Stars Are All You Need: A Distantly Supervised Pyramid Network for Unified Sentiment Analysis

Wenchang Li
Sichuan University
liwenchang97@gmail.com

Yixing Chen
University Notre Dame
ychen43@nd.edu

Shuang Zheng
Dalian University of Technology
zhengshuang99@mail.dlut.edu.cn

Lei Wang
Meituan
wanglei46@meituan.com

John P. Lalor
University of Notre Dame
john.lalor@nd.edu

Abstract

Data for the Rating Prediction (RP) sentiment analysis task such as star reviews are readily available. However, data for aspect-category detection (ACD) and aspect-category sentiment analysis (ACSA) is often desired because of the fine-grained nature but are expensive to collect. In this work, we propose Unified Sentiment Analysis (Uni-SA) to understand aspect and review sentiment in a unified manner. Specifically, we propose a Distantly Supervised Pyramid Network (DSPN) to efficiently perform ACD, ACSA, and RP using only RP labels for training. We evaluate DSPN on multi-aspect review datasets in English and Chinese and find that in addition to the internal efficiency of sample size, DSPN also performs comparably well to a variety of benchmark models. We also demonstrate the interpretability of DSPN's outputs on reviews to show the pyramid structure inherent in unified sentiment analysis.

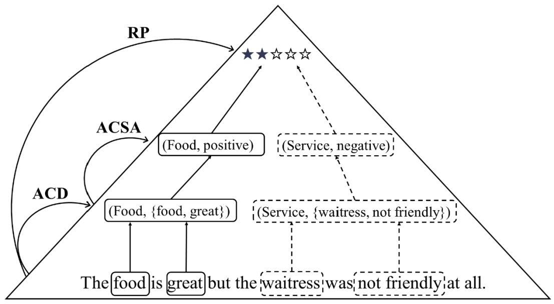


Figure 1: An overview of Unified Sentiment Analysis (Uni-SA). While ACD, ACSA, and RP can be performed individually, by leveraging the implicit pyramid structure of reviews, we can efficiently perform all three tasks with only RP labels.

1 Introduction

Consumers generate online reviews for millions of products and services in various contexts, including hotels, restaurants, products, and schools, on platforms such as Yelp, Amazon, and Tripadvisor. Firms can use online review data to better understand consumer behavior and build predictive models for their businesses (Zhang et al., 2023). Sentiment analysis of an entire document is a widely-used method for understanding unstructured consumer reviews at a high level (Liu and Zhang, 2012). In addition, fine-grained analysis of user generated content can detect aspects in documents (e.g., food quality and price in restaurant reviews). These aspects can be classified according to their sentiment (Schouten and Frasincar, 2015).

A holistic view of sentiment analysis includes three tasks: identifying aspects in the document (Aspect-Category Detection, ACD), classifying aspect sentiment (Aspect-Category Sentiment Analysis, ACSA), and classifying the overall sentiment of the document (Rating Prediction, RP).

For example, consider the review displayed in Figure 1: "*The food is great but the waitress was not friendly at all.*" Sentiment analysis models can first identify the aspects mentioned in this review via ACD (*Food*, *Service*), then predict their corresponding sentiment polarities with ACSA (*Food:Positive*, *Service:Negative*). Finally, an RP model will predict the star rating that a user would give for the review (two stars). With these methods, businesses can use both fine-grained and coarse-grained sentiment information to identify customer pain points and improve service quality.

Typically, NLP models consider ACD, ACSA, and RP independently. In some cases, ACD and ACSA are learned by a single model (e.g., Schmitt et al., 2018; Liu et al., 2021), but these two tasks are rarely connected to RP (Chebolu et al., 2023). However, star rating labels for RP are usually cheaper and easier to obtain than ACSA labels due to widespread availability of user-generated review text and stars online (Li et al., 2020a). More importantly, they can be considered a "coarse-grained synthesis" of ratings across aspects in the review (Bu et al., 2021). For example, if a user

Proceedings of the Ninth Workshop on Noisy and User-generated Text (W-NUT 2024), pages 104–118
March 22, 2024

states that the food is good, but the service quality is unacceptable, they will consider these two aspects together when giving an overall two-star rating (Figure 1), which implies that the aspect-level polarities inform the overall review of two stars (out of a possible five). This relationship provides an opportunity to unify the multiple tasks. Specifically in this work, we hypothesize that review-level star rating labels represent an aggregation of aspect-level sentiments, which themselves can be aggregated from word-level sentiments (Li et al., 2020c). To efficiently model this structure as a *pyramid structure*, we propose a Distantly Supervised Pyramid Network (DSPN) that requires *only RP labels* as signal to unify the three tasks of ACD, ACSA, and RP. We call this unified sentiment task Unified Sentiment Analysis (Uni-SA).

Contributions In this work, we make the following contributions:

- We introduce *Unified Sentiment Analysis* as a unified task of three key sentiment analysis tasks, specifically ACD, ACSA, and RP,
- We propose Distantly Supervised Pyramid Network (DSPN), a novel model for unified sentiment analysis. DSPN shows significant efficiency on training sample size with *only RP labels* as training input.
- We propose a novel aspect-attention mechanism for ACD to inform ACSA and capture the pyramid sentiment structure,
- We validate DSPN through experimental results on Chinese and English multi-aspect datasets and demonstrate the effectiveness and efficiency of DSPN.[1]

2 Unified Sentiment Analysis

Before describing our model, we first define our notation and present the unifying framework of Uni-SA. We borrow notation from the prior work where possible and introduce new notation as needed for consistency across tasks (Pontiki et al., 2016). For reference, we have included a comprehensive notation table in the appendices (Appendix A). Our corpus is a collection of *reviews* $\mathbf{R} = \{R_1, R_2, \ldots, R_{|\mathbf{R}|}\}$. Each review R_i consists of a sequence of word tokens (hereafter "words"): $R_i = \{t_i^{(1)}, t_i^{(2)}, \ldots, t_i^{(n)}\}$.

[1] Code available at https://github.com/nd-ball/DSPN

2.1 Aspect-Category Detection

In the ACD task, there are N predefined aspect categories (hereafter "aspects"): $A = \{A_1, A_2, \ldots, A_N\}$. The set of aspects present in R_i is defined as: $A_{R_i} = \{A_{R_i}^{(1)}, A_{R_i}^{(2)}, \ldots, A_{R_i}^{(K)}\}$, where $K \leq N$. To train unsupervised ACD models, the required training data is simply $\mathbf{R}$.

