# Single-Pass, Depth-Selective Reading for Multi-Aspect Sentiment Analysis

**Yan Xia**[1,2*] **Zhuangzhuang Pan**[1*] **Amirrudin Kamsin**[1] **Chee Seng Chan**[1,3†]
[1]Universiti Malaya, Malaysia
[2]Suzhou University of Technology, China
[3]VinUniversity, Vietnam
{23072126; 23078403}@siswa.um.edu.my, {amir; cs.chan}@um.edu.my

## Abstract

Aspect-Term Sentiment Analysis (ATSA) in multi-aspect sentences faces a fundamental tradeoff between efficiency and expressiveness. Existing models either re-encode the sentence for each aspect or rely on static use of deep representations, leading to redundant computation and limited adaptivity. We argue that Transformer depth is a costly, queryable resource, and propose **DABS**, a single-pass inference framework that encodes each sentence once to construct a reusable, depth-ordered substrate. Each aspect then queries this shared representation to selectively read relevant tokens and abstraction levels, without re-encoding. This decouples shared sentence encoding from lightweight, aspect-conditioned readout. Experiments on four ATSA benchmarks show that DABS achieves competitive performance while reducing end-to-end computation by up to 60% in multi-aspect settings ($M \geq 2$). Further analyses indicate that adaptive depth querying is most beneficial for linguistically complex cases such as negation and contrast. Code is publicly available at https://github.com/panzhzh/acl-dabs.

## 1 Introduction

Aspect-Term Sentiment Analysis (ATSA) predicts the polarity of a given aspect term within a sentence (Zhang et al., 2023a; Wang et al., 2023). Although pretrained Transformer-based systems are strong (Ma et al., 2023; Cabello and Akujuobi, 2024; Ding et al., 2024), multi-aspect sentences expose a mismatch between how the task is structured and how computation is typically performed (Seo et al., 2024; Jin et al., 2025). In a sentence like "The *food* was excellent but the *service* was terrible, and the *atmosphere* felt cramped", the aspects behave like parallel queries over largely shared context. Yet, many existing methods either (i) process each aspect separately, incurring linear cost in the number of aspects (Zheng and Li, 2024), or (ii) collapse representations in a uniform way that can dilute aspect-specific evidence (Wagner and Foster, 2023; Zhu et al., 2024). Consequently, researchers face a binary choice between the computational overhead of per-aspect re-encoding and the information loss inherent in static, depth-compressed representations (Bao et al., 2023a; Lv et al., 2023).

We argue that multi-aspect ATSA is better viewed as aspect-conditioned querying over a shared sentence representation, where each aspect may benefit from a different amount of representational depth. Deeper representations are often helpful for compositional phenomena such as negation, contrast, and discourse reversal, while other aspects can be resolved using shallower or intermediate evidence (Chai et al., 2023; Petty et al., 2024; Xu et al., 2024). Treating all aspects as equally "*deep*" can therefore obscure how linguistic difficulty relates to abstraction level. This motivates viewing Transformer depth as a *queryable resource*, rather than as a uniform feature pool (Bae et al., 2023; Elhoushi et al., 2024).

Under this view, each aspect navigates a tradeoff between efficiency and semantic richness. In this framing, *deeper layers provide more nuanced information, but the model should limit its depth of traversal to the specific requirements of the aspect.*

To this end, we propose **DABS** (**D**epth-Ordered **A**ggregation and **B**udget-Aware **S**election), a *single-pass* framework that separates heavy sentence-level encoding from lightweight, aspect-specific reads. DABS first constructs a reusable *depth substrate* that functions as an aspect-agnostic bank of representations exposing multiple abstraction levels from a single encoder pass. At inference time, each aspect adaptively queries the shared substrate to resolve both *where* relevant evidence lies

[*]equal contribution; authors are listed alphabetically by first name.
[†]Corresponding author (cs.chan@um.edu.my).



*Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14638–14656
July 2-7, 2026 

in the sentence and *how much abstraction* is required to interpret it, without re-encoding the input.

Together, this formulation enables budget-aware selection, allowing representational capacity across depth to be allocated adaptively while amortizing the encoder computation. Our subsequent analysis, conducted via controlled depth masking, demonstrates that prediction quality is sensitive to specific depth regions, particularly in linguistically challenging cases, such as negation and contrast.

