | 0.0078 | 0.0006 | negative | negative |
| Service | Ambience | People dress in suits or evening gowns as well as shirts jeans. | 0.0019 | 0.0031 | 0.0017 | **0.9933** | neutral | positive |
| Food | Service | Again, the waitress was awesome. | 0.0002 | 0.0011 | 0.0005 | **0.9982** | positive | positive |
| Food | Price | You'll pay at least double at any other Italian restaurant in the city, and most still don't compare. | 0.0317 | **0.7004** | 0.1307 | 0.1372 | positive | not assigned |

### 3.3.2 Cross-Aspect Review Generation

The second step of the proposed framework is Cross-Aspect Review Generation, which aims to augment the target-domain training data by generating new labeled review sentences. As illustrated in the dotted frame (B) of Fig. 3.1, this process leverages both lexical resources and a masked language model (MLM) to transfer aspect-specific expressions from the source to the target domain. The overall procedure consists of three stages: extraction of sentiment words and domain-specific keywords, generation of candidate sentences using an MLM, and filtering of unnatural sentences.

#### 3.3.2.1 Extraction of Sentiment Words

For each source and target domain, we extract domain-specific sentiment words that frequently appear in the reviews. Sentiment words are words

such as “excellent” or “bad” that explicitly convey the writer’s attitude or evaluation. To identify such words, we employ SentiWordNet [1], a lexical resource that provides sentiment-related scores for all WordNet synsets.

In SentiWordNet, each word sense is associated with a positive score and a negative score, both ranging from 0 to 1. A score closer to 1 indicates a stronger polarity. Since a word may have multiple senses depending on its part of speech (POS), SentiWordNet assigns polarity scores to each sense ranked by its frequency in natural texts.

To compute a sentiment score for a given word, we first conduct POS tagging on the review sentences. In this study, we use the NLTK library. The procedure is as follows:

1. Acquire the review sentence and tokenize it into words.
2. Use the pos_tag function to assign a POS tag to each word.
3. Format the output into POS-tagged sentences for further analysis.

Once POS tagging is completed, we calculate the sentiment score of each word using SentiWordNet:

1. Load the SentiWordNet database, which contains multiple senses for each word according to its POS.
2. For each sense $i$ of word $w$, obtain its polarity score by computing the absolute difference between its positive and negative scores: $|pos(w,i) - neg(w,i)|$.
3. Apply a rank-based weighting to each sense, where higher-frequency senses receive larger weights (e.g., $1, \frac{1}{2}, \frac{1}{3}, \dots$).
4. Sum all weighted scores to obtain the final sentiment score $SS(w)$.

Formally, the sentiment score of word $w$ is defined as:

$$SS(w) = \sum_{i=1}^{n} \frac{1}{i} \cdot |pos(w,i) - neg(w,i)|, \tag{3.1}$$

where $i$ is the rank of the sense by frequency, $pos(w,i)$ and $neg(w,i)$ are the polarity scores of the $i$-th sense, and $n$ is the total number of senses of $w$. Words with $SS(w)$ values greater than a threshold $T_s$ are selected as sentiment words in the domain.

Table 3.2 presents the part-of-speech categories, sense identifiers, and polarity scores of the word “great” in SentiWordNet. The word “great” can

function both as a noun and an adjective. It has one sense as a noun and three senses as an adjective. The order of sense frequency is n.01 → s.01 → s.02 → s.03.

Following Eq. (3.1), the sentiment score of "great" is computed separately for the noun and adjective cases. For the noun usage, the polarity score is calculated only from sense n.01, resulting in Eq. (3.2).

$$SS(great.n) = \frac{1}{1} \cdot |0 - 0| = 0 \tag{3.2}$$

For the adjective usage, the three senses (s.01, s.02, s.03) are combined with weights 1, $\frac{1}{2}$, and $\frac{1}{3}$, respectively. The corresponding polarity score is computed as Eq. (3.3).

$$SS(great.a) = \frac{1}{1} \cdot |0 - 0| + \frac{1}{2} \cdot |0.750 - 0| + \frac{1}{3} \cdot |0.250 - 0.125| = 0.417 \tag{3.3}$$

Table 3.2: Information of the word "great" in SentiWordNet

| Word | POS | Sense ID | Positive score | Negative score |
|---|---|---|---|---|
| great | noun | n.01 | 0.000 | 0.000 |
| | adjective | s.01 | 0.000 | 0.000 |
| | adjective | s.02 | 0.750 | 0.000 |
| | adjective | s.03 | 0.250 | 0.125 |

Table 3.3 further illustrates an example of the calculation of the sentiment scores of words from the Service domain, where POS tagging is applied to the review sentence and the sentiment scores of each words are calculated according to Eq. (3.1).

This process ensures that polarity words, such as "happy" receives higher scores, while reducing scores for ambiguous or weakly polarity words (e.g., "bring" or "couple"). This establishes a reliable emotional expression vocabulary for subsequent review generation.

#### 3.3.2.2 Extraction of Domain-Specific Keywords

In addition to extracting sentiment words, it is also essential to identify domain-specific keywords, which capture the semantic content associated with a particular aspect or domain. These words do not necessarily carry sentiment but are frequently used in reviews to describe domain-relevant entities or attributes. For instance, "dinner" and "dessert" are closely tied to the food domain, while "staff" and "waiter" are characteristic of the service

Table 3.3: Examples of sentiment scores of words in the Service domain.

| Word | POS | Sentiment Score (by Eq. 3.1) | Example Sentence |
|---|---|---|---|
| bring | verb | 0.0173 | It took them 25 minutes to **bring** our appetite. |
| understand | verb | 0.1752 | I **understand** the area and folks; you need not come here for the romantic. |
| friendly | adj | 0.1900 | Service was slow, but the people were **friendly**. |
| happy | adj | 0.6950 | I'm **happy** to have Nosh in the neighborhood, and the food is very comforting. |
| attentive | adj | 0.3333 | Waitstaff is great, very **attentive**. |
| obviously | adv | 0.5000 | The place looked nice, with people **obviously** enjoying their pizzas. |
| couple | noun | 0.0912 | As we were leaving, the **couple** standing by the door said to another waiter, "we're not in a hurry." |

domain. Incorporating such keywords ensures that generated sentences not only contain sentiment but also remain contextually appropriate to the target aspect.

Formally, let $D_a$ denote the collection of review sentences belonging to domain $a$. To extract keywords for domain $a$, we first calculate the term frequency of each word in $D_a$. Next, we remove words that also appear frequently in other domains, since these words are not specific to domain $a$. Finally, we select the top $T_d$ words that occur most frequently in $D_a$ but rarely elsewhere, treating them as domain-specific keywords.

Representative examples of top-ranked TF-IDF keywords for several domains are shown in Table 3.4. As observed, words such as "server" and "attentive" are salient in the service domain, while "chicken" and "tasty" characterize the food domain. These keywords, when combined with sentiment words, provide both sentiment expressiveness and contextual appropriateness for generated reviews.

Table 3.4: Examples of top-ranked words extracted by TF-IDF.

| Aspect | Service | Food | Price | Ambience | Anecdotes |
|---|---|---|---|---|---|
| Top-ranked words by TF-IDF | service | fresh | price | decor | trip |
| | attentive | chicken | reasonable | music | favorite |
| | friendly | delicious | pricey | atmosphere | friend |
| | server | sauce | inexpensive | romantic | recommended |
| | slow | food | cheap | cozy | disappointed |
| | manager | tasty | steal | outdoor | stumbled |
| | minutes | beef | dollars | cramped | neighborhood |
| | prompt | pizza | bargain | loud | highly |
| | helpful | steak | cost | seating | anniversary |

#### 3.3.2.3 Generation of Target-Domain Review Sentences

Using the extracted sentiment words and domain-specific keywords, new labeled reviews in the target domain are generated by modifying existing source-domain reviews. Specifically, sentiment words and keywords in a source review are replaced with [MASK] tokens, which are then filled by an MLM.

We employ a BERT-based MLM that adapted to review text by continued pre-training on a large collection of unlabeled restaurant reviews (e.g., YELP). In this adaptation, 15% of the tokens are randomly masked, and the MLM is trained to predict them, thereby acquiring domain-relevant lexical distributions. This strategy allows the model to better capture review-style lexical semantics compared to generic pre-trained MLM.

Target-domain review sentences are generated by a beam-search strategy:

1. Assume that a source sentence contains $n$ [MASK] tokens. For each [MASK], the MLM predicts a ranked list of candidate tokens. The top $T_k$ tokens are selected, resulting in $T_k$ candidate sentences.

2. For each candidate, the next [MASK] is predicted, again selecting the top $T_k$ substitutions.

3. This process continues until all [MASK] tokens are replaced. To avoid exponential growth of computation cost, after each masking step (except the last), only the top $T_k$ sentences ranked by cumulative prediction probability are preserved. For the final [MASK], the full $T_k \times T_k$ candidates are retained.

4. Each generated sentence inherits the polarity label of its original source sentence.

The full procedure of CARG is presented in Algorithm 1. The inputs are the source-domain labeled reviews $R^s$, sentiment word sets $P^s$ and $P^t$ for source and target domains, and keyword sets $K^s$ and $K^t$, where the subscripts $s$ and $t$ refer to the source and target domains, respectively. The output is the set of newly generated labeled reviews $R^t$ in the target domain.

**Algorithm 1** Cross-Aspect Review Generation (CARG)

**Input:** $R^s = \{(s_k, l_k)\}$, $P^s$, $P^t$, $K^s$, $K^t$
**Output:** $R^t$
$R^t \leftarrow \emptyset$
**for** each $(s_k, l_k) \in R^s$ **do**
    $ms \leftarrow$ replace words in $P^s \cup K^s$ with [MASK] in $s_k$
    $R_{posit} \leftarrow GetMaskPosition(ms, [MASK])$
    $R_{new} \leftarrow \{ms\}$
    **for** $i = 1$ to $n$ **do**
        $R'_{new} \leftarrow \emptyset$
        **for** each $s_j \in R_{new}$ **do**
            $PW \leftarrow Unmask(s_j, [MASK]_i)$
            add all $s_j$ with $[\text{MASK}]_i$ replaced by $w \in PW$ to $R'_{new}$
        **end for**
        **if** $i \neq n$ **then**
            $R_{new} \leftarrow Select(R'_{new}, T_k)$
        **else**
            $R_{new} \leftarrow R'_{new}$
        **end if**
    **end for**
    $R_{select} \leftarrow \emptyset$
    **for** each $r_{new} \in R_{new}$ **do**
        validity_flag $\leftarrow$ True
        **for** each $r_{posit} \in R_{posit}$ **do**
            $r_w \leftarrow GetPositionWord(r_{new}, r_{posit})$
            **if** $r_w \notin K^s \cup K^t$ **then**
                validity_flag $\leftarrow$ False
            **end if**
        **end for**
        **if** validity_flag **then**
            $R_{select} \leftarrow R_{select} \cup \{r_{new}\}$
        **end if**
    **end for**
    **for** each $r_{new} \in R_{select}$ **do**
        $R^t \leftarrow R^t \cup \{(r_{new}, l_k)\}$
    **end for**
**end for**
**Function** $Unmask(s, [MASK], T_k)$
    $PW \leftarrow PredictByMLM(s, [MASK])$
    **return** $Select(PW, T_k)$

For each source review $s$, sentiment and domain-specific keywords are replaced with [MASK] tokens. For each [MASK], the Unmask function predicts candidate substitutions using the MLM, then filters them against the target-domain lexicons ($P^t$ for sentiment words and $K^t$ for keywords). The top $T_k$ candidates are selected. At each step, $T_k$ new sentences are created, and beam-search pruning ensures that only the most likely sentences survive. Finally, the generated set $R_{new}$ is added to the target review set $R^t$.

This algorithm enables a single labeled source review to yield multiple labeled target-domain reviews, significantly enriching the training set for cross-domain sentiment classification.

**Example of Sentence Generation by CARG**

An illustrative example is presented in Table 3.5, where a review sentence from the *service* domain is transformed into sentences in the *food* domain using the CARG procedure. In the table, underlined words represent tokens predicted by the BERT MLM and inserted into the [MASK] positions, while boldface highlights those predictions that correspond to sentiment words or domain-specific keywords in the target domain.

As an illustrative case, we consider the source domain Service. The review sentence "The service was attentive and her suggestions of menu items was right on the mark" is given as the input. In Step 1, domain-specific keywords and sentiment words from the source domain are replaced with [MASK] tokens. In Step 2(a), the MLM predicts the top $T_k$ candidate words for the leftmost [MASK], generating $T_k$ candidate sentences. For instance, words such as "staff," "service," and "food" are filled into the first [MASK]. In Step 2(b), the next [MASK] is expanded in the same way, producing $T_k \times T_k$ candidate sentences. Examples include predictions such as "friendly" and "good." Among these candidates, the top $T_k$ sentences with the highest scores are retained.

Step 3 repeats this process for all remaining [MASK] tokens. At this stage, $T_k \times T_k$ sentences are retained. A generated sentence is only considered as a valid target-domain review if all replaced tokens correspond to domain-specific keywords and sentiment words in the target domain; otherwise, the sentence is discarded. As shown in Table 3.5, the top-6 candidate sentence ("The food was good and her choice of menu items was right on the menu") satisfies the criteria (indicated by ✓), whereas the top-1 to top-5 sentences do not (marked with ✗).

Finally, Step 4 assigns the polarity label of the source-domain review to the generated target-domain sentence. For example, the top-6 candidate sentence is assigned the label "positive." Through this process, CARG en-

ables the transfer of sentiment-labeled reviews across different aspects by generating target-domain sentences that preserve both sentiment polarity and contextual relevance.

Table 3.5: An illustrative example of sentence generation by CARG (Service → Food). Underlined words are predicted by the MLM and filled into [MASK] tokens, while bold words indicate sentiment-bearing or aspect-related words in the target domain. ✓ denotes a valid generated sentence, and ✗ denotes an invalid one.

| **Source-domain review** | The service was attentive and her suggestions of menu items was right on the mark |
|---|---|
| **Step 1** | The [MASK] was [MASK] and her [MASK] of menu items was [MASK] on the [MASK] |
| **Step 2(a)** | top1: The staff was [MASK] and her [MASK] of menu items was [MASK] on the [MASK]<br>top2: The service was [MASK] and her [MASK] of menu items was [MASK] on the [MASK]<br>top3: The food was [MASK] and her [MASK] of menu items was [MASK] on the [MASK]<br>$\cdots$<br>top$k$: $\cdots$ |
| **Step 2(b)** | top1: The staff was friendly and her [MASK] of menu items was [MASK] on the [MASK]<br>top2: The service was good and her [MASK] of menu items was [MASK] on the [MASK]<br>top3: The food was good and her [MASK] of menu items was [MASK] on the [MASK]<br>$\cdots$<br>top$k$: $\cdots$ |
| **Step 3** | top1: The staff was friendly and her **choice** of menu items was **great** on the **menu** (✗)<br>top2: The service was **good** and her **choice** of menu items was **great** on the **menu** (✗)<br>top3: The **food** was **good** and her **choice** of menu items was **great** on the inside (✗)<br>top4: The staff was friendly and her **choice** of menu items was **great** on the inside (✗)<br>top5: The service was **good** and her **choice** of menu items was **great** on the inside (✗)<br>top6: The **food** was **good** and her **choice** of menu items was **right** on the **menu** (✓)<br>$\cdots$<br>top$k$: $\cdots$ |
| **Step 4** | top6: The **food** was **good** and her **choice** of menu items was **right** on the **menu** (✓) (positive)<br>$\cdots$<br>top$n$: $\cdots$ (polarity) |

#### 3.3.2.4 Filtering of Generated Sentences

The sentences generated through CARG contain many unnatural ones. To filter out inappropriate sentences, we evaluate the fluency of each sentence using the pseudo-log-likelihood (PLL) score [54], defined as:

$$\log P_{MLM}(W) = \sum_{t=1}^{|W|} \log P_{MLM}(w_t \mid W_{\setminus t}), \tag{3.4}$$

where $W$ denotes the candidate sentence, $w_t$ the $t$-th token, and $W_{\setminus t}$ the sentence with $w_t$ replaced by [MASK]. $P_{MLM}(w_t \mid W_{\setminus t})$ is the probability that the MLM correctly predicts $w_t$. Reviews whose PLL values fall below a

threshold $T_f$ are removed. This filtering ensures that the constructed dataset for the target aspect maintains both grammaticality and semantic coherence.

### 3.3.3 Training of Polarity Classifier

The final step of the proposed method is the training of a polarity classifier for the target domain. For this purpose, the labeled reviews constructed through the two procedures described in 3.3.1 and 3.3.2 are combined to form the training dataset of the target domain. This dataset integrates both automatically labeled target reviews and synthetically generated reviews, thereby ensuring sufficient coverage of sentiment expressions in the target domain.

A BERT-based model is fine-tuned using the constructed target-domain dataset so that it can directly predict the polarity labels of unseen target reviews.