2.2 Aspect-Category Sentiment Analysis

For a given review R_i and one of its aspects $A_{R_i}^{(j)}$, the goal of ACSA is to predict the polarity of the aspect: $\hat{y}_{A_{R_i}^{(j)}}$. Aspect polarity is typically binary (*positive* or *negative*) or categorical (with a third option of *neutral*). Supervised ACSA models require review-aspect-polarity triples: $\{R_i, (A_{R_i}^{(j)}, y_{A_{R_i}^{(j)}})_{j=1}^{K}\}_{i=1}^{|\mathbf{R}|}$. In the case of multi-aspect ACSA, there are multiple aspects present in each review, and therefore ACSA requires $K \times |\mathbf{R}|$ labels, a factor of K larger than in RP.

2.3 Rating Prediction

Given a review R_i, RP aims to predict the star rating $\hat{y}_{R_i}$. Supervised RP models requires review-sentiment tuples: $\{(R_i, y_{R_i})\}_{i=1}^{|\mathbf{R}|}$

2.4 Model Running

Typically ACD, ACSA, and RP are considered standalone tasks. Here we propose a unified approach, where with training data of *only* RP labels, a model can output present aspects (ACD), the sentiment of those aspects (ACSA), and an overall document-level sentiment score (RP). This approach uses training labels from a single task to efficiently learn multiple distinct sentiment analysis tasks.

More specifically, for a model M, the training data required is the same as the RP task: $\{(R_i, y_{R_i})\}_{i=1}^{|\mathbf{R}|}$. At run-time, the model provides three outputs for a new review R_i: (1) The predicted aspects present in the review ($\hat{A}_{R_i}$), (2) the sentiment polarity of each identified aspect ($\hat{y}_{A_{R_i^{(j)}}} \forall A_{R_i^{(j)}} \in \hat{A}_{R_i}$), and (3) the overall sentiment prediction for the review ($\hat{y}_{R_i}$).

3 Distantly Supervised Pyramid Network

In this section, we describe DSPN for Uni-SA. The overall model architecture is illustrated in Figure 2.

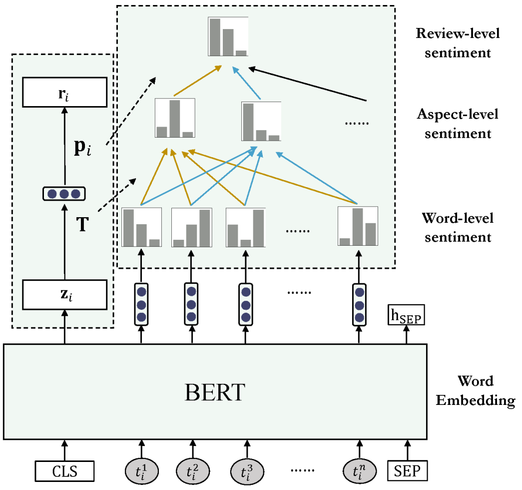


Figure 2: Overall architecture of DSPN. Aspect embedding matrix $\mathbf{T}$ is used to calculate the distance between words and aspects, which is regarded as the word-level attention weights for each aspect. Aspect importance $\mathbf{p}_i$ is learned by Module 1 and is used as the attention weights of aspects.

3.1 Module 1: Aspect-Category Detection

For the ACD task, we utilize an autoencoder-style network (He et al., 2017). For a review R_i, the input sequence X_i is constructed as $\{[CLS], t_i^{(1)}, t_i^{(2)}, \ldots, t_i^{(n)}, [SEP]\}$. We use BERT (Devlin et al., 2019) to generate embeddings for each example, $\mathbf{z}_i$.

To generate aspect embeddings, we first set the aspect and keyword map dictionary for each aspect. Then for each aspect, we use BERT to encode the sentence composed of key words related to the aspect and obtain its output as the initial embedding of the aspect. In this way, we initialize the aspect embedding matrix $\mathbf{T}$. [2] Lastly, Module 1 performs sentence reconstruction at the aspect-level through a linear layer:

$$\mathbf{z}_i = \text{BERT}(X_i) \tag{1}$$

$$\mathbf{p}_i = \text{softmax}(\mathbf{W_1} \cdot \mathbf{z}_i + \mathbf{b_1}) \tag{2}$$

$$\mathbf{r}_i = \mathbf{T}^\top \cdot \mathbf{p}_i \tag{3}$$

[2]There are N predefined aspects in ACD task, and many prior works have identified the representative words for each one of them (Bu et al., 2021; Wang et al., 2010). For example, "staff", "customer", and "friendly" can be the representative words for "Service" aspect. Based on this, we proposed to firstly construct a sentence that contains top representative words, then use the embedding of this sentence as the initial embedding for the aspect.

where $\mathbf{r}_i$ is the reconstructed sentence embedding and $\mathbf{p}_i$ is the aspect importance vector.

The loss function for Module 1 is defined as a hinge loss to maximize the inner product between the input sentence embedding and its reconstruction while minimizing the inner product between the input sentence embedding and randomly sampled negative examples:

$$L(\theta_{\text{ACD}}) = \sum_{R_i \in \mathbf{R}} \sum_{j=1}^{m} \phi_{R_i,j} + \lambda_{\text{ACD}} U(\theta) \tag{4}$$

$$\phi_{R_i,j} = max(0, 1 - \mathbf{r}_i \mathbf{z}_i + \mathbf{r}_i \mathbf{n}_j) \tag{5}$$

where $\mathbf{n}_i$ represents each negative sample, and $U(\theta)$ represents the regularization term to encourage unique aspect embeddings (He et al., 2017).

The aspect embedding matrix $\mathbf{T}$ and aspect importance vector $\mathbf{p}_i$ are inputs for attention calculation in DSPN's pyramid network (Module 2).

3.2 Module 2: Pyramid Sentiment Analysis

Module 2 is based on the intuition that the sentiment of a review is an aggregation of the sentiments of the aspects contained in the review (Bu et al., 2021). In addition, the sentiment of an aspect is an aggregation of the sentiments of the words indicating that aspect, forming a three-layer structure. We propose using a pyramid network to capture this structure, and we can use easy-to-obtain RP ratings as training labels.