Overall, our contributions are:

- We cast multi-aspect ATSA as aspect-conditioned queries over a shared depth substrate, explicitly linking marginal compute to aspect-specific expressiveness.
- We propose DABS, a single-pass architecture that constructs a reusable depth substrate per sentence, enabling aspects to query both evidence location (tokens) and abstraction level (depth) without re-encoding the input.
- We provide controlled depth-masking analyses demonstrating that performance depends non-uniformly on specific depth regions, supporting the view that depth selection plays a functional role in sentiment reasoning rather than serving as a redundant feature ensemble.
- Across four benchmarks, DABS achieves competitive results. In multi-aspect settings, it reduces end-to-end computation by up to 60%[1] compared to standard per-aspect encoding baselines, showing that explicit depth allocation can improve efficiency while preserving fine-grained sentiment modeling.

## 2 Related Work

**Structure-aware ATSA.** A substantial body of work injects syntactic structure (typically dependency graphs) to align aspects with opinion expressions and to route information along aspect-centered paths (Bao et al., 2023b; Yin and Zhong, 2024; Zhang et al., 2023b). While these models offer explicit alignment signals, they rely on external parsers that are brittle on noisy text and computationally expensive to run. Furthermore, many structure-aware designs fundamentally rely on aspect-specific graph construction (*e.g.*, graph reweighting or masking), which prevents representation sharing and incurs linear computational costs in multi-aspect settings.

**PLM Fine-Tuning and Hybrid Models.** The dominant paradigm fine-tunes pretrained encoders by concatenating aspect-sentence pairs or injecting aspect markers (Zheng et al., 2024; Gou et al., 2023; Mukherjee et al., 2023). While effective, this approach suffers from two limitations central to our study. First, it treats the sentence as an aspect-dependent object, necessitating a full encoder pass for every aspect (Zhang et al., 2023a; Zheng and Li, 2024). Second, regarding representational depth, these models typically adopt a static usage pattern, either relying solely on the final layer or performing fixed scalar mixing across all layers (He et al., 2025; Ruan et al., 2025). Some works have attempted to complement PLMs with local mechanisms (*e.g.*, CNNs or RNNs) to capture compositional cues (Feng et al., 2023; Wang et al., 2024). However, these are invariably implemented as static architectural layers within a per-aspect pipeline. Unlike DABS, they do not decouple the shared representation from the query, nor do they treat depth as a dynamically allocatable resource.

**Generative and LLM-based ATSA.** Recent approaches cast ATSA as instruction-following generation or few-shot in-context learning (Scaria et al., 2024; Shen et al., 2025; Hellwig et al., 2025; Zheng et al., 2024). While Large Language Models (LLMs) encode rich reasoning patterns, they introduce significant inference latency and lack the fine-grained controllability required for span-level polarity classification. This makes them less aligned with the high-throughput, resource-constrained inference goals of this work (Simmering and Huoviala, 2023; Bodke et al., 2025).

## 3 Methods

### 3.1 Problem Definition

ATSA focuses on predicting sentiment polarities for specified aspect terms. Formally, we consider a tokenized sentence $\mathbf{x} = (x_1, \cdots, x_n)$ together with a set of annotated target aspects $\mathcal{A} = \{a^{(k)}\}_{k=1}^{M}$. Each aspect $a^{(k)}$ is represented by its position interval $[i_k, j_k]$, where $1 \leq i_k \leq j_k \leq n$. Our task is to model the conditional distribution $p_\theta\big(y^{(k)} \mid \mathbf{x},\, a^{(k)}\big)$, where $y^{(k)} \in \{\text{positive}, \text{neutral}, \text{negative}\}$ and $\theta$ denotes the

[1]For single-aspect sentences ($M = 1$), DABS introduces a fixed overhead due to depth-substrate construction. This cost is paid only once per sentence and can be significantly amortized for $M \geq 2$. See Figure 2 and Appendix C for detailed efficiency breakdowns.