One challenge in this training process is the severe imbalance in polarity label distributions, as will be reported in Subsection 3.4.1. In particular, positive instances are generally overrepresented, while negative or neutral instances are relatively scarce. Such imbalance often leads to biased models that disproportionately favor the majority class, thereby reducing the effectiveness of ABSA.

To mitigate this issue, we adopt Focal Loss [32] as the optimization objective when fine-tuning BERT. Unlike standard cross-entropy loss, Focal Loss down-weights the contribution of well-classified majority examples while emphasizing that of the minority-class examples which are hard to be classifed. By dynamically adjusting the loss contribution according to prediction confidence, this mechanism alleviates the negative effects of label imbalance and enables the classifier to learn more discriminative decision boundaries.

Formally, given the predicted probability $p_t$ for the ground-truth class, the Focal Loss is defined as:

$$\mathcal{L}_{\mathrm{FL}}(p_t) = -\alpha_t(1-p_t)^{\gamma}\log(p_t), \tag{3.5}$$

where $\alpha_t \in [0,1]$ is a weighting factor to balance classes, and $\gamma \geq 0$ is a focusing parameter that controls how much to down-weight easy examples.

By dynamically adjusting the loss contribution according to prediction confidence, this mechanism alleviates the negative effects of label imbalance and enables the classifier to learn more discriminative decision boundaries. As a result, the final polarity classifier achieves greater robustness and generalization capability across imbalanced datasets, providing reliable performance for the target aspect without the need for manually annotated training data.

## 3.4 Evaluation

### 3.4.1 Datasets

Two benchmark datasets are employed to evaluate the effectiveness of the proposed method.

The first dataset is the SemEval-2014 Task 4 Aspect-Based Sentiment Analysis corpus [48], hereafter referred to as the restaurant dataset. This dataset consists of review sentences about restaurants, annotated with aspect categories and corresponding polarity labels. In our experiments, five aspect categories are treated as separate domains. The overall statistics of the dataset are summarized in Table 3.6. As shown in the table, the distribution of polarity classes is highly imbalanced. While the number of positive instances is relatively large, only a small number of conflict and neutral instances are available, with the exception of the anecdotes aspect. This imbalance presents a challenge for training robust classifiers and highlights the necessity of adopting techniques such as Focal Loss.

The second dataset is the Laptop ACOS (Aspect-Category-Opinion Sentiment) corpus [6], hereafter referred to as the laptop dataset. It contains reviews of laptop products collected from Amazon. To facilitate our experiments, the original fine-grained aspect categories are manually grouped into four coarse categories by the author: general, design, performance, and quality. The statistics of the dataset are provided in Table 3.7. Similar to the restaurant dataset, this corpus also suffers from a severe class imbalance problem, particularly in the neutral category, which contains far fewer samples than the positive and negative categories.

In summary, these two datasets provide diverse domains for evaluation and allow us to assess the adaptability and robustness of the proposed method under varying levels of domain shift and label imbalance.

### 3.4.2 Models and Experiment Settings

For all pairs of aspects, one is used as the source domain and the other is used as the target domain. The following six methods are compared in this experiment. The last three are our proposed methods.

**Baseline 1 (BL1):** The BERT model is fine-tuned with the labeled data of the source domain. No domain adaptation method is applied.

**Baseline 2 (BL2):** The BERT model is fine-tuned with the labeled data obtained by the automatic labeling described in subsection 3.3.1.

Table 3.6: Statistics of restaurant dataset.

| | service | food | price | ambience | anecdotes |
|---|---|---|---|---|---|
| positive | 137 | 542 | 49 | 127 | 472 |
| negative | 12 | 138 | 46 | 55 | 167 |
| conflict | 27 | 54 | 7 | 33 | 24 |
| neutral | 160 | 62 | 6 | 18 | 326 |
| total | 336 | 796 | 108 | 233 | 989 |

Table 3.7: Statistics of laptop dataset.

| | general | quality | performance | design |
|---|---|---|---|---|
| positive | 194 | 285 | 156 | 85 |
| negative | 468 | 114 | 152 | 139 |
| neutral | 22 | 15 | 11 | 31 |
| total | 684 | 414 | 319 | 255 |

**CDRG [81]:** A system similar to CARG, where one sentence is generated from the original sentence by substitution of the sentiment words only. The sentences obtained by the automatic labeling are also used for training.

**CARG (Our1):** The labeled data obtained by automatic labeling (as in **Baseline 2**) and the sentence generation by CARG is used for fine-tuning the BERT.

**CARG+Fil (Our2):** It is similar to **CARG**, but the filtering of unnatural sentences by the PLL score is also applied.

**CARG+FL (Our3):** It is similar to **CARG**, but the Focal Loss is also used for fine-tuning the BERT.

**CARG+Fil+FL (Our4):** The training data is constructed as in **CARG+Fil**, and the Focal Loss is used for fine-tuning the BERT. This is our full model.

In addition, the performance of the in-domain setting (**In-d**), where the source and target domains are the same, is also measured for reference. It can be regarded as a ceiling for the domain adaptation methods.

#### 3.4.2.1 Evaluation Metrics

In the cross-domain experiments, accuracy is used as a primary evaluation metric. Accuracy is defined as the proportion of correctly predicted polarity labels among all predictions. Formally, for a multi-class classification task, accuracy is defined as

$$Accuracy = \frac{\sum_{i=1}^{N} c_{ii}}{\sum_{i=1}^{N} \sum_{j=1}^{N} c_{ij}}, \quad N = 3 \text{ or } 4 \tag{3.6}$$

where $c_{ij}$ denotes the number of samples whose gold label is class $i$ and which are predicted as class $j$. Thus, $c_{ii}$ represents the number of correctly classified samples. In the restaurant dataset, the number of classes is $N = 4$, while in the laptop dataset it is $N = 3$. In the in-domain setting, the macro average of F1-measure of all polarity classes (hereafter "macro F1" in short) In addition, macro F1 is reported to evaluate the balance among different polarity classes.

The accuracy and macro F1 is measured by five-fold cross validation on the reviews in the target domain. The reported accuracy and F1 are the average across five runs.

#### 3.4.2.2 Parameter Settings

The parameters of the proposed method are summarized in Table 3.8. These values are chosen intuitively rather than through empirical hyperparameter tuning, except for $T_p$ and $T_s$, for which we investigate multiple values to analyze their effect on classification accuracy. The number of epochs is set to 10, the learning rate to $3 \times 10^{-5}$, and the batch size to 8 when fine-tuning the BERT. For the restaurant dataset, 30,000 Yelp restaurant reviews are additionally used to re-train the MLM of BERT in CARG. For the laptop dataset, the original pre-trained BERT MLM is used without re-training.

### 3.4.3 Results and Discussion

#### 3.4.3.1 In-domain Experimental Results

Tables 3.9 and 3.10 show the in-domain experimental results on the restaurant and laptop datasets, respectively. Each table reports the accuracy of the five trials in five-fold cross validation, as well as the micro-averaged accuracy across the five runs.

For the restaurant dataset, we observe that the accuracy varies depending on the aspect. In particular, for the price and ambience aspects, there is a

Table 3.8: Parameter settings of the proposed method

| Parameter | Value |
|---|---|
| Threshold $T_p$ for automatic labeling (restaurant) | $\{0.7, 0.8\}$ |
| Threshold $T_p$ for automatic labeling (laptop) | $\{0.5, 0.7, 0.8\}$ |
| Threshold $T_s$ for sentiment word extraction (restaurant) | $\{0, 0.3\}$ |
| Threshold $T_s$ for sentiment word extraction (laptop) | 0 |
| Number of extracted domain-specific keywords $T_d$ | 100 |
| Number of generated sentences in CARG $T_k$ | 100 |
| PLL threshold $T_f$ for filtering generated sentences | $-30$ |
| Hyperparameter $\gamma$ of Focal Loss | 2 |

large variance among the five trials. For example, the accuracy of the second trial for price is only 0.455, while the accuracy of the fourth and fifth trials for ambience is 0.565 and 0.543, respectively, both relatively low compared with other trials. These unstable results contributed to lowering the micro-averaged accuracy. On the other hand, for the service, food, and anecdotes aspects, the accuracy are more stable across trials without large fluctuations.

For the laptop dataset, the accuracy for the general aspect is relatively stable across all trials. For the other aspects, however, some trials show relatively low accuracy. Overall, the results for the laptop dataset are more stable than those of the restaurant dataset, and the micro-averaged accuracy are also higher.

Table 3.9: In-domain experimental results on the restaurant dataset (five-fold cross validation).

| | service | food | price | ambience | anecdotes |
|---|---|---|---|---|---|
| Trial 1 | 0.765 | 0.719 | 0.727 | 0.723 | 0.732 |
| Trial 2 | 0.806 | 0.805 | 0.455 | 0.745 | 0.768 |
| Trial 3 | 0.821 | 0.805 | 0.636 | 0.809 | 0.742 |
| Trial 4 | 0.851 | 0.755 | 0.619 | 0.565 | 0.758 |
| Trial 5 | 0.731 | 0.755 | 0.619 | 0.543 | 0.797 |
| Micro Avg | 0.795 | 0.768 | 0.611 | 0.678 | 0.759 |

#### 3.4.3.2 Cross-domain Experimental Results

Tables 3.11 presents the experimental results on the restaurant dataset. For each pair of source and target aspects. Statistical significance is indicated

Table 3.10: In-domain experimental results on the laptop dataset (five-fold cross validation).

| | general | quality | performance | design |
|---|---|---|---|---|
| Trial 1 | 0.854 | 0.843 | 0.828 | 0.843 |
| Trial 2 | 0.854 | 0.855 | 0.781 | 0.725 |
| Trial 3 | 0.876 | 0.940 | 0.875 | 0.804 |
| Trial 4 | 0.854 | 0.880 | 0.938 | 0.725 |
| Trial 5 | 0.912 | 0.817 | 0.810 | 0.686 |
| Micro Avg | 0.870 | 0.867 | 0.846 | 0.757 |

by the marks * and +, which denote significant improvements over BL1 and BL2, respectively, under the McNemar test ($p < 0.05$). The column "Ave." reports the micro-averaged performance across all domain pairs.

Overall, the proposed methods (Our1, Our2, our3 and Our4) consistently outperform the baseline models. In terms of accuracy, the proposed full model achieve superior performance on 16 out of 20 domain pairs, with significant gains over BL1 for 11 pairs and over BL2 for 8 pairs. A similar trend is observed for macro-F1, where our full model perform equal to or better than the baselines on 15 domain pairs. On average, both accuracy and macro-F1 scores are higher for the proposed methods than for the baselines. Furthermore, comparing Our1 (CARG-based) with CDRG, both accuracy and macro-F1 are improved, demonstrating the effectiveness of substituting aspect- and domain-specific words for cross-aspect adaptation.

The ablation study further clarifies the contributions of each component. Our2, which incorporates the filtering mechanism, consistently outperforms Our1, indicating that removing low fluency sentences effectively improves training quality. Our3, which applies Focal Loss alone without the filtering mechanism, achieves moderate improvements over Our1, suggesting that addressing label imbalance is also beneficial, though its effect is limited when noisy samples remain. Similarly, Our4, which adopts both the filtering mechanism and Focal Loss, surpasses Our2 in both average accuracy and macro-F1. This confirms that Focal Loss alleviates the detrimental impact of label imbalance and contributes to more robust polarity classification.

Tables 3.12 shows the results on the laptop dataset. Similar to the restaurant dataset, all the proposed methods outperform the baselines. However, the number of domain pairs with statistically significant differences is smaller. One plausible explanation is that the MLM for the restaurant dataset was further pre-trained on a large collection of Yelp reviews, whereas the pre-trained MLM was used without additional re-training for the laptop dataset.

Table 3.11: Accuracy and Macro F1 of polarity classification on restaurant dataset

(a) Accuracy of polarity classification

| (source) | S | | | | F | | | | P | | | | Am | | | | An | | | | average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | F | P | Am | An | S | P | Am | An | S | F | Am | An | S | F | P | An | S | F | P | Am | |
| BL1 | .776 | .667 | .687 | .533 | .765 | .574 | .674 | .607 | .679 | .697 | .472 | .421 | .771 | .758 | .620 | .526 | .577 | .707 | .537 | .614 | .633 |
| BL2 | .795 | .676 | .687 | .537 | .771 | .620 | .687 | .607 | .711 | .735 | .459 | .446 | .762 | .794 | .639 | .540 | .574 | .730 | .528 | .597 | .645 |
| CDRG | .785 | .676 | .691 | .547 | .798 | .620 | .674 | .602 | .711 | .747 | .433 | .453 | .804 | .784 | .630 | .533 | .586 | .730 | .537 | .597 | .648 |
| Our1 | $.798^{*}_{+}$ | .694 | .682 | $.540^{*}_{+}$ | $.783^{*}_{+}$ | .630* | .704* | .585 | .708 | .731* | .472 | .461 | .786 | $.789^{*}_{+}$ | .630 | $.531^{*}_{+}$ | .571 | .756* | .546 | .597 | .650 |
| Our2 | $.791^{*}_{+}$ | .731 | .691 | $.541^{*}_{+}$ | .802* | .657* | .691* | .590 | .705 | $.754^{*}_{+}$ | .464 | .433 | $.798^{*}_{+}$ | $.794^{*}_{+}$ | .639 | $.558^{*}_{+}$ | .563 | .727 | $.574^{*}_{+}$ | .597 | .655 |
| Our3 | $.793^{*}_{+}$ | .704+ | .691 | $.539^{*}_{+}$ | $.811^{*}_{+}$ | .649+ | .684* | $.602^{*}_{+}$ | .708 | $.747^{*}_{+}$ | .479* | .446 | $.795^{*}_{+}$ | $.789^{*}_{+}$ | .639 | .529 | .563 | .727 | $.574^{*}_{+}$ | .599* | .653 |
| Our4 | $.799^{*}_{+}$ | .713 | .691 | $.542^{*}_{+}$ | .815* | .657* | .687* | .602 | .717 | $.747^{*}_{+}$ | .476 | .451 | $.795^{*}_{+}$ | $.795^{*}_{+}$ | .639 | .556* | .574 | .731* | $.574^{*}_{+}$ | .605 | .658 |
| In-d | .768 | .611 | .678 | .759 | .795 | .611 | .678 | .759 | .795 | .768 | .611 | .678 | .795 | .768 | .611 | .759 | .795 | .768 | .611 | .678 | |

(b) Macro-F1 of polarity classification

| (source) | S | | | | F | | | | P | | | | Am | | | | An | | | | average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | F | P | Am | An | S | P | Am | An | S | F | Am | An | S | F | P | An | S | F | P | Am | |
| BL1 | .490 | .440 | .459 | .364 | .579 | .387 | .467 | .446 | .360 | .343 | .266 | .250 | .523 | .485 | .338 | .345 | .428 | .494 | .368 | .397 | .411 |
| BL2 | .510 | .447 | .434 | .356 | .568 | .396 | .469 | .404 | .378 | .364 | .260 | .263 | .492 | .489 | .338 | .342 | .381 | .473 | .368 | .380 | .406 |
| CDRG | .503 | .447 | .459 | .367 | .598 | .387 | .467 | .401 | .378 | .373 | .210 | .267 | .538 | .484 | .378 | .354 | .406 | .480 | .378 | .380 | .413 |
| Our1 | $.503^{*}_{+}$ | .457 | .454 | $.364^{*}_{+}$ | $.595^{*}_{+}$ | .383* | .467* | .438 | .379 | .350* | .264 | .245 | .527 | $.494^{*}_{+}$ | .395 | $.358^{*}_{+}$ | .381 | .486* | .394 | .388 | .416 |
| Our2 | $.509^{*}_{+}$ | .476 | .459 | $.352^{*}_{+}$ | .602* | .393* | .464* | .446 | .373 | $.351^{*}_{+}$ | .257 | .252 | $.545^{*}_{+}$ | $.477^{*}_{+}$ | .414 | $.373^{*}_{+}$ | $.397^{*}_{+}$ | .466 | $.378^{*}_{+}$ | .388 | .419 |
| Our3 | $.510^{*}_{+}$ | .447+ | .459 | $.350^{*}_{+}$ | $.607^{*}_{+}$ | .387+ | .459* | $.446^{*}_{+}$ | .379 | $.351^{*}_{+}$ | .257* | .252 | $.541^{*}_{+}$ | $.489^{*}_{+}$ | .414 | .350 | .397 | .466 | $.378^{*}_{+}$ | .391* | .417 |
| Our4 | $.497^{*}_{+}$ | .467 | .459 | $.361^{*}_{+}$ | .609* | .432* | .461* | .428 | .376 | $.371^{*}_{+}$ | .277 | .256 | $.541^{*}_{+}$ | $.508^{*}_{+}$ | .458 | .395* | .381 | .469* | $.377^{*}_{+}$ | .391 | .426 |
| In-d | .524 | .405 | .489 | .632 | .634 | .405 | .489 | .632 | .634 | .524 | .489 | .632 | .634 | .524 | .405 | .632 | .634 | .524 | .405 | .489 | — |

S: service, F: food, P: price, Am: ambience, An: anecdotes.
$*$: vs. BL1 ($p < 0.05$), +: vs. BL2 ($p < 0.05$)

Despite this limitation, Our1 outperforms CDRG in accuracy for 8 out of 12 domain pairs, and also achieves higher average accuracy and macro-F1 scores. Among Our1, Our2, Our3 and Our4, the performance differences are relatively small, with Our2 yielding the best average accuracy and Our1 slightly better macro-F1. These results confirm that CARG provides consistent improvements across domains and datasets.