3.2.1 Word Sentiment Prediction Layer

We use the hidden vector of each word output by BERT to obtain word representations, where $\mathbf{h}_i^{(j)}$ is the representation of the j-th word. We use two fully connected layers to produce a word-level sentiment prediction vector:

$$\mathbf{w}_i^{(j)} = \mathbf{W_3} \cdot ReLU(\mathbf{W_2} \cdot \mathbf{h}_i^{(j)} + \mathbf{b_2}) + \mathbf{b_3} \tag{6}$$

3.2.2 ACSA with Aspect Attention

We can calculate the similarity of words and aspects using the word representations and the aspect embedding matrix $\mathbf{T}$ output by Module 1. This similarity will be treated as the attention weights of words for the aspect. When predicting aspect-level sentiment, for the k-th aspect, the sentiment S_a^k is computed as:

Dataset	Language	MA	MAS	Split	Reviews	Overall Sentiment			Aspect Sentiments		
						Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
TripDMS	English	100%	100%	Train	23,515	8,998	5,055	9,462	64,984	34,200	43,391
				Val	2,939	1,161	613	1,165	8,174	4,245	5,349
				Test	2,939	1,079	647	1,213	8,002	4,355	5,437
ASAP	Chinese	95.97%	63.85%	Train	36,850	29,132	5,241	2,477	77,507	27,329	17,299
				Val	4,940	3,839	784	317	10,367	3,772	2,373
				Test	4,940	3,885	717	338	10,144	3,729	2,403

Table 1: Statistics of the datasets. **MA** is the percentage of multi-aspect instances in the dataset and **MAS** is the percentage of multi-aspect multi-sentiment instances.

$$d_k^{(j)} = \mathbf{T}_k^\top \cdot \mathbf{h}_i^{(j)} \tag{7}$$

$$a_k^{(j)} = \frac{\exp(d_k^{(j)})}{\sum_{m=1}^{n} \exp(d_k^{(m)})} \tag{8}$$

$$S_a^k = softmax(\sum_{j=1}^{n} \mathbf{w}_i^{(j)} a_k^{(j)}) \tag{9}$$

3.2.3 Review Prediction

Review-level sentiment S_r is computed by:

$$S_r = softmax(S_a \cdot \mathbf{p}_i) \tag{10}$$

Here $\mathbf{p}_i$ is the aspect importance vector output by Module 1 (§3.1), which is regarded as the attention weights of aspects in a review. S_a is the matrix concatenation of aspect-level sentiments across the K aspects in the review.

3.3 Loss

For the RP task, as each prediction is a 3-class classification problem, the loss function is defined by the categorical cross-entropy between the true label and the model output:

$$L(\theta_{\text{RP}}) = -\sum_i S_{gold} \cdot log(S_r) \tag{11}$$

We jointly train DSPN for RP and ACD by minimizing the combined loss function:

$$L(\theta) = \lambda L(\theta_{\text{ACD}}) + L(\theta_{\text{RP}}) \tag{12}$$

where λ is the weight of ACD loss. Although no direct supervision is required for ACSA, due to the construction of DSPN, the model inherently learns aspect sentiment predictions.

4 Experiments

4.1 Datasets

To validate DSPN's contribution as an efficient and effective model for unified sentiment analysis, we experiment with two datasets. Statistics of the two datasets are given in Table 1. While DSPN can learn ACD, ACSA, and RP with only RP labels, we require datasets for our benchmarking that have ACD, ACSA, and RP labels.[3]

ASAP ASAP is a Chinese-language restaurant review dataset from a leading e-commerce platform in China (Bu et al., 2021). ASAP includes RP labels and ACSA labels. RP labels are categorical on a 5-star scale. ACSA labels are categorical (*positive*, *negative*, *neutral*) for each aspect#attribute [4] identified in the review text (Pontiki et al., 2016). For ACSA we aggregate sentiment at the entity level for a total of five aspects: {Food, Price, Location, Service, Ambience} by majority vote.

TripDMS TripDMS is an English-language hotel review dataset from Tripadvisor.com (Wang et al., 2010; Yin et al., 2017). TripDMS RP labels are categorical on a 5-star scale. ACSA labels are categorical (*positive*, *negative*, *neutral*) for seven aspects: {Value, Room, Location, Cleanliness, Check-in, Service, Business}.

4.2 Evaluation

DSPN's main contribution is accurate and efficient unified sentiment analysis via distant supervision. We therefore compare DSPN to existing ACD, ACSA, and RP models.

4.2.1 Aspect-Category Detection

In the ACD task, we compare DSPN with fully unsupervised ABAE (He et al., 2017). To more fairly compare with the prior work, we replace the underlying encoder of ABAE with a BERT encoder and update the aspect embedding matrix $\mathbf{T}$ initialization accordingly. We call this ABAE-BERT and

[3]To the best of our knowledge, these datasets are the only ones with RP and ACSA labels for us to evaluate performance.

[4]ASAP defines 5 aspects and 18 attributes.

		Efficiency			Performance		
		Parameters (MM)	Labels (thousands)	Training Time (minutes)	ACD (F1)	ACSA (Acc)	RP (Acc)
TripDMS	ABAE-BERT (ACD)	91.2	0	40	92.3		
	AC-MIMML-BERT (ACSA)	105	164.6	55		64.3	
	BERT-ITPT-FiT (RP)	82.7	23.5	102			72.4
	Pipeline	278.9	188.1	197	92.3	**64.3**	72.4
	DSPN	**102.9**	**23.5**	**95**	**92.7**	53.2	**72.5**
	Delta	-63.1	-87.5	-51.8	0.43	-17.3	0.14
ASAP	ABAE-BERT (ACD)	97.5	0	42	80.1		
	AC-MIMML-BERT (ACSA)	107.2	184.3	55		77.2	
	BERT-ITPT-FiT (RP)	91	36.9	110			80.3
	Pipeline	295.7	221.1	207	**80.1**	**77.2**	80.3
	DSPN	**111**	**36.9**	**88**	79.4	65.4	**81.3**
	Delta	-62.5	-83.3	-57.5	-0.87	-15.3	1.3

Table 2: Comparison between DSPN and a high-performance pipeline approach to unified sentiment analysis.

		Efficiency			Performance		
		Parameters (MM)	Labels (thousands)	Training Time (minutes)	ACD (F1)	ACSA (Acc)	RP (Acc)
TripDMS	ABAE	3.1	0	15	91.2		
	GCAE	4.2	164.6	5		55.1	
	BERT-Feat	80.2	23.5	35			71.4
	Pipeline	**87.5**	188.1	**55**	91.2	**55.1**	71.4
	DSPN	102.9	**23.5**	95	**92.7**	53.2	**72.5**
	Delta	17.60	-87.50	72.73	1.64	-3.45	1.54
ASAP	ABAE	3.1	0	15	79.4		
	GCAE	4.4	184.3	6		70.3	
	BERT-Feat	80.8	36.9	42			79.2
	Pipeline	**88.3**	221.1	**63**	**79.4**	**70.3**	79.2
	DSPN	111	**36.9**	88	**79.4**	65.4	**81.3**
	Delta	25.71	-83.33	39.68	0.00	-6.97	2.65

Table 3: Comparison between DSPN and a high-efficiency pipeline approach to unified sentiment analysis.

report its performance.[5] In the experiment, we follow previous work (Ruder et al., 2016; Ghadery et al., 2019) and use thresholding to assign aspects whose probability exceeds a given threshold to the corresponding review. We choose the threshold that produces the best performance ($1e^{-4}$) in our experiment. We evaluate ACD using F1 score to determine the quality of the identified aspects (He et al., 2017).