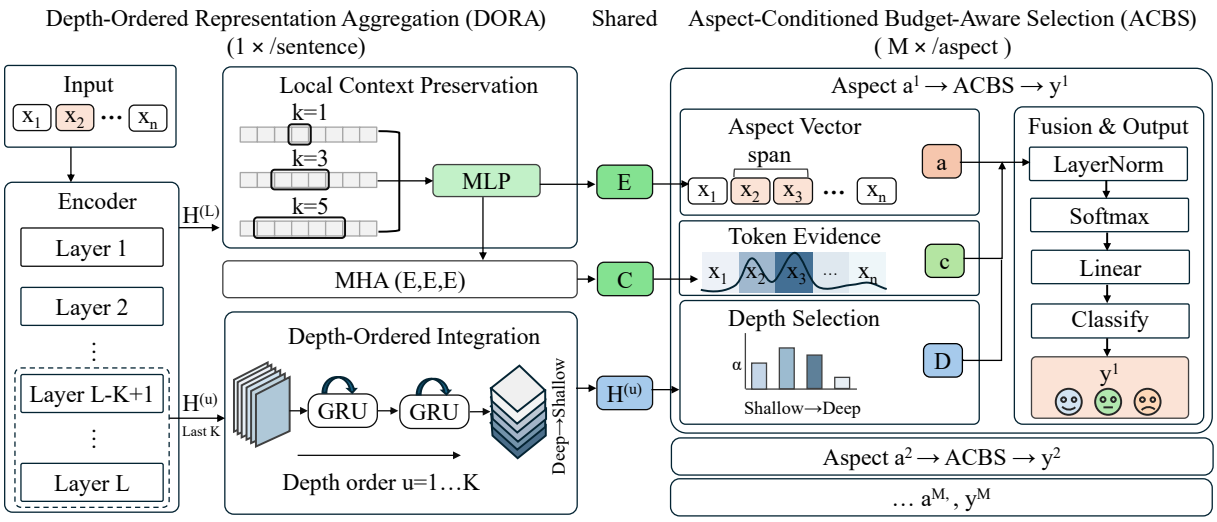


Figure 1: Overview of the proposed DABS framework. DORA constructs a shared depth substrate via a single encoder pass, and ACBS performs aspect-conditioned token localization and budget-aware depth selection.

parameters of both the pretrained encoder and the task-specific components.

Unlike standard approaches that re-encode $\mathbf{x}$ for every $a^{(k)}$, we formulate this as a **query-retrieval problem**: *the sentence acts as a shared depth substrate, and each aspect is a budget-aware query that must independently localize evidence and select the necessary representational depth*.

### 3.2 Overview of Budget-Aware Inference

We view multi-aspect ATSA as a *query problem over shared representations*, rather than as a collection of independent classification instances. In a multi-aspect sentence, all aspects observe the same underlying context, but differ in how much *representational depth* they require to resolve their sentiment. Some aspects can be decided from shallow lexical cues, while others require deeper compositional reasoning involving negation, contrast, or long-range dependency.

Existing approaches do not expose this distinction. They either re-encode the sentence for each aspect, implicitly assuming that every aspect requires full depth, or collapse all layers into a static vector, removing the model's ability to adapt depth usage. As a result, Transformer depth is treated as a free and uniform feature pool, rather than as a constrained and ordered resource.

We propose **DABS**, a single-pass inference framework that makes Transformer depth explicitly queryable under fixed encoder computation. DABS decomposes inference into two lightweight stages: (i) **Depth-Ordered Representation Aggregation (DORA)**, which constructs a reusable, depth-ordered substrate from a single encoder pass, and (ii) **Aspect-Conditioned Budget-Aware Selection (ACBS)**, which performs aspect-specific readout over this substrate. Both stages operate as readout mechanisms over existing representations, rather than adding new iterative reasoning or dynamic depth traversal. Figure 1 summarizes the overall architecture.

### 3.3 Stage I: Constructing a Shared Depth-Ordered Substrate

We first construct a reusable depth substrate via **Depth-Ordered Representation Aggregation (DORA)** that preserves both local compositionality and the multi-level nature of Transformer depth. This aspect-agnostic module constructs the shared depth substrate by reshaping hierarchical encoder hidden states $\{H^{(\ell)}\}_{\ell=1}^{L}$ via two paths.

#### 3.3.1 Preserving Local Sentiment Cues

While Transformers capture long-range dependencies, global mixing can weaken short-range cues such as negation markers adjacent to sentiment expressions (*e.g.*, "*not good*"). To keep these local patterns explicit in the shared substrate, we apply a lightweight local refinement on the final encoder states $H^{(L)}$ using depthwise-separable filters with small kernel sizes $k \in \{1, 3, 5\}$:

$$\tilde{E} = \mathrm{Concat}\big(\{\mathrm{Conv}_k(H^{(L)})\}_{k\in\{1,3,5\}}\big), \quad (1)$$

$$E = \mathrm{LayerNorm}\big(\tilde{E}\,\mathbf{W}_c + H^{(L)}\big). \quad (2)$$

Here, $k \in \{1, 3, 5\}$ provides a small set of receptive fields with minimal overhead.