Finally, when comparing cross-domain and in-domain performance, macro-F1 in the in-domain setting generally exceeds that in the cross-domain setting, as expected. The performance gap is particularly large for certain aspects, such as anecdotes in the restaurant dataset. Nevertheless, there are a few exceptions where cross-domain adaptation performs comparably or even better, e.g., transfers into the "price" aspect in the restaurant dataset or into

Table 3.12: Accuracy and Macro-F1 of polarity classification on laptop dataset

(a) Accuracy of polarity classification

| (source) | G | | | Q | | | P | | | D | | | average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | Q | P | D | G | P | D | G | Q | D | G | Q | P | |
| BL1 | .807 | .784 | .765 | .741 | .837 | .769 | .789 | .850 | .773 | .737 | .821 | .809 | .790 |
| BL2 | .804 | .809 | .757 | .731 | .850 | .765 | .773 | .862 | .788 | .756 | .831 | .803 | .794 |
| CDRG | .804 | .800 | .769 | .731 | .843 | .769 | .780 | .848 | .792 | .756 | .829 | .809 | .794 |
| Our1 | .807 | .803 | .780* | .744+ | .850* | .769 | .768 | .865* | .788 | .770* | .833* | .806 | .799 |
| Our2 | .812 | .800 | .780* | .746+ | .853* | .765 | .779 | .862* | .796* | $.772^{*}_{+}$ | .829* | .815 | .801 |
| Our3 | .807 | .800 | .780* | .744+ | .850* | .765 | .780 | .848 | .788 | .765 | $.833^{*}_{+}$ | .815 | .798 |
| Our4 | .816 | .800 | .780* | .744+ | .846 | .773 | .779 | .855 | .788 | .768* | .829* | .815 | .800 |
| In-d | .867 | .846 | .757 | .870 | .846 | .757 | .870 | .867 | .757 | .870 | .867 | .846 | — |

(b) Macro-F1 of polarity classification

| (source) | G | | | Q | | | P | | | D | | | average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | Q | P | D | G | P | D | G | Q | D | G | Q | P | |
| BL1 | .554 | .549 | .533 | .475 | .575 | .540 | .506 | .568 | .549 | .515 | .514 | .533 | .534 |
| BL2 | .550 | .564 | .520 | .478 | .575 | .543 | .504 | .565 | .546 | .513 | .540 | .551 | .537 |
| CDRG | .549 | .564 | .535 | .476 | .575 | .534 | .505 | .566 | .552 | .514 | .520 | .553 | .537 |
| Our1 | .560 | .566 | .543* | .475+ | .575* | .542 | .504 | .568* | .546 | .527* | .541* | .551 | .542 |
| Our2 | .563 | .559 | .538* | .478+ | .573* | .537 | .507 | .568* | .546* | $.520^{*}_{+}$ | .534* | .557 | .540 |
| Our3 | .560 | .557 | .543* | .475+ | .575* | .546 | .505 | .566 | .546* | .517 | $.540^{*}_{+}$ | .559 | .541 |
| Our4 | .555 | .557 | .528* | .478+ | .575 | .546 | .504 | .565 | .546 | .515* | .536* | .559 | .539 |
| In-d | .685 | .556 | .594 | .661 | .556 | .594 | .661 | .685 | .594 | .661 | .685 | .556 | — |

G: general, Q: quality, P: performance, D: design.
*: vs. BL1 ($p < 0.05$), +: vs. BL2 ($p < 0.05$)

"performance" aspect in the laptop dataset. These anomalies are likely due to the limited sample sizes in certain domains (e.g., "price" and "ambience" in the restaurant dataset; "performance" and "design" in the laptop dataset, as shown in Tables 3.6 and 3.7). To validate this hypothesis, further experiments with larger datasets are required in order to more precisely assess the gap between in-domain and cross-domain settings.

#### 3.4.3.3 Impact of Parameter Settings

We further investigated the impact of two parameters in the proposed framework, the confidence threshold $T_p$ for automatic labeling and the polarity-score threshold $T_s$ for sentiment word extraction.

**Effect of $T_p$.** As described in Section 3.3.1, the parameter $T_p$ controls whether a target-domain review is assigned a label based on the confidence of the source-domain BERT classifier. For the restaurant dataset, we compared BL2 with $T_p = 0.7$ and $T_p = 0.8$. The results in Tables 3.13 show that $T_p = 0.8$ yields higher average accuracy in 14 out of 20 source–target pairs, despite producing slightly fewer labeled sentences. Therefore, $T_p = 0.8$ was adopted in the cross-domain experiments on the restaurant dataset.

For the laptop dataset, we compared $T_p = 0.5, 0.7$, and 0.8. As shown in Tables 3.14, higher thresholds reduce the number of labeled reviews. In some cases (e.g., when the source domain is *design* or *quality*), high thresholds lead to lower classification accuracy. Overall, $T_p = 0.5$ achieved the highest average accuracy in 9 out of 12 domain pairs, and was therefore adopted in the laptop experiments.

Table 3.13: Accuracy and number of labeled reviews with different thresholds $T_p$ (restaurant dataset).

(a) Accuracy of polarity classification

| (source) | Service | | | | Food | | | | Price | | | | Ambience | | | | Anecdotes | | | | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | F | P | Am | An | S | P | Am | An | S | F | Am | An | S | F | P | An | S | F | P | Am | |
| BL2 ($T_p$=0.7) | .786 | .657 | .700 | .532 | .762 | .611 | .665 | .601 | .702 | .667 | .455 | .437 | .807 | .780 | .630 | .547 | .577 | .720 | .537 | .614 | .639 |
| BL2 ($T_p$=0.8) | .795 | .676 | .687 | .537 | .771 | .620 | .687 | .607 | .711 | .735 | .459 | .446 | .762 | .794 | .639 | .540 | .574 | .730 | .528 | .597 | .645 |

(b) Number of labeled reviews

| (source) | Service | | | | Food | | | | Price | | | | Ambience | | | | Anecdotes | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | F | P | Am | An | S | P | Am | An | S | F | Am | An | S | F | P | An | S | F | P | Am |
| BL2 ($T_p$=0.7) | 738 | 96 | 214 | 866 | 308 | 93 | 213 | 885 | 265 | 617 | 169 | 701 | 248 | 654 | 79 | 674 | 291 | 704 | 97 | 203 |
| BL2 ($T_p$=0.8) | 690 | 91 | 202 | 818 | 284 | 89 | 204 | 818 | 216 | 534 | 123 | 549 | 210 | 583 | 60 | 552 | 276 | 669 | 92 | 189 |

S: service, F: food, P: price, Am: ambience, An: anecdotes.

Table 3.14: Accuracy and number of labeled reviews with different thresholds $T_p$ (laptop dataset).

(a) Accuracy of polarity classification

| (source) | General | | | Quality | | | Performance | | | Design | | | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | Q | P | D | G | P | D | G | Q | D | G | Q | P | |
| BL2 ($T_p$=0.5) | .804 | .809 | .757 | .731 | .850 | .765 | .773 | .862 | .788 | .756 | .831 | .803 | .794 |
| BL2 ($T_p$=0.7) | .814 | .781 | .761 | .725 | .784 | .690 | .794 | .853 | .761 | .684 | .275 | .476 | .700 |
| BL2 ($T_p$=0.8) | .775 | .755 | .686 | .396 | .489 | .416 | .791 | .845 | .769 | .684 | .275 | .476 | .613 |

(b) Number of labeled reviews

| (source) | General | | | Quality | | | Performance | | | Design | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | Q | P | D | G | P | D | G | Q | D | G | Q | P |
| BL2 ($T_p$=0.5) | 406 | 317 | 254 | 660 | 317 | 249 | 680 | 408 | 252 | 566 | 371 | 277 |
| BL2 ($T_p$=0.7) | 355 | 297 | 221 | 501 | 245 | 188 | 597 | 377 | 222 | 252 | 83 | 110 |
| BL2 ($T_p$=0.8) | 265 | 232 | 182 | 289 | 153 | 103 | 530 | 351 | 208 | 210 | 74 | 93 |

G: general, Q: quality, P: performance, D: design.

**Effect of $T_s$.** The parameter $T_s$ controls the minimum sentiment score required for a word to be extracted as a domain-specific sentiment word. We compared $T_s = 0$ and $T_s = 0.3$ on the restaurant dataset. The results are shown in Table 3.15). These results indicate that $T_s = 0$ generally performs better: both Our1 and Our3 achieve higher average accuracy when $T_s = 0$, and Our2 shows no overall difference but more domain pairs benefit from $T_s = 0$. This trend can be explained by the fact that a lower threshold allows more sentiment words to be extracted, increasing the candidate pool for MLM substitution and thereby enriching the generated training data. Consequently, $T_s = 0$ was used in all main experiments.

Table 3.15: Accuracy of polarity classification with different thresholds $T_s$ (restaurant dataset).

| (source) | Service | | | | Food | | | | Price | | | | Ambience | | | | Anecdotes | | | | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (target) | F | P | Am | An | S | P | Am | An | S | F | Am | An | S | F | P | An | S | F | P | Am | |
| Our1 ($T_s$=0) | .798 | .694 | .682 | .540 | .783 | .630 | .704 | .585 | .708 | .731 | .472 | .461 | .786 | .789 | .630 | .531 | .571 | .756 | .546 | .597 | .650 |
| Our1 ($T_s$=0.3) | .793 | .685 | .682 | .543 | .783 | .639 | .682 | .598 | .708 | .740 | .481 | .458 | .786 | .783 | .620 | .530 | .565 | .735 | .528 | .588 | .646 |
| Our2 ($T_s$=0) | .791 | .731 | .691 | .541 | .802 | .657 | .691 | .590 | .705 | .754 | .464 | .433 | .798 | .794 | .639 | .558 | .563 | .727 | .574 | .597 | .655 |
| Our2 ($T_s$=0.3) | .795 | .704 | .695 | .538 | .798 | .648 | .682 | .605 | .720 | .735 | .455 | .433 | .795 | .781 | .667 | .537 | .571 | .751 | .611 | .571 | .655 |
| Our3 ($T_s$=0) | .799 | .713 | .691 | .542 | .815 | .657 | .687 | .602 | .717 | .747 | .476 | .451 | .795 | .795 | .639 | .556 | .574 | .731 | .574 | .605 | .658 |
| Our3 ($T_s$=0.3) | .791 | .704 | .691 | .541 | .792 | .657 | .691 | .601 | .720 | .739 | .472 | .442 | .801 | .774 | .639 | .541 | .589 | .747 | .602 | .597 | .657 |

S: service, F: food, P: price, Am: ambience, An: anecdotes.

#### 3.4.3.4 Analysis of Generated Review Sentences by CARG

Examples of sentences generated by CARG are shown in Tables 3.16 and 3.17. In these tables, the "source" review sentence is transformed into a "target" review sentence of another domain. The underlined words indicate the replaced tokens. The label shown after the source domain sentence corresponds to the gold polarity label of the dataset. In CARG, the same polarity label as the original source sentence is assigned to the generated target-domain sentence; the label after the target sentence represents this automatically assigned polarity. If the assigned polarity is correct, the sentence is marked with ✓, while an incorrect polarity assignment is indicated with ✗.

Table 3.16 presents examples in which CARG successfully generated appropriate target-domain sentences. For instance, given a source review from the *service* domain ("The service was attentive and her suggestions of menu items was right on the mark."), sentiment words such as "good" and "great," which are commonly used in the *food* domain are replaced by the original sentiment words, while "service" is replaced by "food." Similarly, in the case of a *food* review sentence ("The wine list is extensive and impressive."), domain-specific words such as "wine" and "list" are replaced with "customer" and "service," and the sentiment adjectives "extensive" and "impressive" are substituted with "good" and "friendly," which are typical for the *service* domain. Another example shows that ambience-related words such as "decor," "hip," and "happening" are replaced with food-related words like "pizza," "salty," and "spicy."

However, not all generated sentences are appropriate. Some sentences are unnatural, irrelevant to the target domain, or assigned an incorrect polarity label. Such cases can potentially harm the performance of polarity classification. Table 3.17 presents such inappropriate examples. In the first example, the *food* review sentence "The dinner menu is diverse and top-notch as well." is transformed into a sentence by replacing "menu" with the service-related word "table." Although grammatically correct, the resulting sentence cannot be considered as a typical service-domain review. In the second example, the source review "All my co-workers were amazed at how small the dish was." from the service domain is labeled as negative due to the presence of "small." In the generated sentence, however, "small" and "dish" are replaced with the positive terms "good" and "service," yet the sentence still inherits the negative label from the source. This results in an incorrect polarity assignment.

Based on these observations, there is room for improving CARG:

1. When predicting candidate words to fill [MASK], the polarity of the word is not considered. If the replaced word has an opposite polarity to the original, the polarity of the entire sentence may change. Therefore,

polarity consistency should be incorporated into the word replacement process.

2. Sometimes, [MASK] is filled with words unrelated to the target domain, often due to insufficient extraction of domain-specific words in domains with fewer reviews. A mechanism to estimate the domain-likelihood of candidate words could address this issue.

3. Although not discussed in detail above, generated sentences often exhibit redundancy, with near-duplicate sentences or limited lexical diversity. This is likely caused by beam search pruning and the threshold $T_k$ limiting the number of candidate words. Improving the generation algorithm to allow more diverse substitutions may mitigate this problem.

Table 3.16: Examples of appropriate target-domain sentences generated by CARG.

| Domain | Review sentence |
|---|---|
| Service (Source) | The service was attentive and her suggestions of menu items was right on the mark. (positive) |
| Food (Target) | The food was good and her choice of menu items was great on the menu. (positive✓) |
| Food (Source) | The wine list is extensive and impressive. (positive) |
| Service (Target) | The customer service is good and friendly. (positive✓) |
| Ambience (Source) | The decor is really blah and not at all hip or happening. (negative) |
| Food (Target) | The pizza is really blah and not at all salty or spicy. (negative✓) |

## 3.5 Summary

In summary, the evaluation results demonstrated that the combination of automatic labeling and cross-aspect review generation can effectively adapt ASC models when unlabeled target-domain data is available. The results showed consistent improvements over baselines, especially in mitigating label imbalance with Focal Loss. However, this framework inherently assumes

Table 3.17: Examples of inappropriate target-domain sentences generated by CARG.

| Domain | Review sentence |
|---|---|
| Food (Source) | The dinner menu is diverse and top-notch as well. (positive✓) |
| Service (Target) | The dinner table is diverse and top-notch as well. (positive✓, irrelevant to domain) |
| Service (Source) | All my co-workers were amazed at how small the dish was. (negative✓) |
| Food (Target) | All my co-workers were amazed at how good the service was. (negative✗) |

that target-domain reviews can be collected in advance, which is not always feasible in practice. To address scenarios where even unlabeled target data is absent, the next chapter investigates a more challenging domain-agnostic setting.

# Chapter 4

# Aspect-Enhanced Prompting for Domain-Agnostic Adaptation

## 4.1 Problem Settings

In this chapter, we also address the problem of ASC. The focus of this study is on ASC under a Multi-Source Domain Adaptation setting, with the additional constraint that the target domain is entirely unseen. More concretely, the objective is to classify the sentiment of an aspect in a review text from a target domain where no data, neither labeled nor unlabeled is available during training. This setting, referred to as domain-agnostic adaptation, reflects practical scenarios in which a new domain suddenly emerges, and there is no opportunity to collect even unlabeled target data beforehand.

Formally, let $D_s = \{D_1, D_2, \ldots, D_n\}$ denote a set of datasets from multiple source domains. Each dataset $D_i$ consists of labeled samples $(x, a, y)$, where $x$ is a review text, $a$ is the corresponding aspect term, and $y \in \{\text{positive}, \text{negative}, \text{neutral}\}$ is the sentiment label toward aspect $a$. In contrast, let $D_{ut}$ denote the set of unseen target domain, which remains completely unavailable during training time.

The research problem is thus formulated as follows: given training data exclusively from multiple source domains $D_s$, the goal is to learn a model that can accurately predict the sentiment labels of aspects in $D_{ut}$, despite the absence of any target-domain supervision or adaptation signals.

## 4.2 Overview of the Proposed Approach

In our Aspect-Enhanced Prompting (AEP) framework, the Aspect Sentiment Classification (ASC) task is reformulated as a text generation problem. Instead of directly predicting a class label, the model generates a sentiment word guided by a prompt. To achieve this, the framework employs two complementary generative models, as illustrated in Figure 4.1.