4.2.2 Aspect-Category Sentiment Analysis

For ACSA, we use several strong supervised ACSA models. Our benchmark models include non-BERT models: GCAE (Xue and Li, 2018), End2end-LSTM/CNN (Schmitt et al., 2018), and AC-MIMLLN (Li et al., 2020c) as well as BERT-based models: AC-MIMLLN-BERT (Li et al., 2020c) and ACSA-Generation (Liu et al., 2021). We use accuracy to evaluate ACSA (Li et al., 2020b).

4.2.3 Rating Prediction

The RP task a text classification task. Therefore, we compare DSPN with several BERT fine tuning strategies (Sun et al., 2019): BERT-Feat, BERT-FiT, and BERT-ITPT-FiT. Consistent with prior work (e.g., Aly and Atiya, 2013; Mudinas et al., 2012), we convert the 5-star RP rating into three classes (Negative, Neural, and Positive). To evaluate RP models, we use accuracy.

4.2.4 Implementation details

We implement models in PyTorch. The batch sizes are set to 32 for all models. Non-BERT models are optimized by the Adam optimizer, while BERT models use BERTAdam optimizer. We set the learning rate as 5e-5, and use early stopping with a pa-

[5] In ABAE-BERT, we don't need to manually define the meaning of aspect by looking at the nearest K words in the embedding space.

tience of 3 during training. We set the negative samples as 5 due to GPU constraints. We report results averaged over five runs.

5 Results

5.1 Overall Performance

To compare DSPN to the existing models, we compare DSPN with a pipeline approach. We create two pipelines: a *high performance* pipeline where we use the best performing model for each task in the pipeline, and a *high efficiency* model, where we use the most efficient benchmark model in terms of parameters in our pipeline.

Tables 2 and 3 presents the results of our comprehensive benchmarking. We first note that DSPN is the *only model capable of performing all three tasks*. What's more, DSPN is able to perform all three tasks with only supervision for the RP task. For RP, DSPN outperforms all of our benchmark models. On TripDMS, DSPN demonstrates stronger F1 score in ACD task than ABAE. On both datasets, our proposed ABAE-BERT outperforms original ABAE, demonstrating that incorporating large language models leads to higher quality aspects.

DSPN's performance on ACSA is lower than the supervised benchmarks. This is to be expected as DSPN's only supervision is RP labels. From an efficiency point of view, ACSA models require 164,605 labels on TripDMS to learn one task (ACSA), while DSPN only requires 23,515 labels (86% fewer) to learn three tasks. Based on an 86% size gap, DSPN performance is 17% lower than the best-performing supervised model for ACSA. Similarly for ASAP, based on an 80% size gap, DSPN performance is 15% lower than the best-performing supervised model for ACSA. In fact, DSPN outperforms the fully-supervised End2end-CNN baseline model.

Our single-task benchmarks serve to set the "upper-bound" of performance for the task when given a fully labeled dataset. However, if for a given dataset, only RP labels exist, then DSPN is the only method for learning all three tasks.

Considering that DSPN does not use any aspect-level labels, that the effectiveness of DSPN is comparable to supervised models on the ACSA task is a strong empirical validation of the unified sentiment analysis framework in general and the DSPN architecture in particular. [6]

[6]Results for all benchmarking models are presented in the Appendix for completeness.

Model	Rest-14	Rest-15	Rest-16	MAMS
ACSA-G	78.43	71.91	73.76	70.30
JASen	26.62	19.44	23.23	14.74
AX-MABSA	49.68	42.74	36.47	29.74
DSPN	30.01	18.23	24.01	12.79

Table 4: ACSA results on datasets with no RP labels. Benchmark results are from (Kamila et al., 2022). ACSA-G is supervised, JASen and AX-MABSA are weakly supervised, and DSPN is distantly supervised.

5.2 DSPN on No-rating Datasets

We have shown DSPN's effectiveness using two datasets that include both review-level star rating labels (for RP) and aspect-level sentiment annotations (for ACSA). However, a large number of current ACSA datasets do not contain rating data (RP), such as Rest-14 (Pontiki et al., 2014), Rest-15 (Pontiki et al., 2015), Rest-16 (Pontiki et al., 2016) and MAMS (Jiang et al., 2019). In order to enable DSPN to run on such datasets, we use the *aggregate value* of aspect ratings as the training labels instead of the star rating labels given by users. What's more, we can also evaluate our distant supervision model against existing weakly supervised ACSA models.

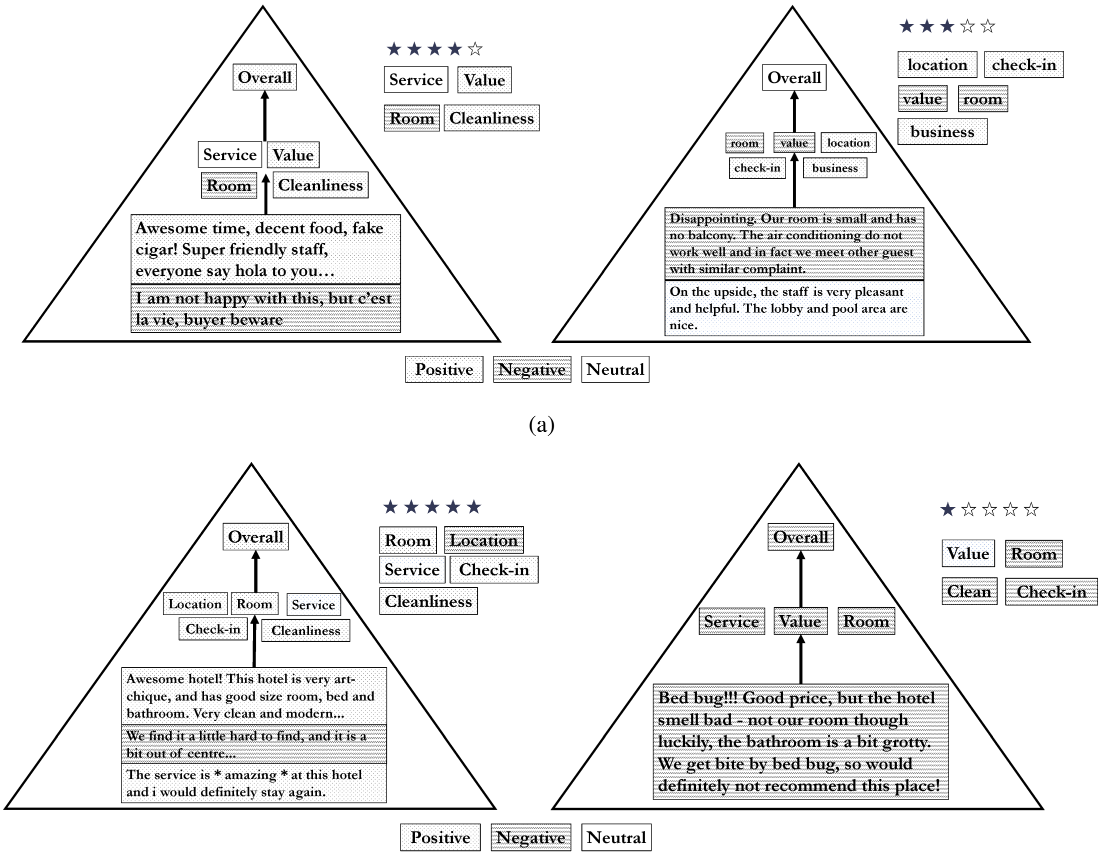
Table 4 shows that DSPN performs comparably to the JASen (Huang et al., 2020) model, which uses a small number of keywords for each aspect-polarity pair as supervision. *This result indicates that RP is not simply an average over ACSA labels, and that the RP labels used by DSPN provide a strong signal.*