#### 3.3.2 Integrating Representations Across Depth

Crucially, we must expose the evolution of representations across layers without treating layers as exchangeable. We therefore apply a gated recurrence over the last $K$ layers to form a depth-ordered stack $\{\tilde{H}^{(u)}\}_{u=1}^{K}$. For each token position $t$, the recurrence state is initialized at the first depth step and updated in increasing depth order:

$$\mathbf{s}_{1,t} = H_t^{(L-K+1)}, \tag{3}$$

$$\mathbf{s}_{u,t} = \mathrm{GRUCell}\big(H_t^{(L-K+u)}, \mathbf{s}_{u-1,t}\big), \quad u = 2, \ldots, K. \tag{4}$$

We then form the depth-ordered outputs by residual addition and LayerNorm:

$$\tilde{H}^{(1)} = \mathrm{LayerNorm}\big(\beta\, \mathbf{s}_1 + H^{(L-K+1)}\big), \quad \beta \in \mathbb{R}, \tag{5}$$

$$\tilde{H}^{(u)} = \mathrm{LayerNorm}\big(\mathbf{s}_u + H^{(L-K+u)}\big), \quad u = 2, \ldots, K. \tag{6}$$

We implement this module as **DepthGRU** whereby a GRUCell is applied over the last $K$ layers in increasing depth order, producing a reusable stack $\{\tilde{H}^{(u)}\}_{u=1}^{K}$ for querying. Note that DepthGRU is one concrete instantiation of ordered depth integration, whereby any mechanism that preserves monotonic abstraction flow could serve the same role.

### 3.4 Stage II: Aspect-Conditioned Reading from the Substrate

Given the shared substrate, each aspect performs a lightweight read via **Aspect-Conditioned Budget-Aware Selection (ACBS)**. ACBS performs a two-axis read over the shared depth substrate to determine ***where*** to read (token evidence) and ***how deep*** to read (depth preference).

**Aspect vector.** For an aspect $a^{(k)} = [i_k, j_k]$, we compute an aspect vector by averaging the enhanced representations within the span, $\mathbf{a}^{(k)} = \frac{1}{j_k - i_k + 1} \sum_{t=i_k}^{j_k} E_t$.

**Shared context reorganization.** We compute a reusable contextualized representation once per sentence by applying multi-head attention over $E$, as $C = \mathrm{MHA}(E, E, E)$.

#### 3.4.1 Axis 1: Selecting Relevant Evidence Tokens

Standard token-level Softmax attention imposes competition between tokens. To allow evidence to be accumulated across multiple positions, we employ independent sigmoid gating:

$$w_t = \sigma\big(\mathrm{MLP}([C_t; \mathbf{a}])\big), \quad \mathbf{c} = \frac{\sum_{t=1}^{n} w_t\, C_t}{\sum_{t=1}^{n} w_t + \varepsilon}. \tag{7}$$

#### 3.4.2 Axis 2: Selecting an Appropriate Depth Level

We predict an aspect-conditioned distribution over the depth substrate:

$$\boldsymbol{\alpha} = \mathrm{Softmax}\Big(\tfrac{1}{\tau_\alpha}\, \mathrm{MLP}([\mathbf{a}; \mathrm{pool}(C)])\Big). \tag{8}$$

where $\sum_{u=1}^{K} \alpha_u = 1$, $\mathrm{pool}(C) = \frac{1}{n}\sum_{t=1}^{n} C_t$ denotes mean pooling over the sequence. The corresponding depth summary is $\mathcal{D} = \frac{1}{n}\sum_{t=1}^{n}\sum_{u=1}^{K} \alpha_u\, \tilde{H}_t^{(u)}$.