The first model is the Prompt Generation Model, which receives a review text and a target aspect as input and generates a corresponding prompt. Here, the review is represented as the original review sentence, and the aspect is represented as its textual aspect name in natural language. These two inputs are concatenated into the prompt generation model to generate its corresponding prompt. A prompt is a natural language query posed to a text generation model, explicitly asking about the sentiment toward a given aspect. For example, a prompt may take the following form: What is the sentiment of [aspect] considering [ARFs] in the review?

Here, Aspect-Related Features (ARFs) are defined as words in a review that are closely related to a given aspect and provide strong contextual or sentimental cues for polarity classification. For instance, in the review "The staff ignored my friends and I the entire time we were there," the aspect is "staff" and the ARFs may include "ignored", "friends", and "entire". The Prompt Generation Model integrates such ARFs into the prompt, thereby highlighting aspect-relevant information.

The second model is the Sentiment Classification Model, which takes both the review text and the generated prompt as input, and produces a sentiment word such as positive, negative, or neutral. This design effectively converts ASC into a prompt-based generation task, allowing sentiment prediction to be guided by linguistically enriched prompts.

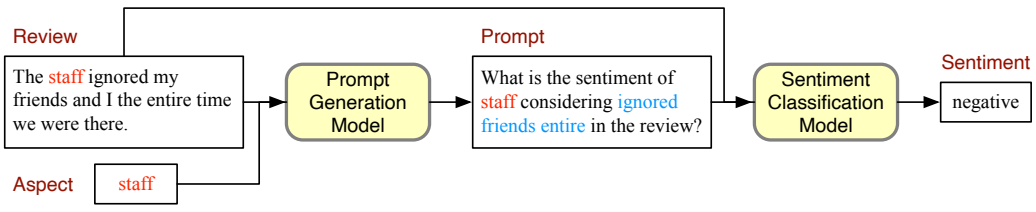


Figure 4.1: Overview of the proposed Aspect-Enhanced Prompting (AEP) framework. A review and an aspect in it are given to the Prompt Generation Model to generate a natural language prompt, which is then used by the Sentiment Classification Model to output the sentiment polarity.

The Prompt Generation Model plays a central role in this framework.

By learning how to automatically extract and integrate ARFs into prompts, it provides crucial signals for guiding sentiment classification. Importantly, since it is trained on samples from multiple source domains, the model can incorporate domain-independent linguistic patterns and generate prompts that remain informative even in unseen domains. This enhances robustness against domain shift. Furthermore, although text generation models are known to be sensitive to prompt design, our Prompt Generation Model is capable of automatically producing the most appropriate prompts for a given review, thereby improving generalization across domains.

## 4.3 Proposed Method

Figure 4.2 shows an overview of the training of two text generation models.

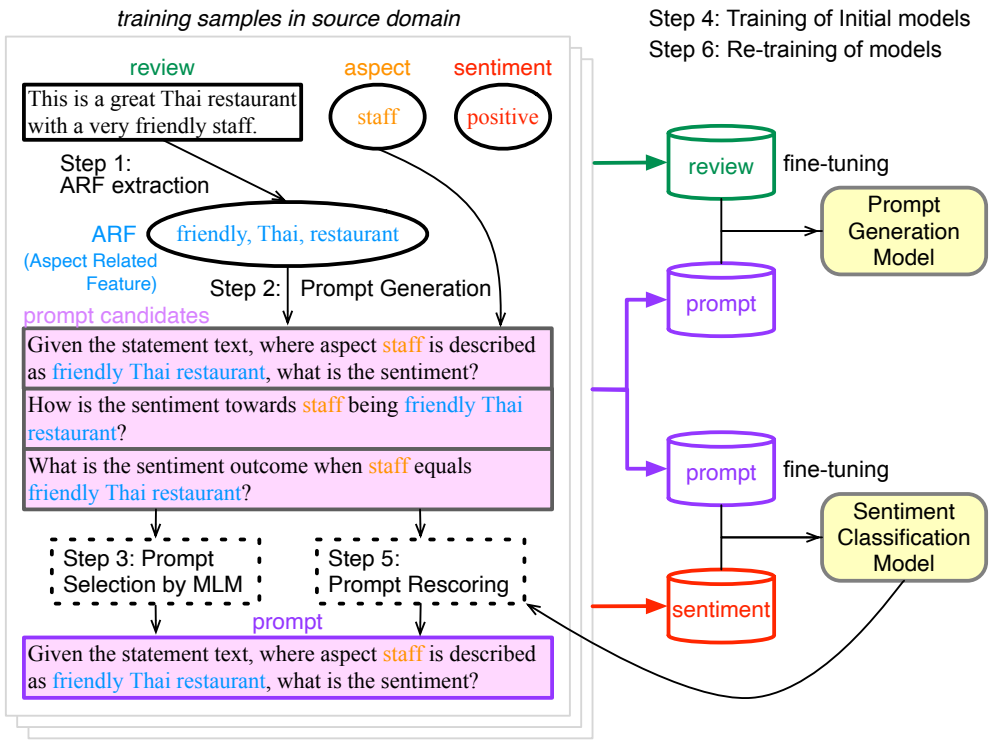


Figure 4.2: Overview of the training

In Step 1, several ARFs are extracted from each training sample in source domains. In Step 2, candidates for the prompts are produced by filling an aspect and ARFs into a template. In Step 3, scores of the prompts are calculated to choose the best prompt. In Step 4, the Prompt Generation Model is obtained by fine-tuning a pre-trained generative language model using pairs

of reviews and chosen prompts. Also, the Sentiment Classification Model is fine-tuned using reviews, prompts, and ground-truth sentiment labels. In Step 5, the prompts are updated by rescoring based on the initially trained Sentiment Classification Model. In Step 6, the prompt generation model and the sentiment classification model are re-trained by fine-tuning with the updated prompts.

The succeeding subsections will explain the details of these steps.

### 4.3.1 Extraction of Aspect-Related Features

A key component of the proposed framework is the extraction of ARFs. ARFs represent words in a review that carry a strong semantic of statistical association with the given aspect. Incorporating such features into prompts provides explicit hints that help generative models focus on the sentiment context relevant to the aspect under consideration.

The extraction procedure is carried out in three steps. First, review texts that contain the target aspect are tokenized. Second, stopwords, punctuation marks, and other non-informative tokens are removed to reduce noise. Finally, the remaining content words are treated as candidate features for further extraction. To quantify the degree of association between an aspect $a$ and a candidate feature $f$, we compute the Pointwise Mutual Information (PMI):

$$\mathrm{PMI}(a, f) = \log \frac{P(a, f)}{P(a)P(f)}$$

where $P(a, f)$ denotes the joint probability that the aspect $a$ and the feature $f$ co-occur, while $P(a)$ and $P(f)$ represent their marginal probabilities. Intuitively, PMI measures how frequently the two terms co-occur compared to what would be expected if they were independent. A higher PMI score indicates a stronger association between the aspect and the candidate feature.

Since domain-specific usage can vary, PMI is computed separately within each source domain. For each aspect in a review sentence, the top $N_{\mathrm{arf}}$ candidate words with the highest PMI values are selected as ARFs. This parameter $N_{\mathrm{arf}}$ determines the number of ARFs incorporated into subsequent prompt construction.

Examples of extracted ARFs from different domains are presented in Table 4.1. The results demonstrate that ARFs capture meaningful associations between aspects and their linguistic context. For instance, in the restaurant domain, the aspect *staff* is closely related to ARFs such as *friendly*, *Thai*, and *restaurant*. Similar patterns can be observed in other domains, confirming that appropriate ARFs are successfully identified across diverse settings.

Table 4.1: Extracted Aspect-Related Features from different domains.

| **Review** | **Domain** | **Extracted ARFs** |
|---|---|---|
| This is a great Thai restaurant with a very friendly staff. | Restaurant | friendly, Thai, restaurant |
| The battery was completely dead, in fact it had grown about a quarter inch thick lump on the underside. | Laptop | thick, grown, quarter |
| It is a perfect phone in such a small and appealing package. | Device | perfect, small, appealing |
| I have heard of people having problems receiving email out of order or not receiving some of their messages at all. | Service | heard, people, problems |
| Most are very expensive to rent or buy even the grotty little flats and bedsits in LOCATION1. | Location | expensive, rent, buy |

Note: $N_{arf}$ is set to 3. The aspects are underlined.

While the PMI-based extraction procedure is effective in identifying aspect-relevant words, it also has certain limitations. First, PMI relies purely on statistical co-occurrence and does not explicitly account for whether a feature word conveys sentiment. As a result, high-PMI words may sometimes include topical or descriptive terms that are semantically related to the aspect but sentiment-neutral (e.g., *restaurant* for the aspect *staff*). Second, the estimation of probabilities is sensitive to data sparsity, meaning that rare but sentimentally important words may not be selected as ARFs if their co-occurrence frequency is too low. Third, since PMI is calculated independently within each source domain, inconsistencies may arise when transferring features across domains with very different vocabularies.

These limitations suggest that ARFs extracted solely by statistical association may not always provide sufficient sentiment hints. Therefore, it becomes crucial to design a mechanism that can effectively utilize ARFs while remaining robust to noise and variability. In the proposed framework, this role is fulfilled by the Prompt Generation Model, which learns how to integrate ARFs into task-specific prompts in a way that enhances the generalization ability of sentiment classification across unseen domains.

### 4.3.2 Generation of Candidate Prompts

Once the ARFs have been extracted, the next step is to generate ARF-based prompts that serve as inputs to the generative sentiment classification framework. The purpose of prompt generation is to explicitly highlight the relationship between a target aspect and its associated ARFs, thereby guiding the sentiment classification process.

To achieve this, we manually designed a collection of natural-language prompt templates. In designing the prompt templates, several principles were considered to ensure that the generated prompts effectively capture the sentiment context. First, the prompts should maintain a natural and fluent sentence structure so that the model can interpret them as realistic language inputs. Second, both the aspect term and the extracted ARFs should appear in semantically meaningful positions within the sentence to preserve their syntactic relationship. Third, the templates should be diverse enough to prevent the model from overfitting. Finally, the templates should cover a wide range of syntactic and pragmatic forms.

Follow these principles, the manual designed templates fall into two categories: (1) question templates, which explicitly ask about the sentiment associated with a given aspect and its ARFs, and (2) description templates, which describe the aspect together with its ARFs in a declarative form and then request a sentiment prediction. This diversity in syntactic structure

exposes the model to multiple ways of expressing sentiment queries, reducing sensitivity to specific patterns. Here are examples of a templte of each category.

- **Question template:** What is the sentiment of [aspect] considering [ARFs] in the review?
- **Description template:** Predict the sentiment for [aspect] described as [ARFs].

In total, a set of 20 prompt templates was created, evenly distributed across the two categories. The complete list of templates is provided in Appendix A. Candidate prompts are obtained by filling a target aspect and its corresponding ARFs into these templates.

In addition to the manually designed templates used in the AEP framework, we also explored a supplementary approach to examine the diversity of possible templates. Specifically, we utilized ChatGPT [43] to automatically generate a larger set of templates that also contained placeholders for both aspect term and ARFs. These automatically generated templates by LLM were not directly employed in the AEP framework. Instead, we used them only as an additional comparison, to compare the performance of AEP with manually designed templates versus AEP with automatically generated templates.

### 4.3.3 Prompt Scoring

Since multiple candidate prompts are generated from different templates, it is necessary to determine which prompt is most suitable for guiding sentiment classification. To this end, we introduce a prompt scoring mechanism that evaluates each candidate and selects the best one. The scoring is carried out using the Masked Language Model (MLM) of BERT [8], which is able to estimate the probability of filling a masked token in a given sentence. In our case, the MLM estimates the likelihood that an extracted Aspect-Related Feature (ARF) can appropriately fill a masked position within a prompt template. Intuitively, a good prompt should provide a natural linguistic context in which the ARFs are highly probable according to the language model.

Formally, let $t_i$ denote the $i$-th prompt template that is filled with a given aspect, and let $f_j$ represent the $j$-th ARF associated with that aspect. The score of a candidate prompt $p_i$ is defined as the sum of MLM-estimated probabilities over all ARFs:

$$Score(p_i) = \sum_j \text{MLM}(p_i, f_j) = \sum_j P(f_j \mid t_i) = \sum_j \text{softmax}(\text{Logits}_{\text{masked\_position}(f_j)}) \tag{4.1}$$

where $t_i$ is the $i$th template filled with the aspect, $f_j$ is the $j$th ARF, and $MLM(t_i, f_j)$ is the MLM-estimated probability that $f_j$ is filled into $t_i$. Finally, the prompt $p_i$ with the highest score is selected as the best prompt for subsequent training and inference.

### 4.3.4 Training of Initial Models

After selecting the best prompt for each training instance, we construct prompt-augmented samples of the form $(x, a, y, p_i)$, where $x$ is a review, $a$ is the target aspect, $y$ is the ground-truth sentiment label, and $p_i$ is the selected prompt. These samples are then used to train two models: the Prompt Generation Model and the Sentiment Classification Model. Both models are built upon the Text-to-Text Transfer Transformer (T5) architecture [15], and in this study we employ the pre-trained T5-base model as the base model.

To represent the input for both models, we define the tokenized sequences as follows. For the Prompt Generation Model, the input is the concatenation of the review $x = \{w_1, w_2, \ldots, w_T\}$ and the aspect $a$:

$$\mathbf{X}_{\text{PG}} = [\texttt{<s>}, w_1, w_2, \ldots, w_T, \texttt{<sep>}, a, \texttt{</s>}], \tag{4.2}$$

where `<s>` and `</s>` denote the start and end tokens, and `<sep>` separates the review and aspect segments. For the Sentiment Classification Model, the input is the concatenation of the review $x$ and the generated prompt $p$:

$$\mathbf{X}_{\text{SC}} = [\texttt{<s>}, w_1, w_2, \ldots, w_T, \texttt{<sep>}, p, \texttt{</s>}]. \tag{4.3}$$

For the Prompt Generation Model, the goal is to generate an appropriate prompt given a review and its aspect. The input to the model is the concatenation of the review $x$ and aspect $a$, while the output is the target prompt $p_i$. The model is fine-tuned using the standard token-level cross-entropy loss:

$$\mathcal{L}_{\text{prompt}} = -\sum_{t=1}^{T} \log P(p_t \mid p_{<t}, [x; a]), \tag{4.4}$$

where $p_t$ is the $t$-th token of the ground-truth prompt, $p_{<t}$ represents all tokens generated before step $t$, and $P(\cdot \mid p_{<t}, [x; a])$ is the conditional probability distribution over the vocabulary produced by the model.

For the Sentiment Classification Model, the goal is to predict the sentiment label of an aspect based on the review and its corresponding prompt. The input is the concatenation of the review $x$ and prompt $p$, and the output is the sentiment label $y$. The model is fine-tuned using the cross-entropy loss defined as:

$$\mathcal{L}_{\text{sentiment}} = -\log P(y \mid [x; p]), \tag{4.5}$$

where $P(y \mid [x; p])$ denotes the probability that the model generates the correct label $y$ given the input pair.

By training these two models with prompt-augmented data, the framework learns not only to produce effective aspect-aware prompts but also to utilize them for robust sentiment classification, thereby improving the generalization ability in unseen domains.

### 4.3.5 Rescoring of the Prompts and Re-Training of the Models

As described in subsection 4.3.3, the initial selection of prompts relies on the MLM of BERT, which primarily evaluates the fluency of prompts. However, a prompt that has higher fluency does not necessarily ensure high effectiveness for sentiment classification. In other words, although MLM-based scoring provides a useful prior, it may fail to identify the prompt that best contributes to improving the performance of the downstream ASC task. To address this limitation, we introduce a further rescoring mechanism that directly evaluates the appropriateness of prompts using the performance of the initially trained Sentiment Classification Model.

In this mechanism, all candidate prompt templates are rescored according to the prediction probability produced by the Sentiment Classification Model. Specifically, given a review $x$, an aspect $a$, and the corresponding label $y$, the initially selected prompt $p_{best}$ is re-evaluated by computing the probability that $M_{sc}$ (the Sentiment Classification Model) correctly predicts $y$ when conditioned on $x$, $a$, and $p_{best}$. If this probability is sufficiently high, $p_{best}$ is retained. Otherwise, the algorithm selects a new prompt $p_{new}$ from the remaining candidates, choosing the one that maximizes the prediction probability of the correct label. This process ensures that the selected prompt is not only linguistically natural, but also empirically validated to improve classification performance.

Algorithm 2 presents the pseudocode of the rescoring mechanism. Here, $\theta$ is a hyperparameter that acts as a threshold: if the confidence of the

Sentiment Classification Model exceeds $\theta$, the initial prompt is accepted; otherwise, a replacement is sought among the remaining prompts.