Moreover, we conduct a simple additional experiment. In the experiment, we utilize several unsupervised sentiment analysis tools (VADER (Hutto and Gilbert, 2014), TextBlob (Loria, 2018), and Zero-shot text classification (Yin et al., 2019)) to directly generate sentiment labels, which will replace the star rating labels given by users for training. We name the version of DSPN as UPN (U for unsupervised), and here we report the ACSA results of DSPN and UPN on TripDMS (Table 5).

5.3 Quality Analysis

5.3.1 Case Study

In order to visualize and analyze DSPN's performance, we first take two reviews from TripDMS as examples (Figure 3a). For each example, the trained DSPN model takes the review text as in-



(a)

(b)

Figure 3: Case studies of correct predictions (3a) and incorrect predictions (3b). True RP and ACSA labels are outside of the pyramid, DSPN's predictions are within the pyramid. For space, we show a portion of the review.

Model	Label Source	Performance
DSPN	Star ratings	0.532
UPN	TextBlob	0.502
UPN	VADER	0.511
UPN	Zero-shot	0.533

Table 5: DSPN results compared to a fully unsupervised pyramid network (UPN).

put, and first outputs word-level sentiment predictions. Then, DSPN (i) identifies aspect keywords via word attention calculation; (ii) obtains the aspect importance; (iii) calculates aspect-level sentiment through the sentiments of their key words, and lastly (iv) combines aspect sentiment with aspect importance to predict the final review-level sentiment ("Overall" in Figure 3).

For case 1 in Figure 3a, DSPN correctly labels the review as positive, and also correctly identifies and labels the *Service*, *Value*, *Room*, and *Cleanliness* aspects with no aspect-level annotations. For case 2, DSPN gives correct predictions on word-, aspect-, and review-level sentiments.

5.3.2 Error Analysis

To exemplify errors in DSPN, we examine two examples of error cases from TripDMS in Figure 3b. We find that DSPN is sometimes influenced by extreme star rating labels. For example, for case 1 in Figure 3b, DSPN gives correct word-level sentiments, but tends to give positive prediction at aspect level due to the overall 5-star rating. Similarly for case 2, DSPN gives negative predictions on all three levels due to 1-star rating. This is to be expected as DSPN's only supervision is star rating labels.

6 Related Work

Sentiment analysis is a widely-studied area of NLP across ACD, ACSA, and RP. Several recent reviews provide comprehensive overviews of the state of the field (Liu and Zhang, 2012; Schouten and Frasincar, 2015). Below we describe the most relevant work.

6.1 Aspect-Category Detection

Extant ACD methods are either rule-based, supervised, or unsupervised. Rule-based methods (e.g., Hai et al., 2011; Schouten et al., 2014) heavily depend on manually defined rules and domain knowledge. Supervised methods (e.g., Toh and Su, 2016; Xue et al., 2017) require that each review is labeled with a subset of the predefined aspect categories. Unsupervised models (e.g., Titov and McDonald, 2008; Brody and Elhadad, 2010; Zhao et al., 2010) typically extract aspects by implicitly finding word co-occurrence patterns in the corpus. The ABAE model (He et al., 2017) uses an autoencoder-style network to extract aspects in a fully unsupervised manner, and is the foundation of our Module 1. Recently, Tulkens and van Cranenburgh (2020) proposed a simple aspect detection model that utilize a POS tagger and word embeddings, with a contrastive attention mechanism that outperforms more complex models. In our work, we utilize a novel aspect-attention mechanism to use ACD model outputs as part of the ACSA task.

6.2 Aspect-Category Sentiment Analysis

Most ACSA methods in the literature are supervised (Schouten and Frasincar, 2015; Li et al., 2020c; Liu et al., 2021) and require costly and time-consuming data annotation at the aspect level. Unsupervised LDA-based ACSA models (e.g., Zhao et al., 2010; Xu et al., 2012; García-Pablos et al., 2018) often rely on external resources such as part-of-speech tagging and sentiment word lexicons. These LDA-based models can suffer from a topic resembling problem (Huang et al., 2020). To address this, Huang et al. (2020) proposed a weakly-supervised approach that can learn a joint aspect-sentiment topic embedding. However, this method can only be applied to documents with a single annotated aspect, which degenerates the task to RP. Recently, Kamila et al. (2022) proposed an exteremely weakly supervised ACSA model, AX-MABSA, which gives a strong performance on ACSA without using any labelled data. However, the model relies on a single word for each class, making it difficult to select a representative word for the "neutral" class. In this work, we propose a distantly supervised pyramid network to efficiently perform ACSA task with only star rating labels.

6.3 Rating Prediction

RP is modeled as a multi-class classification task, and is well-studied (e.g., Ganu et al., 2009; Li et al., 2011; Liu and Zhang, 2012; Chen et al., 2018). There is also a significant body of literature on semi-supervised and unsupervised approaches to RP (Pugoy and Kao, 2021; Yao et al., 2017; Boteanu and Chernova, 2013).

6.4 Multi-Task Sentiment Analysis

There has been work in jointly learning ACSA and RP (Bu et al., 2021), leveraging RP information for ACSA (Yin et al., 2017; Li et al., 2018; He et al., 2018), and leveraging ACSA information for RP (Cheng et al., 2018; Wu et al., 2019). Prior work on document-level multi-aspect sentiment classification predicted user's ratings on different aspects of products or services (Yin et al., 2017; Li et al., 2018). By adding user information and star rating labels, the methods give strong performances. In each of these cases, the extra information augments the task labels, improving performance at the cost of efficiency. Other works (Bu et al., 2021; Fei et al., 2022) have done ACD and ACSA via joint learning; these methods require costly and time-consuming aspect-level data annotation, hindering efficiency. Schmitt et al. (2018) proposed joint learning models to simultaneously perform ACD and ACSA in an end-to-end manner. To the best of our knowledge, this is the first work to learn all three tasks simultaneously using a single task source for supervision.