#### 3.4.3 Combining Evidence and Producing Predictions

Token evidence vector $\mathbf{c}$, depth summary $\mathcal{D}$, and aspect vector $\mathbf{a}$ capture complementary signals. We first normalize them, then compute fusion gates:

$$\begin{aligned} \hat{\mathbf{x}} &= \mathrm{LayerNorm}(\mathbf{x}), \quad \forall \mathbf{x} \in \{\mathbf{c}, \mathcal{D}, \mathbf{a}\}, \\ \mathbf{g} &= \mathrm{Softmax}\Big(\tfrac{1}{\tau_g}\, \mathrm{MLP}([\hat{\mathbf{c}}; \hat{\mathcal{D}}; \hat{\mathbf{a}}])\Big), \\ \mathbf{h} &= g_1 \hat{\mathbf{c}} + g_2 \hat{\mathcal{D}} + g_3 \hat{\mathbf{a}}. \end{aligned} \tag{9}$$

We obtain logits $\mathbf{z} \leftarrow \mathrm{Linear}(\mathbf{h})$ and $p_\theta \leftarrow \mathrm{Softmax}(\mathbf{z})$.

### 3.5 Training Objectives

We train end-to-end using a cross-entropy classification loss $\mathcal{L}_{\text{cls}}$. We further apply three regularizers (described next) to stabilize selection and discourage degenerate solutions:

#### 3.5.1 Sparsity Regularization ($\mathcal{R}_{\text{sparse}}$).

To encourage the model to focus only on the most informative sentiment cues and filter out irrelevant background noise, we impose a sparsity penalty. This regularizer minimizes the average activation of the token selection gates: $\mathcal{R}_{\text{sparse}} = \frac{1}{n}\sum_{t=1}^{n} w_t$, where $n$ is the sequence length and $w_t \in [0, 1]$ represents the learned importance weight for the $t$-th token. By minimizing $\mathcal{R}_{\text{sparse}}$, we force the model to be selective, preventing the trivial solution where all tokens are retained (*i.e.*, $w_t \approx 1$).

#### 3.5.2 Span masking ($\mathcal{R}_{\text{mask}}$).

We discourage the token selector from relying on tokens inside the aspect span itself. Let $\boldsymbol{m} \in \{0,1\}^n$ be a binary mask such that $m_t = 1$ iff $i_k \le t \le j_k$. We penalize gate activations on the span, $\mathcal{R}_{\text{mask}} = \text{BCE}(w \odot \boldsymbol{m}, \mathbf{0})$, where $\text{BCE}(\cdot,\cdot)$ denotes elementwise binary cross-entropy averaged over positions. Equivalently, $\|w \odot \boldsymbol{m}\|_1$ yields the same effect, while remaining neutral outside the span.

#### 3.5.3 Fusion-gate Entropy ($\mathcal{R}_{\text{gate}}$).

To discourage premature collapse of the fusion module onto a single information source (Context, Layer, or Aspect), we maximize the entropy of the gating distribution, $\mathcal{R}_{\text{gate}} = \sum_{i=1}^{3} g_i \log g_i$, where $g_i$ corresponds to the normalized gating weight for the $i$-th branch (context, layer, and aspect, respectively). Minimizing $\mathcal{R}_{\text{gate}}$ penalizes low-entropy (overly peaked) distributions, thereby discouraging premature collapse onto a single branch.

As a summary, the total objective is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}} + \lambda_s \mathcal{R}_{\text{sparse}} + \lambda_m \mathcal{R}_{\text{mask}} + \lambda_{\text{ent}} \mathcal{R}_{\text{gate}}. \tag{10}$$

### 3.6 Computational Amortization

The efficiency of DABS stems from decoupling the shared depth substrate from per-aspect queries. In sentences with $M$ aspects, the complexity scales:

$$\text{Cost} \approx \underbrace{\mathcal{O}(\text{Encoder} + \text{DORA})}_{\text{Fixed Cost (Paid Once)}} + M \cdot \underbrace{\mathcal{O}(\text{Selection})}_{\text{Lightweight per-aspect cost}}. \tag{11}$$

This amortization is most relevant when $M > 1$ or when compute is constrained, where repeated re-encoding of the full encoder per-aspect is costly.

## 4 Experiment

### 4.1 Setup and Baselines

We evaluate DABS on four standard ATSA benchmarks: SemEval-2014 Laptop (Lap14) and Restaurant (Rest14), SemEval-2015 Restaurant (Rest15), and SemEval-2016 Restaurant (Rest16) (Pontiki et al., 2014, 2015, 2016). Detailed statistics are provided in Appendix A. We report Accuracy (Acc) and Macro-F1 (MF1). We follow the official SemEval train/test protocol and report results on the official test split.