**Algorithm 2** Prompt Rescoring Algorithm

1: **Input:** Training sample $(x, a, y)$, selected prompt $p_{best}$, set of all prompts $P$, initial Sentiment Classification Model $M_{sc}$
2: **Output:** Updated prompt $p_{new}$
3: $prob \leftarrow \text{Prediction\_Prob}(M_{sc}, x, a, p_{best}, y)$
4: **if** $prob > \theta$ **then**
5: $\quad p_{new} \leftarrow p_{best}$
6: **else**
7: $\quad P' \leftarrow P \setminus \{p_{best}\}$
8: $\quad p_{new} \leftarrow \arg\max_{p_i \in P'} \text{Prediction\_Prob}(M_{sc}, x, a, p_i, y)$
9: **end if**
10: **return** $p_{new}$

After rescoring is applied to all training samples, the dataset is updated with the newly assigned prompts. The Prompt Generation Model and the Sentiment Classification Model are then fine-tuned again using this updated dataset. This re-training procedure has two key benefits: (1) it reduces the reliance on fluency-based scoring and directly optimizes prompts for sentiment classification accuracy, and (2) it provides a feedback loop in which the Sentiment Classification Model helps refine the prompts, which in turn enhances its own performance. Through this refinement, the overall framework achieves more robust generalization in unseen domains.

### 4.3.6 Cluster-Based Prompt Expansion

While the prompt selection procedure described earlier helps to reduce prompt bias, relying on a single prompt during inference can still make the classification model sensitive to the specific wording of the prompt. This phenomenon, often referred to as *prompt bias*, occurs when a given prompt unintentionally favors a particular sentiment class, thereby reducing performance. To alleviate this issue, we propose a **cluster-based prompt expansion** mechanism, which is applied at the inference step to increase the diversity of prompts and improve the stability of sentiment classification.

Figure 4.3 illustrates an overview of the cluster-based prompt expansion process, and Algorithm 3 presents the pseudocode. Given a review $x$ and a target aspect $a$ from the unseen domain, the Prompt Generation Model $M_{pg}$ first generates an initial prompt $p_{orig}$ (Line 3). From this prompt, the

ARFs are extracted as a set $F$ (Line 4). Next, a collection of prompts is constructed by instantiating each template $t_i \in T$ with the aspect $a$ and ARFs $F$ (Lines 5–8).

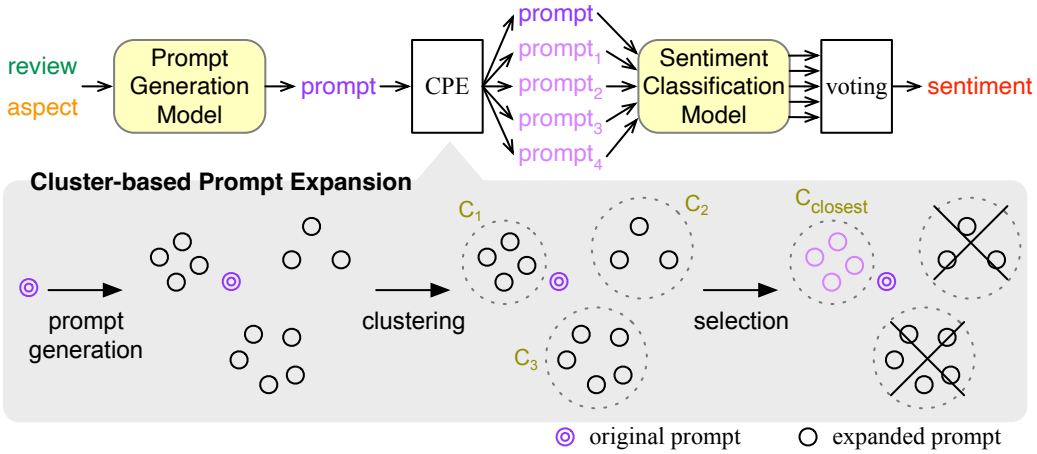


Figure 4.3: Overview of Cluster-Based Prompt Expansion

**Algorithm 3** Cluster-Based Prompt Expansion Algorithm

1: **Input:** Review $(x, a)$, set of templates $T$, Prompt Generation Model $M_{pg}$, Sentiment Classification Model $M_{sc}$
2: **Output:** Predicted sentiment label $\hat{y}$
3: $p_{orig} \leftarrow \text{Prompt\_Generation}(M_{pg}, x, a)$
4: $F \leftarrow \text{Extract\_ARF}(p_{orig})$
5: $P \leftarrow \emptyset$
6: **for** $t_i \in T$ **do**
7: $\quad p_i \leftarrow \text{Fill\_Template}(t_i, a, F)$ ; $P \leftarrow P \cup \{p_i\}$
8: **end for**
9: $C \leftarrow \text{Kmeans\_Clustering}(P, k)$
10: $C_{closest} \leftarrow \arg\max_{C_i \in C} \text{Similarity}(p_{orig}, C_i)$
11: $P_{all} \leftarrow \{p_{orig}\} \cup C_{closest}$
12: $\hat{y} \leftarrow \text{Voting}(M_{sc}, P_{all})$
13: **return** $\hat{y}$

To diversify prompts, we apply clustering on the constructed set $P$ (Line 9). We choose K-Means [39] as the major clustering method. Each prompt is first converted into a dense embedding using the pre-trained BERT model, and clustering is performed in this embedding space. The number of clusters

$k$ is set to 5 empirically, since the goal of clustering is not to obtain fine-grained semantic groups, but rather to group similar prompts before voting. Given that the total number of prompt templates is is $N = 20$, this setting results in each cluster containing approximately four prompt candidates. The cluster whose centroid is most similar to the embedding of $p_{orig}$ is selected as $C_{closest}$ (Line 10). Finally, the original prompt together with the prompts in $C_{closest}$ (forming the set $P_{all}$) are used for sentiment classification (Line 11).

The sentiment label is determined by aggregating the predictions of $M_{sc}$ across all prompts $p_i \in P_{all}$. We employ two aggregation mechanisms: *majority voting* and *weighted voting*.

In the majority voting scheme, the sentiment class with the highest frequency among the predictions is chosen:

$$\hat{y} = \arg\max_{y} \sum_{p_i \in P_{all}} \mathbf{1}\big(M_{sc}(x, a, p_i) = y\big), \tag{4.6}$$

where $\mathbf{1}(\cdot)$ is an indicator function that returns 1 if the predicted label equals $y$, and 0 otherwise.

In the weighted voting scheme, we treat the original prompt with higher importance so that it is assigned with higher weighs, while all supporting prompts in the cluster share equal lower weights:

$$\hat{y} = \arg\max_{y} \Big( w_{orig} \cdot \mathbf{1}(M_{sc}(x, a, p_{orig}) = y) + \sum_{p_i \in C_{closest}} w_{sup} \cdot \mathbf{1}(M_{sc}(x, a, p_i) = y) \Big), \tag{4.7}$$

where $w_{orig} = 1$ and $w_{sup} = \frac{1}{|C_{closest}|}$.

By expanding the set of prompts through clustering and voting, this method mitigates prompt bias and ensures that sentiment predictions are less sensitive to the choice of a single prompt. As a result, cluster-based prompt expansion provides a more robust and reliable inference mechanism for aspect-level sentiment classification in unseen domains.

## 4.4 Evaluation

### 4.4.1 Datasets

To evaluate the effectiveness and robustness of the proposed framework, we conduct experiments on five publicly available datasets from distinct domains: Restaurant (R), Laptop (La), Device (D), Service (S), and Location (Lo). Each dataset consists of review–aspect–sentiment triples, where the

sentiment label is annotated as positive, negative, or neutral. The use of multiple domains allows us to assess not only in-domain performance but also the generalization ability to unseen domains.

**Restaurant and Laptop:** The datasets from the SemEval-2014 ABSA task [49] are used for the restaurant and laptop domains. These benchmark datasets contain reviews from real users. Each review is annotated with fine-grained aspect terms and corresponding sentiment labels. They are widely adopted in ABSA research and provide a evaluation baseline.

**Device:** The dataset for the device domain is taken from Toprak et al. [62]. It includes user reviews of electronic devices such as smartphones, covering a wide variety of product features.

**Service:** The service domain dataset originates from Hu and Liu [23]. It contains user opinions on aspects related to service quality, such as staff responsiveness and customer satisfaction.

**Location:** For the location domain, we employ the Sentihood dataset [53], which focuses on location-based sentiment analysis. It consists of reviews that evaluate public perceptions of specific places or neighborhoods, making it distinct from the other datasets.

These five datasets were selected to represent a wide range of scenarios, covering both product and service-oriented contexts. While the Laptop and Device datasets may share some similarities due to their focus on consumer electronics, the other domains differ substantially in vocabulary and content. This diversity enables a comprehensive evaluation of the adaptability of the proposed framework under significant domain shifts.

The class distributions of the datasets are summarized in Table 4.2. It is worth noting that the Device and Location datasets contain only positive and negative labels, whereas the other three datasets include the neutral class. Accordingly, binary sentiment classification is performed for the Device and Location domains, and ternary classification is conducted for the remaining domains.

In the evaluation, we adopt a *leave-one-domain-out* strategy: one domain is designated as the unseen target domain, and the remaining four serve as multiple source domains. This setting directly reflects the research problem addressed in this dissertation, namely, how to generalize the model to perform well in an unseen target domain without any target-domain supervision.

Table 4.2: Statistics of the five datasets used in the evaluation.

| Domain | Positive | Negative | Neutral | Total |
|---|---|---|---|---|
| Restaurant | 795 | 287 | 217 | 1,299 |
| Laptop | 482 | 505 | 177 | 1,164 |
| Device | 589 | 375 | – | 964 |
| Service | 920 | 629 | 135 | 1,684 |
| Location | 954 | 480 | – | 1,434 |

### 4.4.2 Experimental Settings

The experimental settings, including the hyperparameters used throughout the training and evaluation process, are summarized in Table 4.3. For the extraction of ARFs, the number of extracted tokens per review, denoted as $N_{arf}$, is fixed at 3. As described in Section 4.3, both the Prompt Generation Model $M_{pg}$ and the Sentiment Classification Model $M_{sc}$ are implemented using the T5-base model. Since these models play different roles within the framework, they are fine-tuned with different hyperparameters. Specifically, $M_{pg}$ requires a relatively higher learning rate and a longer training schedule to effectively adapt to the prompt generation task, whereas $M_{sc}$ is optimized with a smaller learning rate and fewer epochs to prevent overfitting on the classification task. In addition, after the prompt re-scoring process, both models perform an additional fine-tuning step with a smaller learning rate to refine their performance.

Table 4.3: Hyperparameter settings. $M_{pg}$ and $M_{sc}$ denote the Prompt Generation Model and the Sentiment Classification Model, respectively.

| Process | Parameter | Value |
|---|---|---|
| Extraction of ARFs | $N_{arf}$ | 3 |
| Fine-tuning of $M_{pg}$ | Learning Rate | $1 \times 10^{-4}$ |
| | Epochs | 10 |
| Fine-tuning of $M_{sc}$ | Learning Rate | $3 \times 10^{-5}$ |
| | Epochs | 2 |
| Additional fine-tuning of $M_{pg}$ & $M_{sc}$ | Learning Rate | $3 \times 10^{-6}$ |
| | Epochs | 1 |

For the ASC task, we adopt two widely used evaluation metrics: accuracy and the macro-averaged F1-score. The macro-averaged F1-score is computed over either two sentiment classes (positive and negative) when the target domain lacks neutral labels (e.g., Device and Location), or over three sentiment classes (positive, negative, and neutral) for the other domains.

### 4.4.3 Models for Comparison

To evaluate the effectiveness of the proposed method, we compared it against several baseline and state-of-the-art models. The compared methods can be categorized into three groups: (1) basic fine-tuning baselines, (2) prompt-based learning methods from prior work, and (3) state-of-the-art domain adaptation methods. Finally, our proposed AEP framework with additional modules is included for comparison.

**(1) Basic fine-tuning baseline**

- **T5-base**: A standard T5-base model fine-tuned on the training data from multiple source domains. Unlike our framework, this model does not employ any domain adaptation techniques. Instead, it was directly finetuned with the review and the corresponding aspect, and then generates a sentiment word as the classification output.

**(2) Prompt-based learning approaches**

- **AutoPrompt (AP)** [56]: A prompt-based method that automatically constructs discrete prompts through gradient-guided search. In this experiment, we applied AP to the multi-source unseen domain adaptation setting for ASC. Two pre-trained language models, BERT and RoBERTa (the latter being used in the original paper), are employed.

- **LM-BFF** [13]: A prompt-based fine-tuning approach designed to improve model performance in few-shot scenarios by leveraging automatically generated prompts and demonstrations. Although originally developed for few-shot learning, we extended LM-BFF to the multi-source unseen domain adaptation setting by utilizing the entire training dataset. RoBERTa is used as the pre-trained model.

**(3) Domain adaptation method**

- **PADA** [2]: A state-of-the-art model for multi-source unseen domain adaptation, which adapts knowledge from source domains to target domains by jointly capturing domain-specific and domain-invariant features. For a fair comparison, we reimplemented PADA and applied it to our dataset under the same experimental conditions.

**(4) Our proposed method**

- **AEP+RS+CE**: Our proposed Aspect-Enhanced Prompting (AEP) method, which integrates aspect-related features into prompt construction. This model reported here further incorporates two modules: prompt rescoring (**+RS**) and cluster-based prompt expansion (**+CE**), designed to improve robustness and generalization across domains.

## 4.5 Results

Table 4.4 shows the aspect sentiment classification performance of six models in terms of accuracy and macro F1-score across five domain adaptation scenarios, where one of the five domains is selected as the unseen target domain. The column "Average" reports the macro average over these five cases. An asterisk (*) indicates that the difference between the baseline and our proposed model is statistically significant ($p < 0.05$).

When compared with T5-base, our AEP+RS+CE model achieves clear improvements across all five target domains. On average, the accuracy of AEP+RS+CE is 0.832, which is 0.034 points higher than T5-base. The largest improvements are observed in the Location domain (+0.066) and the Device domain (+0.047), where the domain discrepancy is relatively large. A similar tendency is observed for the macro F1-score. AEP+RS+CE obtains the highest average F1-score of 0.697, outperforming T5-base by 0.036 points. These results clearly demonstrate that our Aspect-Enhanced Prompting framework provides statistically significant gains over T5-base, and that its advantage is consistent across all target domains.

Compared to AutoPrompt (BERT), AEP+RS+CE consistently achieves superior accuracy and F1-score across all target domains, with significant accuracy improvements observed in the Service (+0.043) and Location (+0.057) domains. On average, AEP+RS+CE demonstrates higher accuracy and macro F1-score, highlighting its robustness over AutoPrompt (BERT). When compared to AutoPrompt (RoBERTa), AEP+RS+CE maintains higher accuracy and F1-score in most target domains. However, in the Restaurant and Laptop domains, AutoPrompt (RoBERTa) performs better. Additionally, AutoPrompt (RoBERTa) consistently surpasses AutoPrompt (BERT)

Table 4.4: Performance of Aspect Sentiment Classification

(a) accuracy

| Model | $\mathcal{MS}$→R | $\mathcal{MS}$→La | $\mathcal{MS}$→D | $\mathcal{MS}$→S | $\mathcal{MS}$→Lo | Average |
|---|---|---|---|---|---|---|
| T5-base | 0.746* | 0.754* | 0.882* | 0.798* | 0.809* | 0.798* |
| AP(BERT) | 0.743* | 0.758* | 0.893* | 0.793* | 0.818* | 0.801* |
| AP(RoBERTa) | 0.767* | 0.770 | 0.901* | 0.819* | 0.857* | 0.823* |
| LM-BFF | 0.752 | 0.764 | 0.891* | 0.802* | 0.806* | 0.803* |
| PADA | **0.756** | **0.780*** | 0.907* | 0.832 | 0.815* | 0.818* |
| AEP+RS+CE | 0.754 | 0.769 | **0.928** | **0.835** | **0.875** | **0.832** |

(b) macro F1-score

| Model | $\mathcal{MS}$→R | $\mathcal{MS}$→La | $\mathcal{MS}$→D | $\mathcal{MS}$→S | $\mathcal{MS}$→Lo | Average |
|---|---|---|---|---|---|---|
| T5-base | 0.522 | 0.545 | 0.892 | 0.550 | 0.796 | 0.661 |
| AP(BERT) | 0.519 | 0.552 | 0.895 | 0.546 | 0.807 | 0.664 |
| AP(RoBERTa) | **0.551** | 0.582 | 0.897 | 0.561 | 0.842 | 0.687 |
| LM-BFF | 0.539 | 0.558 | 0.893 | 0.552 | 0.796 | 0.668 |
| PADA | 0.544 | 0.575 | 0.908 | **0.577** | 0.805 | 0.682 |
| AEP+RS+CE | 0.527 | **0.596** | **0.925** | 0.576 | **0.862** | **0.697** |

Note: $\mathcal{MS}$ stands for "multi-source domains". Bold indicates the best among all models.

across all target domains, likely due to its use of RoBERTa-large, a model with substantially more parameters and richer pre-trained representations than the base versions of BERT. Overall, while AutoPrompt (RoBERTa) achieves competitive results, our model AEP+RS+CE achieves better performance on average.