7 Conclusion

In this paper, we introduce *unified sentiment analysis* to connect three important sentiment analysis tasks. To perform the task, we propose a Distantly Supervised Pyramid Network (DSPN) that shows significant efficiency advantage by only using star rating labels for training. Experiments conducted on two multi-aspect datasets demonstrate the good performance of DSPN on RP and ACD as well as the effectiveness with only RP labels as supervision.

DSPN's performance demonstrates the validity of considering sentiment analysis holistically and this empirical evidence shows that it is possible to use signal from a single task (RP) to efficiently and effectively learn three tasks. We hope this work spurs research on leveraging one label source for efficient learning for multiple tasks.

8 Limitations

There are several limitations to this work that shed light on promising avenues for future research.

Aspect and Review Sentiment Mismatch DSPN uses star rating labels for training. However, the user rating may not be consistent with the overall sentiment of the review text, thus generating the noise of distant labels. This is because the user may not have written all the aspects in the review, or the user's sentiment is heavily dominated by a certain aspect. It is not obvious how to model this within DSPN. While attention should address this to an extent, future work could consider methods from label noise research.

Evaluation Data Availability Another limitation has to do with data availability. There are a number of ACSA and RP datasets separately in the literature. However, it is very rare that datasets support unified sentiment analysis, i.e. they include both aspect-level sentiments and review-level star rating labels. Therefore, we were restricted to TripDMS and ASAP as the only two datasets available for our main evaluation. However, we feel that by demonstrating the capability of DSPN on one English dataset and one Chinese dataset helps demonstrate the generalization capability of the model. We encourage future work on the creation of more datasets with both ACSA and RP labels to drive further research in unified sentiment analysis.

Unsupervised ACD A final limitation concerns ACD. We compare to ABAE as our ACD module is unsupervised. However, there are supervised ACD methods in the literature, including some that do ACD and ACSA jointly. Future work can investigate injecting further supervision into the unified sentiment analysis task for ACD and/or ACSA.

9 Ethics Statement

The authors state that this research was conducted in accordance with the ACL Code of Ethics. We note that our experiments are on two controlled datasets and do not provide any guarantees of effectiveness or performance on out-of-domain data. In addition, although we experiment with English and Chineses languages, we cannot make claims as to how our research performs on other languages, including low-resource languages.

References

Mohamed Aly and Amir Atiya. 2013. Labr: A large scale arabic book reviews dataset. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 494–498.

Adrian Boteanu and Sonia Chernova. 2013. Unsupervised rating prediction based on local and global semantic models. In *2013 AAAI Fall Symposium Series*.

Samuel Brody and Noemie Elhadad. 2010. An unsupervised aspect-sentiment model for online reviews. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*, pages 804–812.

Jiahao Bu, Lei Ren, Shuang Zheng, Yang Yang, Jingang Wang, Fuzheng Zhang, and Wei Wu. 2021. Asap: A chinese review dataset towards aspect category sentiment analysis and rating prediction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2069–2079.

Siva Uday Sampreeth Chebolu, Franck Dernoncourt, Nedim Lipka, and Thamar Solorio. 2023. Survey of aspect-based sentiment analysis datasets. In *IJCNLP-AACL 2023*.

Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 World Wide Web Conference*, pages 1583–1592.

Zhiyong Cheng, Ying Ding, Lei Zhu, and Mohan Kankanhalli. 2018. Aspect-aware latent factor model: Rating prediction with ratings and reviews. In *Proceedings of the 2018 world wide web conference*, pages 639–648.

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

Hao Fei, Jingye Li, Yafeng Ren, Meishan Zhang, and Donghong Ji. 2022. Making decision like human: Joint aspect category sentiment analysis and rating prediction with fine-to-coarse reasoning. In *Proceedings of the ACM Web Conference 2022*, pages 3042–3051.

Gayatree Ganu, Noémie Elhadad, and Amélie Marian. 2009. Beyond the stars: Improving rating predictions using review text content. In *International Workshop on the Web and Databases*.

Aitor García-Pablos, Montse Cuadros, and German Rigau. 2018. W2vlda: almost unsupervised system for aspect based sentiment analysis. *Expert Systems with Applications*, 91:127–137.

Erfan Ghadery, Sajad Movahedi, Masoud Jalili Sabet, Heshaam Faili, and Azadeh Shakery. 2019. Licd: A language-independent approach for aspect category detection. In *European Conference on Information Retrieval*, pages 575–589.

Zhen Hai, Kuiyu Chang, and Jung-jae Kim. 2011. Implicit feature identification via co-occurrence association rule mining. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 393–404.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2017. An unsupervised neural attention model for aspect extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 388–397.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018. Exploiting document knowledge for aspect-level sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 579–585.

Jiaxin Huang, Yu Meng, Fang Guo, Heng Ji, and Jiawei Han. 2020. Weakly-supervised aspect-based sentiment analysis via joint aspect-sentiment topic embedding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6989–6999.

Clayton Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international AAAI conference on web and social media*, volume 8, pages 216–225.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6280–6285.

Sabyasachi Kamila, Walid Magdy, Sourav Dutta, and MingXue Wang. 2022. Ax-mabsa: A framework for extremely weakly supervised multi-label aspect based sentiment analysis. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6136–6147.

Avinash Kumar, Pranjal Gupta, Nisarg Kotak, Raghunathan Balan, Lalita Bhanu Murthy Neti, and Aruna Malapati. 2022. Barlat: A nearly unsupervised approach for aspect category detection. *Neural Processing Letters*, 54(5):4495–4519.

Fangtao Li, Nathan Nan Liu, Hongwei Jin, Kai Zhao, Qiang Yang, and Xiaoyan Zhu. 2011. Incorporating reviewer and product information for review rating prediction. In *Twenty-second international joint conference on artificial intelligence*.

Junjie Li, Haitong Yang, and Chengqing Zong. 2018. Document-level multi-aspect sentiment classification by jointly modeling users, aspects, and overall ratings. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 925–936.

Qiudan Li, Daniel Dajun Zeng, David Jingjun Xu, Ruoran Liu, and Riheng Yao. 2020a. Understanding and predicting users' rating behavior: A cognitive perspective. *INFORMS Journal on Computing*, 32(4):996–1011.

Yuncong Li, Zhe Yang, Cunxiang Yin, Xu Pan, Lunan Cui, Qiang Huang, and Ting Wei. 2020b. A joint model for aspect-category sentiment analysis with shared sentiment prediction layer. In *China National Conference on Chinese Computational Linguistics*, pages 388–400.