Table 1: Sentence-level aspect multiplicity statistics on the official test splits. Avg $M$ denotes the average number of aspects per sentence, together with the proportions of sentences with $M{=}1$ and $M{>}1$.

| Dataset | Avg $M$ | $P(M{=}1)$ | $P(M{>}1)$ | $P(M{=}2)$ | $P(M{>}2)$ |
|---|---|---|---|---|---|
| Lap14 | 1.55 | 63.0% | 37.0% | 24.8% | 12.2% |
| Rest14 | 1.87 | 47.5% | 52.5% | 32.0% | 20.5% |
| Rest15 | 1.49 | 63.6% | 36.4% | 26.4% | 10.0% |
| Rest16 | 1.55 | 64.7% | 35.3% | 25.5% | 9.8% |

**Aspect multiplicity statistics.** Since the efficiency gain of DABS is realized through amortization across aspect queries, we report sentence-level aspect multiplicity on the official test splits in Table 1. This contextualizes the practical relevance of the multi-aspect setting targeted by DABS on standard ATSA benchmarks.

**Baselines.** Our implementation fine-tunes DeBERTa-v3-base (He et al., 2023). We compare against three baseline families: (i) **Structure-aware methods** utilizing external graphs (KGAN-BERT (Zhong et al., 2023), ASH-GAT (Ouyang et al., 2024), DSSK-GAN-BERT (Liu et al., 2024), DC-GCN (Sun et al., 2025), CABiLSTM-BERT (He et al., 2025)); (ii) **Fine-tuning methods** (PConvRoBERTa (Feng et al., 2023), ITGCN (Shi et al., 2024), DeBERTa+RCL (Jian et al., 2024), Flan-T5-base+Syn-Chain (Fan et al., 2025)); and (iii) **LLMs-based** (Llama-3, Qwen3, GPT-3.5) which we evaluated in a 5-shot in-context learning setup. Full hyperparameter details and baseline descriptions are available in Appendix B.

### 4.2 Quantitative Results

Table 2 reports the main results. DABS achieves competitive or better Acc/MF1 across all four datasets, with clear gains on Lap14 and Rest16. Compared to structure-aware baselines, DABS remains competitive without requiring external parsers, and it also outperforms strong fine-tuning baselines in several settings. While LLMs show general reasoning ability, 5-shot prompting is less effective for span-level polarity attribution whereby DABS exceeds the best 5-shot LLM results by large margins (*e.g.*, $>$ 10 pp on Rest16). Additional breakdowns are provided in Appendix C.

To verify that these gains are robust, we conduct paired statistical tests against an encoder-only baseline. Table 4 shows that DABS yields consistent significant improvements. For instance, $\Delta$ MF1 = 6.53 pp on Lap14 and 8.62 pp on Rest16, with

Table 2: Main results on four ATSA benchmarks (Acc/MF1). DABS reports *mean*±*std* over 3 seeds. LLM results are from our 5-shot evaluation setup. Other baseline numbers are taken from prior work. Best within each block is in **bold** and "–" denotes missing values.

| Models | Lap14 | | Rest14 | | Rest15 | | Rest16 | |
|---|---|---|---|---|---|---|---|---|
| | Acc | MF1 | Acc | MF1 | Acc | MF1 | Acc | MF1 |
| *Structure-aware methods* | | | | | | | | |
| KGAN-BERT | 82.66 | 78.98 | 87.15 | 82.05 | 86.21 | 74.20 | 92.34 | 81.31 |
| ASHGAT | 77.91 | 73.89 | 83.45 | 76.55 | 81.06 | 68.72 | 88.96 | 74.02 |
| DSSK-GAN-BERT | 82.94 | 78.99 | 87.32 | 82.35 | 87.32 | 74.53 | 93.44 | 80.03 |
| DC-GCN | 80.70 | 77.92 | 87.29 | 81.48 | 84.90 | 72.11 | 92.42 | 79.45 |
| CABiLSTM-BERT | 77.91 | 73.04 | 83.75 | 75.87 | 82.84 | 66.10 | 86.61 | 73.54 |
| *Fine-tuning methods* | | | | | | | | |
| PConvRoBERTa | 83.54 | 80.89 | 89.29 | 84.27 | – | – | – | – |
| ITGCN | 82.76 | 79.37 | 88.21 | 82.35 | 87.64 | 75.53 | 93.51 | 79.75 |
| DeBERTa+RCL | 82.76 | 80.28 | 89.38 | 84.68 | – | – | – | – |
| Flan-T5-base+Syn-Chain | 83.22 | 80.04 | 88.39 | 82.79 | 87.82 | **76.86** | 93.50 | 79.25 |
| *Large Language Models (LLM-based, 5-shot)* | | | | | | | | |
| Llama-3-8B-Inst | 64.80 | 55.79 | 82.87 | 71.82 | 84.07 | 75.30 | 79.00 | 68.41 |
| Qwen3-8B | 68.02 | 62.24 | 82.75 | 73.52 | 81.38 | 70.63 | 86.60 | 73.85 |
| GPT-3.5 Turbo | 65.48 | 59.74 | 78.85 | 66.50 | 77.63 | 67.28 | 81.30 | 72.91 |
| **DABS (Ours)** | **84.41**$_{\pm 0.15}$ | **81.56**$_{\pm 0.29}$ | **89.76**$_{\pm 0.19}$ | **84.87**$_{\pm 0.47}$ | **89.18**$_{\pm 1.20}$ | 74.06$_{\pm 1.61}$ | **94.87**$_{\pm 0.25}$ | **84.38**$_{\pm 1.65}$ |