Compared to LM-BFF (RoBERTa), AEP+RS+CE achieves higher accuracy across all target domains. In terms of macro F1-score, it outperforms LM-BFF in all domains except Restaurant. Notably, the performance of LM-BFF (RoBERTa) is comparable to that of T5-base, which suggests that its advantage is limited in a full-data training setting. This may be due to LM-BFF's design, which is specifically optimized for few-shot learning. When applied to a full training dataset, traditional fine-tuning methods like T5-base can achieve similar results. Overall, this indicates that AEP+RS+CE provides robust performance across diverse domains.

In comparison to the PADA, the AEP+RS+CE does not consistently demonstrate superior performance. When the target domain is Restaurant or Laptop, i.e., the datasets of the SemEval 2014 ABSA task, the PADA performs better than our method. Nevertheless, for three out of the five domains, the accuracy of the AEP+RS+CE is better than the PADA. In terms of the F1-score, our method is less effective than the PADA for the Restaurant domain, but better or comparable for the other domains. On average across the five domains, the AEP+RS+CE outperforms the PADA in terms of both the accuracy and F1-score. These results indicate that the AEP has an excellent ability to adapt the sentiment classification model to different types of domains. The mechanism for automatically generating the prompts including ARFs could contribute to preventing the model from performing worse on an unseen target domain.

## 4.6 Detailed Evaluation of the Components

Several additional experiments have been conducted to evaluate the contribution of the components in the AEP model. As the results for accuracy and macro-weighted F1-score are nearly the same, only the macro-weighted F1-score will be presented in this subsection.

### 4.6.1 Ablation Study

To observe the contribution of each component in the proposed AEP framework, we conducted an ablation study by selectively removing the prompt rescoring (RS) and cluster-based prompt expansion (CE) modules. The re-

sults are summarized in Table 4.5. In the table, the symbol × in the columns "RS" and "CE" indicates that the corresponding module was removed. The column "Δ" represents the difference in macro F1-score compared with the full AEP+RS+CE model.

Table 4.5: Ablation study of the AEP model.

| Model | RS | CE | $\mathcal{MS}$→R | $\mathcal{MS}$→La | $\mathcal{MS}$→D | $\mathcal{MS}$→S | $\mathcal{MS}$→Lo | Average |
|---|---|---|---|---|---|---|---|---|
| AEP+RS+CE | | | 0.527 | 0.596 | 0.925 | 0.576 | 0.862 | 0.697 |
| AEP+CE | × | | 0.526 | 0.596 | 0.921 | 0.575 | 0.853 | 0.694 |
| (Δ) | | | (−0.001) | (0.000) | (−0.004) | (−0.001) | (−0.009) | (−0.003) |
| AEP+RS | | × | 0.525 | 0.583 | 0.923 | 0.572 | 0.857 | 0.692 |
| (Δ) | | | (−0.002) | (−0.013) | (−0.002) | (−0.004) | (−0.005) | (−0.005) |
| AEP | × | × | 0.521 | 0.549 | 0.916 | 0.572 | 0.846 | 0.681 |
| (Δ) | | | (−0.006) | (−0.047) | (−0.009) | (−0.004) | (−0.016) | (−0.016) |
| T5-base | × | × | 0.522 | 0.545 | 0.892 | 0.550 | 0.796 | 0.661 |

When the prompt rescoring module is removed, the average macro F1-score decreases slightly to 0.694. The most notable decline is observed in the Location domain, where the F1-score drops from 0.862 to 0.853 (−0.009). Although the Laptop domain shows no decline, the other drops suggest that the prompt rescoring module plays an important role in selecting more suitable prompts based on the initial Sentiment Classification Model, thereby improving performance in domains where domain discrepancy is more severe.

The removal of the cluster-based prompt expansion module results in a further decrease of the average macro F1-score to 0.692. While the expansion generally improves classification performance, its effect varies across domains. For example, the Laptop domain shows a noticeable reduction in performance without expansion, whereas the Restaurant and Device domains are less affected. This indicates that the benefit of prompt expansion is domain-dependent, but overall contributes to the robustness of the AEP framework.

A comparison between the two modules reveals that the cluster-based prompt expansion contributes more substantially than prompt rescoring. Specifically, removing CE leads to a larger drop in average performance (0.692) compared with removing RS (0.694). Moreover, when both modules are removed simultaneously, the macro F1-score declines more considerably to 0.681, with the most pronounced degradation occurring in the Laptop domain (−0.047). Importantly, the performance difference between the full model (AEP+RS+CE) and the variant without both modules (AEP) ex-

ceeds the sum of the individual improvements, suggesting that RS and CE complement each other in enhancing the overall robustness of sentiment classification.

We also compared AEP with the non-AEP baseline T5-base, which does not incorporate aspect-enhanced prompting. The results demonstrate that incorporating AEP consistently improves the performance of ABSA across most domains, with an average gain of 0.020 macro F1-score points. This further highlights the effectiveness of explicitly modeling Aspect-Related Features through our framework.

Finally, we analyzed the computational cost introduced by the cluster-based prompt expansion. The additional processing time per sample was measured on an NVIDIA RTX A6000 GPU server, with an observed range from 0.014 seconds in the Service domain to 0.044 seconds in the Device domain. These results suggest that the expansion module imposes only a moderate computational burden, which is well justified by the consistent improvements in performance.

### 4.6.2 Impact of Parameters on Prompt Rescoring

As described in subsection 4.3.5, the proposed rescoring module refines the initially generated prompts by filtering out those with low sentiment prediction probabilities. This mechanism is controlled by a threshold parameter $\theta$, which determines whether a prompt should be retained or replaced by an alternative with higher confidence. To observe the effect of this parameter, we conducted experiments with different values of $\theta$. In order to isolate the contribution of the rescoring module, the cluster-based prompt expansion was disabled during this evaluation. The results are presented in Table 4.6.

The experimental results show that incorporating prompt rescoring with different values of $\theta$ consistently improves the performance over the base AEP model without rescoring, with the only exception being the Service domain where the improvement is marginal. This observation indicates that the prompt rescoring module is relatively robust and not highly sensitive to the exact choice of the threshold across most domains. In other words, as long as rescoring is applied, the model benefits from the filtering mechanism, which enhances the reliability of the selected templates.

Among the tested settings, the model with $\theta = 0.98$ achieves the best performance in four out of the five domains (Restaurant, Laptop, Device, and Location), as well as in the overall average macro F1-score. This suggests that a moderate threshold reaches an effective balance between the two scoring criteria employed in the framework: (i) the initial score of probability assigned by the masked language model when filling templates with

Table 4.6: F1-score of Models with Different Parameters for Rescoring

| Model | $\mathcal{MS}$→R | $\mathcal{MS}$→La | $\mathcal{MS}$→D | $\mathcal{MS}$→S | $\mathcal{MS}$→Lo | Ave. |
|---|---|---|---|---|---|---|
| AEP | 0.521 | 0.549 | 0.916 | 0.572 | 0.846 | 0.681 |
| AEP+RS($\theta$=0.99) | 0.522 | 0.581 | 0.920 | **0.574** | 0.853 | 0.690 |
| AEP+RS($\theta$=0.98) | **0.525** | **0.583** | **0.923** | 0.572 | **0.857** | **0.692** |
| AEP+RS($\theta$=0.95) | 0.524 | 0.576 | 0.920 | **0.574** | 0.856 | 0.690 |

ARFs, and (ii) the reliability of the sentiment classification model in the prompt rescoring module. A very high threshold (e.g., $\theta = 0.99$) places excessive emphasis on the sentiment classification model's confidence and may filter out potentially useful prompts, while a relatively low threshold (e.g., $\theta = 0.95$) increases reliance on initial score by the MLM, but risks retaining prompts of lower quality for sentiment classification. The intermediate setting of $\theta = 0.98$ effectively integrates both perspectives, thereby achieving better classification performance across domains.

These results highlight the importance of tuning the threshold parameter to balance the contributions of the generative and discriminative components in the AEP framework. They also suggest that the prompt rescoring module provides a stable and generalizable enhancement to prompt quality, as improvements are observed in nearly all domains regardless of the specific value of $\theta$. This robustness makes the rescoring strategy particularly suitable for domain adaptation scenarios, where the quality of prompts can vary significantly depending on the target domain.

### 4.6.3 Investigation of Voting Strategy in Cluster-based Prompt Expansion

In the cluster-based prompt expansion module, multiple prompts are aggregated to determine the final sentiment prediction. We investigated two aggregation strategies: majority voting (ma) and weighted voting (we). The weighted voting scheme incorporates confidence scores according to Equation (4.7), thereby assigning greater importance to more reliable predictions. Table 4.7 presents the results of models using these two strategies.

Here, "AEP+CE" refers to the model without prompt rescoring module, while "AEP+RS+CE(we)" corresponds to the full model reported in Tables 4.4 and 4.5.

The results show that weighted voting strategy generally achieved better performance than simple majority voting. For instance, in the case of AEP+RS+CE, the weighted voting strategy achieves superior results in the

Table 4.7: Macro F1 of Models with Two Voting Methods in Cluster-based Prompt Expansion

| Model | $\mathcal{MS}$→R | $\mathcal{MS}$→La | $\mathcal{MS}$→D | $\mathcal{MS}$→S | $\mathcal{MS}$→Lo | Ave. |
|---|---|---|---|---|---|---|
| AEP+CE(ma) | 0.523 | **0.603** | 0.910 | 0.573 | **0.853** | 0.692 |
| AEP+CE(we) | **0.526** | 0.596 | **0.921** | **0.575** | **0.853** | **0.694** |
| AEP+RS+CE(ma) | **0.528** | **0.597** | 0.918 | 0.571 | **0.863** | 0.695 |
| AEP+RS+CE(we) | 0.527 | 0.596 | **0.925** | **0.576** | 0.862 | **0.697** |

Device and Service domains, as well as in the overall average macro F1-score. This improvement highlights the effectiveness of incorporating prediction confidence as weights when aggregating the outputs of multiple prompts, ensuring that stronger predictions have a greater influence on the final decision.

When prompt rescoring is disabled, weighted voting also outperforms majority voting in three out of five domains and in the overall average score. Although the margin of improvement is relatively modest, the nearly consistent advantage suggests that weighted voting provides a more stable aggregation mechanism compared with majority voting. In particular, by leveraging confidence scores, the weighted strategy helps mitigate the negative influence of low-quality prompts, which can occasionally dominate in simple majority voting.

In addition, across both aggregation strategies, models with prompt rescoring consistently outperform their counterparts without rescoring. This observation further validates the complementary effect of the prompt rescoring module: by filtering out less reliable prompts before aggregation, rescoring provides a cleaner set of candidate predictions for the voting mechanism to operate on. Consequently, the combination of rescoring and weighted voting achieves the best overall results, as reflected in the superior performance of AEP+RS+CE(we).

### 4.6.4 Investigation of Input Format of the Sentiment Classification Model

In the proposed AEP framework, we further observe how the input format to the Sentiment Classification Model influences performance. Specifically, we compare two alternative strategies for combining the review and the generated prompt.

The first strategy, referred to as AEP-Separate, concatenates the review and the generated prompt as two separate segments. The second strategy,

denoted as AEP-Insert, integrates the review directly into the template of the prompt, producing a single extended sequence that contains the review, the aspect, and the ARFs in a unified input. An illustrative example of these two formats is provided in Table 4.8.

Table 4.8: Two Input Formats of Sentiment Classification Model. The review text is underlined.

| **Method** | **Example** |
|---|---|
| AEP-Separate | *This is a great Thai restaurant with a very friendly staff.* With *staff* being *great Thai restaurant*, how is the sentiment? |
| AEP-Insert | Consider the text: '*This is a great Thai restaurant with a very friendly staff.*' what sentiment does *great Thai restaurant* convey about *staff*? |

Table 4.9 shows the performance of the two input strategies. In this experiment, neither the prompt rescoring module nor the cluster-based prompt expansion module was applied, in order to isolate the effect of the input format. The results demonstrate that AEP-Separate outperforms AEP-Insert in all domains except the Restaurant domain, and that the difference in the average macro F1-scores between the two methods is substantial.

Table 4.9: F1-scores of AEP-Separate and AEP-Insert.

| Model | $\mathcal{MS}$→R | $\mathcal{MS}$→La | $\mathcal{MS}$→D | $\mathcal{MS}$→S | $\mathcal{MS}$→Lo | Ave. |
|---|---|---|---|---|---|---|
| AEP-Separate | 0.521 | **0.549** | **0.916** | **0.572** | **0.846** | **0.681** |
| AEP-Insert | **0.525** | 0.548 | 0.910 | 0.520 | 0.806 | 0.662 |

The inferior performance of AEP-Insert can be attributed to the difficulty of resolving sentiment polarity from a complex single-sentence input, in which the review, aspect, and ARFs are simultaneously embedded. Such an integrated structure increases the cognitive load for the model, making it harder to disentangle the sentiment-related information. In contrast, AEP-Separate maintains a clear division between the review text and the guiding prompt, allowing the model to process these two components more distinctly. This separation facilitates a more accurate mapping between the aspect and its associated ARFs, leading to consistently better classification performance.

In summary, these results indicate that a separated input structure is preferable for the Sentiment Classification Model within the proposed framework. They also suggest that explicitly distinguishing between raw review

content and guiding prompts can enhance the interpretability and robustness of aspect sentiment classification, particularly in cross-domain adaptation scenarios where input distributions may vary significantly.

### 4.6.5 Impact of Prompt Templates

To observe the sensitivity of the proposed framework to different prompt templates, we conducted a series of experiments by varying both the number and the type of templates used in the AEP framework. Specifically, we compared the following variants: (i) AEP (20), which employs the original 20 manually designed templates, (ii) AEP-Q-only, which uses only the 11 question-style templates, (iii) AEP-D-only, which employs the 9 description-style templates, (iv) AEP (50), which utilizes 50 templates automatically generated by a large language model (LLM), and (v) AEP (20+50), which combines both the original 20 handcrafted templates and the additional 50 LLM-generated templates. The 50 templates were generated by the ChatGPT with the following instruction:

> Generate diverse templates that describe an opinion about a given aspect. Each template should include the placeholders [ASPECT] and [MASK], where [ASPECT] denotes the target aspect and [MASK] represents a sentiment expression.

A complete list of the generated templates is provided in Appendix Table A.3. Figure 4.4 and Table 4.10 summarize the comparison results.

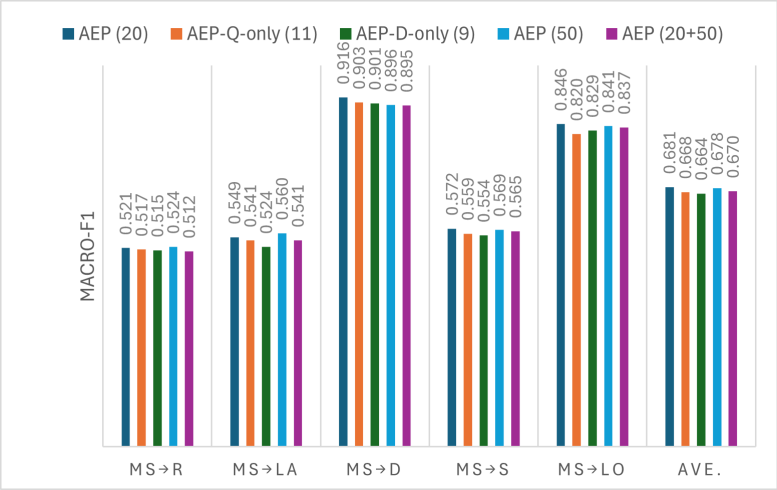


Figure 4.4: Comparison of different sets of prompt templates. The number of used templates is shown in parentheses.

Table 4.10: Macro F1-scores of AEP with different prompt template sets.

| Model | $\mathcal{MS}$→R | $\mathcal{MS}$→La | $\mathcal{MS}$→D | $\mathcal{MS}$→S | $\mathcal{MS}$→Lo | Average |
|---|---|---|---|---|---|---|
| AEP (20) | 0.521 | 0.549 | **0.916** | **0.572** | **0.846** | **0.681** |
| AEP-Q-only (11) | 0.517 | 0.541 | 0.903 | 0.559 | 0.820 | 0.668 |
| AEP-D-only (9) | 0.515 | 0.524 | 0.901 | 0.554 | 0.829 | 0.664 |
| AEP (50) | **0.524** | **0.560** | 0.896 | 0.569 | 0.841 | 0.678 |
| AEP (20+50) | 0.512 | 0.541 | 0.895 | 0.565 | 0.837 | 0.670 |

The results indicate several trends. First, reducing the number of templates consistently leads to performance drop. Both AEP-Q-only and AEP-D-only models achieve lower macro F1-scores compared with AEP (20), confirming that template diversity plays an important role in enhancing robustness. By providing varied linguistic contexts, a larger set of templates enables the model to generalize better across unseen domains.