Yuncong Li, Cunxiang Yin, Sheng-hua Zhong, and Xu Pan. 2020c. Multi-instance multi-label learning networks for aspect-category sentiment analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3550–3560.

Ming Liao, Jing Li, Haisong Zhang, Lingzhi Wang, Xixin Wu, and Kam-Fai Wong. 2019. Coupling global and local context for unsupervised aspect extraction. 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 4579–4589.

Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. In *Mining text data*, pages 415–463. Springer.

Jian Liu, Zhiyang Teng, Leyang Cui, Hanmeng Liu, and Yue Zhang. 2021. Solving aspect category sentiment analysis as a text generation task. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4406–4416.

Steven Loria. 2018. textblob documentation. *Release 0.15*, 2.

Ling Luo, Xiang Ao, Yan Song, Jinyao Li, Xiaopeng Yang, Qing He, and Dong Yu. 2019. Unsupervised neural aspect extraction with sememes. In *IJCAI*, pages 5123–5129.

Andrius Mudinas, Dell Zhang, and Mark Levene. 2012. Combining lexicon and learning based approaches for concept-level sentiment analysis. In *Proceedings of the first international workshop on issues of sentiment discovery and opinion mining*, pages 1–8.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *International workshop on semantic evaluation*, pages 19–30.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495.

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

Reinald Adrian Pugoy and Hung-Yu Kao. 2021. Unsupervised extractive summarization-based representations for accurate and explainable collaborative filtering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2981–2990.

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016. Insight-1 at semeval-2016 task 5: Deep learning for multilingual aspect-based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 330–336.

Martin Schmitt, Simon Steinheber, Konrad Schreiber, and Benjamin Roth. 2018. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1114.

Kim Schouten and Flavius Frasincar. 2015. Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, 28(3):813–830.

Kim Schouten, Flavius Frasincar, and Franciska De Jong. 2014. Commit-p1wp3: A co-occurrence based approach to aspect-level sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 203–207.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *Chinese Computational Linguistics: 18th China National Conference, CCL 2019, Kunming, China, October 18–20, 2019, Proceedings 18*, pages 194–206. Springer.

Ivan Titov and Ryan McDonald. 2008. Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th international conference on World Wide Web*, pages 111–120.

Zhiqiang Toh and Jian Su. 2016. Nlangp at semeval-2016 task 5: Improving aspect based sentiment analysis using neural network features. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 282–288.

Stéphan Tulkens and Andreas van Cranenburgh. 2020. Embarrassingly simple unsupervised aspect extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3182–3187.

Cong Wan, Shan Jiang, Cong Wang, Ying Yuan, and Cuirong Wang. 2020. A novel sentence embedding based topic detection method for microblogs. *IEEE Access*, 8:202980–202992.

Hongning Wang, Yue Lu, and Chengxiang Zhai. 2010. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 783–792.

Chuhan Wu, Fangzhao Wu, Junxin Liu, Yongfeng Huang, and Xing Xie. 2019. Arp: Aspect-aware neural review rating prediction. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2169–2172.

Xueke Xu, Songbo Tan, Yue Liu, Xueqi Cheng, and Zheng Lin. 2012. Towards jointly extracting aspects and aspect-specific sentiment knowledge. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1895–1899.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523.

Wei Xue, Wubai Zhou, Tao Li, and Qing Wang. 2017. Mtna: a neural multi-task model for aspect category classification and aspect term extraction on restaurant reviews. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 151–156.

Huaxiu Yao, Min Nie, Han Su, Hu Xia, and Defu Lian. 2017. Predicting academic performance via semi-supervised learning with constructed campus social network. In *International Conference on Database Systems for Advanced Applications*, pages 597–609.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019*, pages 3914–3923. Association for Computational Linguistics.

Yichun Yin, Yangqiu Song, and Ming Zhang. 2017. Document-level multi-aspect sentiment classification as machine comprehension. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2044–2054.

Zelin Zhang, Kejia Yang, Jonathan Z Zhang, and Robert W Palmatier. 2023. Uncovering synergy and dysergy in consumer reviews: A machine learning approach. *Management Science*, 69(4):2339–2360.

Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, page 56–65.

A Notation

For clarity and consistency, we provide a comprehensive description of the notation we use in this article (Table 6).

Variable	Description	Dimension
d_w	Embedding dimension	$\mathbb{R}^{768}$
$\mathbf{R}$	Reviews in our dataset	-
R_i	i-th review consisted of a sequence of word tokens	$\mathbb{R}^{n \times d_w}$
n	Number of word tokens in R_i	$\mathbb{R}^{100}$
$t_i^{(j)}$	j-th word in R_i	$\mathbb{R}^{1 \times d_w}$
A	Predefined aspect categories	$\mathbb{R}^{N}$
A_{R_i}	The set of aspects present in R_i	$\mathbb{R}^{K}(K \leq N)$
$A_{R_i}^{(j)}$	j-th aspect in A_{R_i}	$\mathbb{R}^{1 \times d_w}$
$y_{A_{R_i}^{(j)}}$	Sentiment polarity of $A_{R_i}^{(j)}$	$\mathbb{R}^{3}$
y_{R_i}	Star rating of R_i	$\mathbb{R}^{3}$
M	Model	-
$\hat{A}_{R_i}$	Prediction of A_{R_i}	-
$\hat{y}_{A_{R_i}^{(j)}}$	Prediction of $y_{A_{R_i}^{(j)}}$	$\mathbb{R}^{3}$
$\hat{y}_{R_i}$	Prediction of y_{R_i}	$\mathbb{R}^{3}$
$\mathbf{X}_i$	Input sequence	$\mathbb{R}^{n \times d_w}$
$\mathbf{z}_i$	Sentence embedding of $\mathbf{X}_i$ (pooler_output of BERT)	$\mathbb{R}^{n \times d_w}$
$\mathbf{T}$	Aspect embedding matrix	$\mathbb{R}^{N \times d_w}$
$\mathbf{T}_k$	Embedding of k-th aspect	$\mathbb{R}^{1 \times d_w}$
$\mathbf{r}_i$	Reconstructed sentence embedding	$\mathbb{R}^{1 \times d_w}$
$\mathbf{p}_i$	Weight vectors of K aspect embeddings (aspect importance)	$\mathbb{R}^{N \times d_w}$
$L(\theta_{ACD})$	Loss function of ACD task (Module 1)	-
λ_{ACD}	Weight of regularization term	-
$U(\theta)$	Regularization term	-
n_j	Each negative sample	$\mathbb{R}^{1 \times d_w}$
$\mathbf{h}_i^{(j)}$	hidden state of j-th word (last_hidden_state of BERT)	$\mathbb{R}^{1 \times d_w}$
$\mathbf{w}_i^{(j)}$	Sentiment prediction vector of j-th word	$\mathbb{R}^{1 \times d_w \times 3}$
$d_k^{(j)}$	Distance between j-th word and k-th aspect	-
$a_k^{(j)}$	Attention weight of j-th word towards k-th aspect	-
S_a^k	Prediction of aspect-level sentiment	$\mathbb{R}^{N \times 3}$
S_r	Prediction of review-level sentiment	$\mathbb{R}^{3}$
S_a	Matrix concatenation of S_a^k	$\mathbb{R}^{K \times 3}$
S_{gold}	True review-level sentiment (star rating labels)	$\mathbb{R}^{3}$
$L(\theta_{RP})$	Loss function of RP task (Module 2)	-
λ	Weight of $L(\theta_{ACD})$	-
$L(\theta)$	Overall loss function	-

Table 6: Description of variables in our formulation.