Second, although AEP (50) introduces more templates generated by an LLM, its performance is not superior to the original AEP (20). While the LLM-generated templates provide stylistic and lexical diversity, they also introduce redundancy and inconsistency. For example, some automatically generated templates are extremely short (only 3–5 words), while others differ only slightly in form, such as: "The [aspect] is [arfs]." vs. "The [aspect] tends to be [arfs].", or "What do you think about the [aspect] on [arfs]?" vs. "How do you feel about the [aspect] on [arfs]?". Such minor variations increase the number of templates without adding substantial semantic diversity. As a result, although the model is exposed to a wider range of expressions, its decision boundary becomes broader and less discriminative, leading to a slight decrease in average performance compared with AEP (20).

Third, the combination model AEP (20+50) also fails to outperform AEP (20), and in fact shows a further decrease in average macro F1-score. This can be explained by the uneven distribution of template usage during prompt generation. In practice, the generated prompts for a target domain tend to rely more heavily on certain favored templates, even when more candidates are available. Therefore, while training with a larger set of 70 templates allows the model to observe a wide range of linguistic styles in the source domains, this diversity does not necessarily translate into improved robustness in the target domains.

In summary, the results demonstrate that template diversity is indeed beneficial, but that an excessive number of templates, especially those generated automatically without careful curation, may harm performance. The original manually designed set of 20 templates remains the most effective,

achieving a balance between linguistic and semantic variaty. In the future work, a promising research direction is to explore dynamic template generation using pre-trained language models such as T5 [51] and GPT [4], while incorporating mechanisms to filter redundant or low-quality templates. Such adaptive strategies could overcome the limitations of static, manually crafted templates and further improve cross-domain generalization in ABSA.

### 4.6.6 Error Analysis

To gain deeper insight into the weaknesses of the proposed model, we conducted an error analysis focusing on the Restaurant domain as the unseen target domain, where the performance of our full model (AEP+RS+CE) is worse than PADA in both accuracy and macro-F1. We manually investigate representative misclassified cases in this domain to better understand the sources of error.

Table 4.11 provides several examples of misclassifications. In the first example, the gold labels are neutral for the aspect *food* and negative for the aspect *ambience*. However, since the negative word "annoying" was extracted as an ARF and injected into the prompt for the aspect *food*, the model incorrectly predicted the sentiment of food as negative. This illustrates how sentiment-related but aspect-irrelevant tokens may mislead the sentiment classification.

In the second example, the positive ARF "friendly" is semantically associated with the overall tone of the review but not with the specific target aspect *service*. As a result, the presence of this misleading ARF caused the model to predict an incorrect positive label, despite the ground-truth label being negative due to the "slow" service.

The third example shows a failure case where irrelevant ARFs ("much," "bring," and "back") were extracted. These words carry little sentiment information in relation to the target aspect *food*, yet their inclusion led to an incorrect negative prediction for a positive instance. Such cases demonstrate that the primary limitation of AEP lies in the extraction of irrelevant or noisy ARFs, which may carry unintended sentiment features and bias the final prediction.

Although PADA avoids such errors by relying on domain-general features in these examples, it also frequently fails to capture fine-grained aspect-level sentiment. For example, in the second case of Table 4.11, PADA correctly avoided the spurious positive influence of the word "friendly" but nevertheless failed to capture the overall negative sentiment tied to service. Thus, while AEP suffers from aspect-specific ARF noise, PADA suffers from the other limitation, that is its reliance on coarse-grained features makes it less

sensitive to fine-grained aspect-level features. This suggests that an ideal framework may require a more refined balance between aspect-specific and domain-general representations. For example, in addition to extracting explicit ARFs such as sentiment-bearing adjectives directly linked to the aspect term (e.g., "tasty" for *food*), the framework could also incorporate implicit features that capture underlying sentiment cues not overtly expressed in the surface text. For instance, the sentence "the soup came cold" implicitly conveys a negative opinion about food, even though no explicit negative sentiment word is present. By jointly leveraging both explicit and implicit features, the model would be better positioned to filter out irrelevant tokens while preserving aspect-relevant sentiment information.

Table 4.11: Misclassified examples in the Restaurant domain.

| **Input Text (aspect, gold label)** | **Prompt** | | **Prediction** | |
|---|---|---|---|---|
| | **AEP** | **PADA** | **AEP** | **PADA** |
| The food is decent at best, and the ambience, well, it's a matter of opinion, some may consider it to be a sweet thing, I thought it was just annoying. (*food, neutral*) | With food being decent ambience annoying, how is the sentiment? | food egroups overall quickly table | neg | neu |
| The service was a bit slow, but they were very friendly. (*service, negative*) | Predict the sentiment for service described as bit slow friendly. | service slow egroups dinner toshiba | pos | neg |
| As much as I like the food there, I can't bring myself to go back. (*food, positive*) | With food being much bring back, how is the sentiment? | food egroups week extremely simple | neg | neg |

Figure 4.5 further quantifies these errors by showing the number of instances that PADA classified correctly but AEP misclassified, broken down by sentiment class. While there is no neutral sample in the Device and Location datasets, no neutral sample that only our model misclassified is found in the Service domain.

While the PADA model generates general domain-specific features in a prompt, which are not always extracted from a review sentence, ARFs are usually extracted from a review sentence. By inappropriately extracting

sentiment words as ARFs, our model incorrectly predicts a neutral sample as positive or negative. In addition, the poor performance in the Restaurant domain can be attributed to the diverse and descriptive nature of reviews about restaurants, where aspects with neutral and non-neutral sentiments can appear in a sentence.

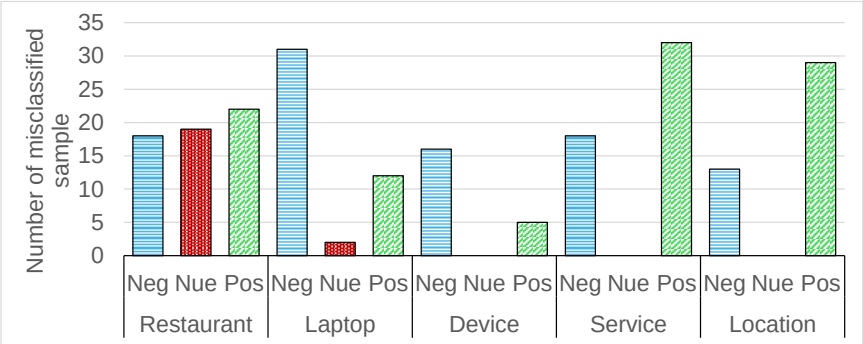


Figure 4.5: Number of misclassifications by AEP compared to PADA, broken down by sentiment class.

## 4.7 Summary

In summary, to handle Scenario 2, we proposed the Aspect-Enhanced Prompting framework, which enables ASC models to be generalized to unseen domains without any target-domain supervision. Through ARF-based prompt construction, rescoring, and cluster-based expansion, the method consistently outperformed strong baselines across multiple domains, demonstrating its robustness in domain-agnostic adaptation. Nevertheless, its performance depends on the quality of ARFs, and neutral classes remain challenging.

# Chapter 5

# Conclusion

## 5.1 Summary of this dissertation

This dissertation systematically investigated the domain shift problem in Aspect Sentiment Classification under two complementary settings with different levels of target-domain availability.

In scenario A, where unlabeled target-domain data is assumed to be available, we proposed a new unsupervised domain adaptation framework. The method integrated two complementary strategies: (i) automatic labeling of unlabeled reviews using a source-trained classifier with confidence-based filtering, and (ii) cross-aspect review generation that generates target domain reviews through sentiment word substitution using masked language modelling. To mitigate class imbalance, the Focal Loss was further applied during classifier training. Experiments on the restaurant and laptop datasets demonstrated that the proposed framework consistently outperformed baselines, confirming the effectiveness of leveraging unlabeled target data for cross-aspect adaptation.

In scenario B, where the target domain is completely unknown, we introduced the Aspect-Enhanced Prompting framework. AEP formulated ASC as a text generation task and integrated Aspect-Related Features into prompts for guiding sentiment prediction. The framework consisted of prompt generation, rescoring, and cluster-based expansion, enabling robustness to domain shifts. Experiments across five diverse domains showed that AEP achieved superior performance compared with strong baselines and existing domain adaptation methods, demonstrating its ability to adapt to unseen domains without any target-domain supervision.

In summary, these two different scenarios contributed to a systematic understanding of domain adaptation for ASC.

## 5.2 Answer for Research Questions

As described in Section 1.3, this dissertation is guided by one overall research question: *How can ASC models be effectively adapted to domain shift under different levels of target domain availability: (1) with unlabeled target data, and (2) with no target-domain access?* To address this major question, several sub-questions were formulated. The following provides answers to each sub-question.

- **RQ1:** *How can we construct or augment effective training data when labeled target-domain data is unavailable?*

  In Scenario A, training data were constructed through a combination of pseudo-labeling and cross-aspect review generation. First, pseudo-labels were assigned to unlabeled target reviews using a source-trained classifier under a control of threshold. Second, CARG generated additional target-like reviews by substituting sentiment words and domain-specific keywords. This two-stage process effectively expanded the target domain data space and reduced discrepancy between source and target domain, providing a reliable and sufficiently large training data for adaptation.

  In Scenario B, training data were constructed relies only on source domain data. Since no target data are available during training, the framework replaced conventional data augmentation with a structured process of representation augmentation. First, Aspect-Related Features were extracted from multiple labeled source domains using a PMI-based scoring mechanism, which identified words that were statistically related to each aspect. Second, these ARFs were integrated into a set of manually designed templates. This process converted aspect–ARF pairs from the source data into prompts that simulated various sentiment contexts. Third, these prompts are used to train a generative sentiment classification model so that it learned to associate each aspect with its related feature expressions. Even though the model never used any target domain data during training, it learned general aspect–ARF relationships, such as how the "service" aspect related to other words like "friendly" or "slow", so that these relationships can be effectively transferred to unseen domains. Consequently, the framework built domain-invariant representations through prompt-based supervision rather than relying on explicit target-domain samples.

- **RQ2:** *How can we extract and integrate aspect-related information or features to guide sentiment classification in a way that remains useful?*

In scenario A, we aligned aspects and sentiments by explicitly identifying sentiment words and domain-specific keywords for substitution in CARG. This ensured that generated reviews were not only sentimentally related but also contextually relevant to the aspect.

In scenario B, aspect-sentiment alignment was addressed through the extraction of ARFs using PMI. These ARFs served as anchors in prompts, highlighting aspect-specific hints that guided the sentiment classification model. By embedding ARFs directly into prompts, the model maintained focus on the target aspect even in unseen domains.

Both scenarios demonstrated that aspect-aware information was useful for effective ASC. The key difference lay in how such information was incorporated: source review sentences were used to obtain target-aspect sentences through substitution and generation in Scenario A, whereas ARFs were embedded into prompts to explicitly emphasize aspect–sentiment relations in Scenario B. Together, these complementary approaches confirmed that integrating aspect-oriented features was crucial to robust adaptation under varying levels of target-domain availability.

- **RQ3:** *How can we mitigate noise introduced by pseudo-labels, generated sentences, or irrelevant features?*

  In Scenario A, noise mainly arose from two sources: (1) unreliable pseudo-labels generated with low confidence, and (2) syntactically unnatural or semantically inconsistent generated reviews. To mitigate these issues, a confidence-based filtering mechanism discarded low-confidence pseudo-labeled samples, while fluency-based filtering using pseudo-log-likelihood scores excluded grammatically illogical or contradictory samples. This two filtering steps significantly improved data reliability and quality, resulting in reduction of reduced error propagation during the training time.

  In Scenario B, the major sources of noise were irrelevant ARFs and suboptimal prompt templates. To handle this, the framework adopted a two-step refinement strategy. First, a prompt rescoring mechanism evaluated templates according to their performance in the sentiment classification task, retaining only those that produced consistent predictions. Second, a cluster-based prompt expansion technique aggregated semantically similar prompts and applied weighted voting across clusters, thus reducing the influence of noisy or domain-biased prompts. These mechanisms jointly enhanced robustness and ensured that the

model's predictions remained stable even under domain-agnostic adaptation.

## 5.3 Future Work

This dissertation has proposed two complementary frameworks for ABSA under challenging domain adaptation scenarios: aspect-oriented unsupervised domain adaptation and domain-agnostic adaptation. While the proposed methods have demonstrated significant improvements over prior approaches, several directions remain for future exploration.

### 5.3.1 Future work for Aspect-Oriented unsupervised domain adaptation

In this scenario, we introduced an aspect-oriented perspective, treating each aspect as a domain and transferring knowledge across aspects by generating synthetic labeled sentences and leveraging automatic labeling. Although this method is effective, several directions can further enhance this line of research:

- **Quality of Generated Data:** The CARG process could be improved by leveraging more advanced generative models to generate more fluent and diverse review sentences. Recent open large language models such as LLaMA [63] provide scalable and efficient generation capability, making them promising candidates for improving synthetic data quality in the future.
- **Handling Data Imbalance:** Polarity distributions across aspects are often imbalanced, which negatively impacts classification. While Focal Loss already provides an adaptive weighting mechanism, future work may integrate it with contrastive learning to further enhance class balance and separability. Recent work has shown that contrastive learning can effectively improve sentiment representation separability at the aspect level by pulling together samples with the same polarity and pushing apart those with different polarities [73].

### 5.3.2 Future work for Domain-Agnostic Adaptation

In this scenario, we proposed the AEP framework, which enables adaptation to completely unseen target domains by generating and scoring aspect-aware

prompts. While the results confirm its robustness, the following improvements are promising:

- **Dynamic Prompt Generation:** Current templates are manually designed and rescored. Recent surveys on prompt-based learning indicate that prompts can be automatically generated or refined using large language models, reducing reliance on handcrafted templates [35]. Future work may employ large language models to dynamically generate or refine prompts at inference time.

- **Soft Labels and Fine-Grained Sentiment:** Incorporating soft labels or sentiment valence could enhance the ability of the classification model to capture subtle emotional variations beyond categorical polarity, as suggested by prior work on dimensional sentiment representations and affective lexicons [5, 40].

- **Scalability to Large Models:** As pre-trained models continue to grow in size and capability, adapting AEP to leverage instruction-tuned large language models may further improve adaptability across unseen domains.

- **Cross-Lingual Generalization:** Expanding the framework to multilingual and cross-lingual ABSA would allow testing its robustness under linguistic variations, which are common in global applications. Recent multilingual for ABSA resources such as M-ABSA [71], which covers multiple domains and languages, provide a strong benchmark for evaluating such cross-lingual generalization ability.

# Publications

## Journal

- Binghan Lu, Kiyoaki Shirai, and Natthawut Kertkeidkachorn. Aspect-Enhanced Prompting Method for Unsupervised Domain Adaptation in Aspect-Based Sentiment Analysis. Information, Vol. 16, No. 5, p. 411, 2025. DOI: 10.3390/info16050411.

## International Conference

- Binghan Lu, Kiyoaki Shirai, and Natthawut Kertkeidkachorn. Aspect-Oriented Unsupervised Domain Adaptation for Polarity Classification. In Proceedings of the 18th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP 2023), pp. 1-6, Nov. 2023.

# Appendix A

# List of Templates

Table A.1 shows all the templates of the prompt. “Q” and “D” indicate the question and description templates, respectively. Table A.2 shows the insertion templates used in the experiment, where a review is filled into [TEXT].

Table A.1: List of prompt templates.

| **Template** | **Type** |
|---|---|
| Given the statement text, where aspect [ASPECT] is described as [ARF], what is the sentiment? | Q |
| For the statement text and focusing on [ASPECT] being [ARF], what is the sentiment? | Q |
| Analyze the sentiment of text with emphasis on [ASPECT] being [ARF]. | D |
| How does text portray [ASPECT] as [ARF] in terms of sentiment? | Q |
| Considering the text, what sentiment does [ARF] convey about [ASPECT]? | Q |
| Evaluate the sentiment towards [ASPECT] being [ARF]. | D |
| What is the emotional tone when [ASPECT] is [ARF]? | Q |
| In the text, [ASPECT] is described as [ARF]. What sentiment does this reflect? | Q |
| Assess the feeling towards [ASPECT] being [ARF]. | D |
| Determine the sentiment of [ASPECT] is characterized as [ARF]. | D |
| What emotion is associated with [ASPECT] being [ARF]? | Q |
| Identify the sentiment when [ASPECT] is mentioned as [ARF]. | D |
| Predict the sentiment for [ASPECT] described as [ARF]. | D |
| In the text, [ASPECT] is [ARF]. How does this make the sentiment? | Q |
| Sentiment analysis with [ASPECT] as [ARF]. | D |
| How is the sentiment towards [ASPECT] being [ARF]? | Q |
| What is the sentiment outcome when [ASPECT] equals [ARF]? | Q |
| Review the sentiment with [ASPECT] as [ARF]. | D |
| With [ASPECT] being [ARF], how is the sentiment? | Q |
| Analyze for sentiment with a focus on [ASPECT] as [ARF]. | D |

Table A.2: List of prompt templates (insertion templates).

| **Template** | **Type** |
|---|---|
| Given the statement [TEXT], where aspect [ASPECT] is described as [ARF], what is the sentiment? | Q |
| For the statement [TEXT] and focusing on [ASPECT] being [ARF], what is the sentiment? | Q |
| Analyze the sentiment of [TEXT] with emphasis on [ASPECT] being [ARF]. | D |
| How does [TEXT] portray [ASPECT] as [ARF] in terms of sentiment? | Q |
| Considering the [TEXT], what sentiment does [ARF] convey about [ASPECT]? | Q |
| In the [TEXT], evaluate the sentiment towards [ASPECT] being [ARF]. | D |
| In the [TEXT], what is the emotional tone when [ASPECT] is [ARF]? | Q |
| In the [TEXT], [ASPECT] is described as [ARF]. What sentiment does this reflect? | Q |
| In the [TEXT], assess the feeling towards [ASPECT] being [ARF]. | D |
| In the [TEXT], determine the sentiment of [ASPECT] is characterized as [ARF]. | D |

Table A.2: *Cont.*

| **Template** | **Type** |
|---|---|
| In the [TEXT], what emotion is associated with [ASPECT] being [ARF]? | Q |
| In the [TEXT], identify the sentiment when [ASPECT] is mentioned as [ARF]. | D |
| In the [TEXT], predict the sentiment for [ASPECT] described as [ARF]. | D |
| In the [TEXT], [ASPECT] is [ARF]. How does this make the sentiment? | Q |
| In the [TEXT], sentiment analysis with [ASPECT] as [ARF]. | D |
| In the [TEXT], how is the sentiment towards [ASPECT] being [ARF]? | Q |
| In the [TEXT], what is the sentiment outcome when [ASPECT] equals [ARF]? | Q |
| In the [TEXT], review the sentiment with [ASPECT] as [ARF]. | D |
| In the [TEXT], with [ASPECT] being [ARF], how is the sentiment? | Q |
| In the [TEXT], analyze for sentiment with a focus on [ASPECT] as [ARF]. | D |

Table A.3 lists the additional prompt templates automatically generated by a large language model (LLM). The placeholder [ARF] indicates the position where the aspect-related feature is inserted.

Table A.3: List of LLM-generated prompt templates.

| **Template** | **Type** |
|---|---|
| The [ASPECT] is [ARF]. | D |
| Many people find the [ASPECT] to be [ARF]. | D |
| Users often describe the [ASPECT] as [ARF]. | D |
| One common opinion is that the [ASPECT] is [ARF]. | D |
| It is said that the [ASPECT] is [ARF]. | D |
| Reviews frequently mention the [ASPECT] being [ARF]. | D |
| Customers have noted the [ASPECT] as [ARF]. | D |
| The [ASPECT] seems to be [ARF] according to reviews. | D |
| People consider the [ASPECT] to be [ARF]. | D |
| A lot of feedback describes the [ASPECT] as [ARF]. | D |
| From the reviews, it appears that the [ASPECT] is [ARF]. | D |
| The [ASPECT] tends to be [ARF]. | D |
| Comments suggest that the [ASPECT] is [ARF]. | D |
| Based on user feedback, the [ASPECT] is [ARF]. | D |
| Many reviewers point out that the [ASPECT] is [ARF]. | D |
| The general sentiment toward the [ASPECT] is that it's [ARF]. | D |
| People commonly think the [ASPECT] is [ARF]. | D |
| The [ASPECT] has been described as [ARF]. | D |
| Several users reported that the [ASPECT] is [ARF]. | D |
| The perception of the [ASPECT] is that it is [ARF]. | D |
| Most users agree that the [ASPECT] is [ARF]. | D |
| The [ASPECT] consistently appears to be [ARF]. | D |
| Experience shows the [ASPECT] is [ARF]. | D |
| In many reviews, the [ASPECT] comes across as [ARF]. | D |
| It's generally believed that the [ASPECT] is [ARF]. | D |

(continued)

| **Template** | **Type** |
|---|---|
| The [ASPECT] gets described as [ARF] in user comments. | D |
| According to feedback, the [ASPECT] is [ARF]. | D |
| It can be observed that the [ASPECT] is [ARF]. | D |
| The [ASPECT] is usually seen as [ARF]. | D |
| Users repeatedly mention that the [ASPECT] is [ARF]. | D |
| What do people think about the [ASPECT] being [ARF]? | Q |
| Is the [ASPECT] considered [ARF] by users? | Q |
| How do reviewers describe the [ASPECT] by [ARF]? | Q |
| Why is the [ASPECT] said to be [ARF]? | Q |
| Do people agree that the [ASPECT] is [ARF]? | Q |
| In what way is the [ASPECT] [ARF]? | Q |
| Can the [ASPECT] be described as [ARF]? | Q |
| What kind of opinion is common about the [ASPECT] as [ARF]? | Q |
| Has the [ASPECT] been called [ARF] in reviews? | Q |
| How often is the [ASPECT] labeled as [ARF]? | Q |
| What are the impressions of the [ASPECT] being [ARF]? | Q |
| Do users find the [ASPECT] to be [ARF]? | Q |
| What does user feedback say about the [ASPECT] as [ARF]? | Q |
| Is there consensus that the [ASPECT] is [ARF]? | Q |
| What makes the [ASPECT] [ARF] according to reviews? | Q |
| How is the [ASPECT] usually perceived by [ARF]? | Q |
| What reasons do users give for calling the [ASPECT] [ARF]? | Q |
| What is a common description of the [ASPECT] as [ARF]? | Q |
| Are there many mentions of the [ASPECT] being [ARF]? | Q |
| How do people feel about the [ASPECT] described as [ARF]? | Q |

# Bibliography

[1] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 2200–2204, 2010.

[2] Eyal Ben-David, Nadav Oved, and Roi Reichart. PADA: Example-based Prompt Learning for on-the-fly Adaptation to Unseen Domains. *Transactions of the Association for Computational Linguistics*, 10:414–433, 2022.

[3] John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 120–128, Sydney, Australia, 2006. Association for Computational Linguistics.

[4] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

[5] Sven Buechel and Udo Hahn. EmoBank: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In Mirella Lapata, Phil Blunsom, and Alexander Koller, editors, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 578–585, Valencia, Spain, April 2017. Association for Computational Linguistics.

[6] Hongjie Cai, Rui Xia, and Jianfei Yu. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 340–350, 2021.

[7] Minmin Chen, Zhixiang Xu, Kilian Weinberger, and Fei Sha. Marginalized denoising autoencoders for domain adaptation, 2012.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[9] Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. Target-oriented opinion words extraction with target-fused neural sequence labeling. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *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 2509–2518, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks, 2016.

[11] Tianhao Gao, Jun Fang, Hanyu Liu, Zhiyuan Liu, Chao Liu, Pengzhang Liu, Yongjun Bao, and Weipeng Yan. LEGO-ABSA: A prompt-based task assemblable unified generative framework for multi-task aspect-based sentiment analysis. In Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na, editors, *Proceedings of the 29th International Conference on Computational Linguistics*, pages 7002–7012, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics.

[12] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners, 2021.

[13] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. In *Association for Computational Linguistics (ACL)*, 2021.

[14] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *International Conference on Machine Learning*, 2011.

[15] Google. T5-base model. `https://huggingface.co/google/t5-base`, 2024. Accessed on 1 October 2024.

[16] Han Guo, Ramakanth Pasunuru, and Mohit Bansal. Multi-source domain adaptation for text classification via distancenet-bandits, 2020.

[17] Hongyu Guo, Yongyi Mao, and Richong Zhang. Augmenting data with mixup for sentence classification: An empirical study, 2019.

[18] Jiang Guo, Darsh Shah, and Regina Barzilay. Multi-source domain adaptation with mixture of experts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4694–4703, Brussels, Belgium, 2018. Association for Computational Linguistics.

[19] Xiaochuang Han and Jacob Eisenstein. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. 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 4238–4248, Hong Kong, China, 2019. Association for Computational Linguistics.

[20] Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. An unsupervised neural attention model for aspect extraction. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 388–397, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[21] Mickel Hoang, Oskar Alija Bihorac, and Jacobo Rouces. Aspect-Based sentiment analysis using BERT. In Mareike Hartmann and Barbara Plank, editors, *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 187–196, Turku, Finland, September–October 2019. Linköping University Electronic Press.

[22] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 168–177, New York, NY, USA, 2004. Association for Computing Machinery.

[23] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177, Seattle, Washington, USA, 2004.

[24] Lianzhe Huang, Xin Sun, Sujian Li, Linhao Zhang, and Houfeng Wang. Syntax-Aware graph attention network for aspect-level sentiment classification. In Donia Scott, Nuria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 799–810, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[25] Constantinos Karouzos, Georgios Paraskevopoulos, and Alexandros Potamianos. UDALM: Unsupervised domain adaptation through language modeling. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2579–2590, Online, 2021. Association for Computational Linguistics.

[26] Yoon Kim. Convolutional neural networks for sentence classification. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics.

[27] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

[28] Chengxi Li, Feiyu Gao, Jiajun Bu, Lu Xu, Xiang Chen, Yu Gu, Zirui Shao, Qi Zheng, Ningyu Zhang, Yongpan Wang, and Zhi Yu. SentiPrompt: Sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis, 2021.

[29] Rui Li, Cheng Liu, Yu Tong, and Jiang Dazhi. Feature structure matching for multi-source sentiment analysis with efficient adaptive tuning. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7153–7162, Torino, Italia, 2024. ELRA and ICCL.

[30] Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard Hovy. Dual graph convolutional networks for aspect-based sentiment analysis. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *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, August 2021. Association for Computational Linguistics.

[31] Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. Exploiting BERT for end-to-end aspect-based sentiment analysis. In Wei Xu, Alan Ritter, Tim Baldwin, and Afshin Rahimi, editors, *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 34–41, Hong Kong, China, November 2019. Association for Computational Linguistics.

[32] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *arXiv:1708.02002*, 2017.

[33] Bing Liu. Sentiment analysis and subjectivity. In *Handbook of Natural Language Processing*, 2010.

[34] Bing Liu and Lei Zhang. *A Survey of Opinion Mining and Sentiment Analysis*, pages 415–463. Springer US, Boston, MA, 2012.

[35] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, 2021.

[36] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks, 2015.

[37] Xiaofei Ma, Peng Xu, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. Domain adaptation with BERT-based domain classification and data selection. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 76–83, Hong Kong, China, 2019. Association for Computational Linguistics.

[38] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *Proceedings of the 49th Annual Meeting of the Association for*

*Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[39] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[40] Saif Mohammad. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 English words. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–184, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[41] Sajad Movahedi, Erfan Ghadery, Heshaam Faili, and Azadeh Shakery. Aspect category detection via Topic-Attention network, 2019.

[42] Tetsuji Nakagawa, Kentaro Inui, and Sadao Kurohashi. Dependency tree-based sentiment classification using CRFs with hidden variables. In Ron Kaplan, Jill Burstein, Mary Harper, and Gerald Penn, editors, *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 786–794, Los Angeles, California, June 2010. Association for Computational Linguistics.

[43] OpenAI. ChatGPT (june 2025 version). `https://chat.openai.com/`, 2024. [Large language model].

[44] Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, pages 751–760, 2010.

[45] Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278, Barcelona, Spain, July 2004.

[46] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135, 2008.

[47] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques, 2002.

[48] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 486–495, 2015.

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

[50] Amir Pouran Ben Veyseh, Nasim Nouri, Franck Dernoncourt, Dejing Dou, and Thien Huu Nguyen. Introducing syntactic structures into target opinion word extraction with deep learning. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8947–8956, Online, November 2020. Association for Computational Linguistics.

[51] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023.

[52] Alan Ramponi and Barbara Plank. Neural unsupervised domain adaptation in NLP—A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online), 2020. International Committee on Computational Linguistics.

[53] Marzieh Saeidi, Guillaume Bouchard, Maria Liakata, and Sebastian Riedel. SentiHood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods. In Yuji Matsumoto and Rashmi Prasad, editors, *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1546–1556, Osaka, Japan, 2016. The COLING 2016 Organizing Committee.

[54] Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, 2020.

[55] Tian Shi, Liuqing Li, Ping Wang, and Chandan K. Reddy. A simple and effective self-supervised contrastive learning framework for aspect detection, 2020.

[56] Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

[57] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In David Yarowsky, Timothy Baldwin, Anna Korhonen, Karen Livescu, and Steven Bethard, editors, *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

[58] Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. *Targeted Sentiment Classification with Attentional Encoder Network*, page 93–103. Springer International Publishing, 2019.

[59] Xinjie Sun, Kai Zhang, Qi Liu, Meikai Bao, and Yanjiang Chen. Harnessing domain insights: A prompt knowledge tuning method for aspect-based sentiment analysis. *Knowledge-Based Systems*, 298:111975, 2024.

[60] Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. Effective LSTMs for target-dependent sentiment classification. In *International Conference on Computational Linguistics*, 2015.

[61] Duyu Tang, Bing Qin, and Ting Liu. Document modeling with gated recurrent neural network for sentiment classification. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1432, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[62] Cigdem Toprak, Niklas Jakob, and Iryna Gurevych. Sentence and expression level annotation of opinions in user-generated discourse. In Jan Hajič, Sandra Carberry, Stephen Clark, and Joakim Nivre, editors, *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 575–584, Uppsala, Sweden, 2010. Association for Computational Linguistics.

[63] 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. LLaMA: Open and efficient foundation language models, 2023.

[64] Stéphan Tulkens and Andreas van Cranenburgh. Embarrassingly simple unsupervised aspect extraction. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3182–3187, Online, July 2020. Association for Computational Linguistics.

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

[66] Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. Recursive neural conditional random fields for aspect-based sentiment analysis. In Jian Su, Kevin Duh, and Xavier Carreras, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 616–626, Austin, Texas, November 2016. Association for Computational Linguistics.

[67] Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 3316–3322. AAAI Press, 2017.

[68] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. Attention-based LSTM for aspect-level sentiment classification. In Jian Su, Kevin Duh, and Xavier Carreras, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas, November 2016. Association for Computational Linguistics.

[69] Zengzhi Wang, Rui Xia, and Jianfei Yu. UnifiedABSA: A unified absa framework based on multi-task instruction tuning, 2022.

[70] Dustin Wright and Isabelle Augenstein. Transformer based multi-source domain adaptation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7963–7974, Online, 2020. Association for Computational Linguistics.

[71] Chengyan Wu, Bolei Ma, Yihong Liu, Zheyu Zhang, Ningyuan Deng, Yanshu Li, Baolan Chen, Yi Zhang, Yun Xue, and Barbara Plank. M-absa: A multilingual dataset for aspect-based sentiment analysis, 2025.

[72] Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. Double embeddings and CNN-based sequence labeling for aspect extraction. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 592–598, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[73] Lingling Xu and Weiming Wang. Improving aspect-based sentiment analysis with contrastive learning. *Natural Language Processing Journal*, 3:100009, 2023.

[74] Lu Xu, Hao Li, Wei Lu, and Lidong Bing. Position-Aware tagging for aspect sentiment triplet extraction. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2339–2349, Online, November 2020. Association for Computational Linguistics.

[75] Wei Xue and Tao Li. Aspect based sentiment analysis with gated convolutional networks. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[76] Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. A unified generative framework for aspect-based sentiment analysis. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *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 2416–2429, Online, August 2021. Association for Computational Linguistics.

[77] Songhua Yang, Xinke Jiang, Hanjie Zhao, Wenxuan Zeng, Hongde Liu, and Yuxiang Jia. FaiMA: Feature-aware in-context learning for multi-

domain aspect-based sentiment analysis. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7089–7100, Torino, Italia, 2024. ELRA and ICCL.

[78] Wen Yin, Cencen Liu, Yi Xu, Ahmad Raza Wahla, Huang Yiting, and Dezhang Zheng. SynPrompt: Syntax-aware enhanced prompt engineering for aspect-based sentiment analysis. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15469–15479, Torino, Italia, May 2024. ELRA and ICCL.

[79] Yichun Yin, Furu Wei, Li Dong, Kaimeng Xu, Ming Zhang, and Ming Zhou. Unsupervised word and dependency path embeddings for aspect term extraction, 2016.

[80] Jianfei Yu, Chenggong Gong, and Rui Xia. Cross-domain review generation for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4767–4777, Online, 2021. Association for Computational Linguistics.

[81] Jianfei Yu, Chenggong Gong, and Rui Xia. Cross-domain review generation for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics*, pages 4767–4777, 2021.

[82] Chen Zhang, Qiuchi Li, and Dawei Song. Aspect-based sentiment classification with aspect-specific graph convolutional networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *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 4568–4578, Hong Kong, China, November 2019. Association for Computational Linguistics.

[83] Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. Towards generative aspect-based sentiment analysis. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*

*and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 504–510, Online, August 2021. Association for Computational Linguistics.

[84] Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges, 2022.

[85] Zishuo Zhao, Ziyang Ma, Zhenzhou Lin, Jingyou Xie, Yinghui Li, and Ying Shen. Source-free domain adaptation for aspect-based sentiment analysis. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15076–15086, Torino, Italia, 2024. ELRA and ICCL.

[86] Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. Representation learning for aspect category detection in online reviews. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 417–423. AAAI Press, 2015.