B Additional Error Analyses

For a more comprehensive analysis, we look into the DSPN errors in more detail. Due to the imbalanced label distribution in the original data (Table 1), DSPN tends to predict more extreme sentiment polarities (positive or negative) on TripDMS, and tends to predict positive sentiments on ASAP. The confusion matrices for aspect-level sentiments predicted by DSPN are consistent with the distribution of the original data (Tables 7a and 7b).

True \ Pred	Neg	Neu	Pos	Total
Neg	3,511	982	944	5,437
Neu	1,672	884	1,799	4,355
Pos	1,962	1,560	4,480	8,002
Total	7,145	3,426	7,223	17,794

(a) Confusion Matrix of DSPN on TripDMS

True \ Pred	Neg	Neu	Pos	Total
Neg	589	521	1,293	2,403
Neu	260	712	2,757	3,729
Pos	127	760	9,257	10,144
Total	976	1,993	13,307	16,276

(b) Confusion Matrix of DSPN on ASAP

Table 7: DSPN confusion matrices.

C Budget Constraint Experiment

For a more direct comparison between DSPN and the supervised ACSA models, we designed a budget-constraining experiment. Specifically, we randomly selected ACSA labels for TripDMS and ASAP so that the supervised models have the same training set size as DSPN.

In this setting, DSPN's performance is closer to the supervised models' performance (Table 8). In particular, DSPN outperforms both End2end-LTSM and End2end-CNN on ASAP. Overall, the supervised models still outperform DSPN, but this is to be expected given that the labels used for training are ACSA labels. DSPN is trained to perform RP, but is also able to perform ACSA in a way that is comparable to these supervised models under the same budget constraint.

Model	TripDMS	ASAP
End2end-LSTM	0.542	0.651
End2end-CNN	0.536	0.649
GCAE	0.540	0.701
AC-MIMLLN	0.614	0.758
AC-MIMLLN-BERT	0.639	0.766
ACSA-Generation	0.602	0.758
DSPN (Ours)	0.532	0.654

Table 8: ACSA results when all models are trained with the same amount of data.

D Benchmarking Details

- End2end-LSTM/CNN: The method uses an end-to-end network for ACSA. It can simultaneously perform aspect category detection and aspect-level sentiment analysis.
- GCAE: This method is a simple and effective supervised model based on convolutional neural networks and gating mechanisms.
- AC-MIMLLN: It utilized multi-instance multi-label learning for ACSA and found that the aspect-level sentiment can be regarded as an aggregation of the word-level sentiments indicating the aspect.
- AC-MIMLLN-BERT: It replaces the embedding layer for ACSA and the multi-layer Bi-LSTM in AC-MIMLLN with the BERT.
- ACSA-generation: This is the first method that solve ACSA task with natural language generation paradigm, and achieved good results.
- BERT-Feat: BERT as features.
- BERT-FiT: BERT + Fine-Tuning as features.
- BERT-ITPT-FiT: BERT + withIn-Task Pre-Training + Fine-Tuning as features.

E On Sentence Reconstruction for ACD

Sentence reconstruction is standard for unsupervised ACD task. Table 9 shows that sentence reconstruction is widely used and effective for this task.

F Additional Benchmarking

Tables 10, 11, and 12 present the comprehensive results of our benchmarking. We selected our pipeline models from these benchmarks based on predictive performance and efficiency.

Reference	Mechanism	Datasets	Performance
(He et al., 2017)	sentence reconstruction	CitySearch, BeerAdvocate	SOTA
(Kumar et al., 2022)	seed words + sentence reconstruction + adversarial training	CitySearch, Laptop	SOTA
(García-Pablos et al., 2018)	topic model	CitySearch	Competitive results
(Liao et al., 2019)	multiple context modeling + representation reconstruction	SemEval 14, 15, 16	SOTA
(Luo et al., 2019)	lexical semantic enhancing + sentence reconstruction	CitySearch, BeerAdvocate	SOTA
(Wan et al., 2020)	sentence embedding + sentence reconstruction	Sina microblog	Effective results
This paper	sentence reconstruction + multi-task learning + distant supervision	ASAP, TripDMS	Comparable results

Table 9: Mechanisms Used in Unsupervised ACD Task

	TripDMS			ASAP		
Model	**Accuracy**	**Params**	**Train Time**	**Accuracy**	**Params**	**Train Time**
DSPN	70.5	5.28M	12min	78.5	6.1M	13min
DSPN-BERT	72.5	102.92M	95min	81.3	111M	88min
BERT-Feat	71.4	80.15M	35min	79.2	80.8M	42min
BERT-FiT	72.2	81M	37min	81	81.25M	30min
BERT-ITPT-FiT	72.4	82.7M	102min	80.3	91M	110min

Table 10: Comprehensive RP Results

	TripDMS			ASAP		
Model	**F1**	**Params**	**Train Time**	**F1**	**Params**	**Train Time**
DSPN	92.7	5.28M	12min	78.6	6.1M	13min
DSPN-BERT	92.7	102.92M	95min	79.4	111M	88min
ABAE	91.2	3.1M	15min	79.4	3.1M	15min
ABAE-BERT	92.3	91.2M	40min	80.1	97.5M	42min

Table 11: Comprehensive ACD Results

	TripDMS			ASAP		
Model	**Accuracy**	**Params**	**Train Time**	**Accuracy**	**Params**	**Train Time**
DSPN	51.4	5.28M	12min	64.4	6.1M	13min
DSPN-BERT	53.2	102.92M	95min	65.4	111M	88min
End2end-LSTM	57.4	5.3M	8min	66.1	6.22M	8min
End2end-CNN	57.9	5.12M	7min	65.2	5.32M	7min
GCAE	55.1	4.23M	5min	70.3	4.4M	6min
AC-MIMLLN	62.1	31M	50min	76	31.2M	50min
AC-MIMLLN-BERT	64.3	105M	55min	77.2	107,2M	55min
ACSA-generation	64.1	142M	208min	76.1	145.18M	210min

Table 12: Comprehensive ACSA Results