Table 3: Matched-backbone comparison on RoBERTa-base. Both models use the same encoder and training protocol. Acc/MF1 (%) and ΔMF1 relative to the matched encoder-only baseline are reported.

| Dataset | Encoder-only | DABS | ΔMF1 |
|---|---|---|---|
| Lap14 | 80.00 / 76.85 | 83.62 / 80.84 | +3.99 |
| Rest14 | 81.83 / 72.70 | 87.29 / 81.09 | +8.39 |
| Rest15 | 83.03 / 67.73 | 88.01 / 71.37 | +3.64 |
| Rest16 | 90.51 / 73.79 | 94.11 / 82.26 | +8.47 |

Table 4: Paired significance test against the encoder-only baseline (3 matched seeds). $\star$ indicates $p < 0.05$.

| Metric | Lap14 | Rest14 | Rest16 |
|---|---|---|---|
| $\text{MF1}_{\text{Full}}$ | 81.56$_{\pm 0.29}$ | 84.87$_{\pm 0.47}$ | 84.38$_{\pm 1.65}$ |
| $\text{MF1}_{\text{Enc}}$ | 75.03$_{\pm 0.43}$ | 76.33$_{\pm 1.24}$ | 75.76$_{\pm 2.49}$ |
| Δ MF1 | 6.53 | 8.54 | 8.62 |
| $t$-statistic | 17.38 | 16.19 | 17.69 |
| $p$-value | 0.0033 | 0.0038 | 0.0032 |
| Significance | $\star$ | $\star$ | $\star$ |

$p<0.05$ in both cases. Low standard deviations confirm that these gains are consistently driven by our query-readout formulation. Per-seed results are provided in Appendix D.

**Matched-backbone validation.** To isolate framework gains from backbone differences, we additionally evaluate DABS under a matched-backbone setting using RoBERTa-base for both the encoder-only baseline and the full model. Table 3 reports the results. DABS remains consistently stronger across all four datasets, improving MF1 by +3.64 to +8.47 points over the matched encoder-only baseline. These results indicate that the gains are not attributable to backbone choice, but arise from the proposed single-pass depth substrate and aspect-conditioned readout. Full cross-backbone comparisons are provided in Appendix C.2.

### 4.3 Multilingual Generalization

To assess multilingual generalization beyond English, we evaluate DABS on three non-English SemEval-2016 ABSA benchmarks (French, Russian, and Spanish), all converted into the same ATSA format (Table 6). DABS consistently improves over the encoder-only baseline on all three languages in Acc and MF1. Notably, most of the gains come from ACBS-only, suggesting that ACBS transfers well across languages. Instead, DORA primarily serves as an enabling component for *single-pass representation reuse* and for exposing an ordered depth substrate that ACBS can query. Combining both components yields the best overall results, supporting that DORA and ACBS are complementary rather than individually sufficient. As a whole, the ability to navigate the **depth substrate** is language-agnostic. Detailed per-seed multilingual results and depth-control analyses via region/layer masking are provided in Appendix E.

### 4.4 Efficiency and Reuse Analysis

A central advantage of DABS lies in its ability to **amortize representational cost** across multiple aspect queries while allowing each aspect to in-

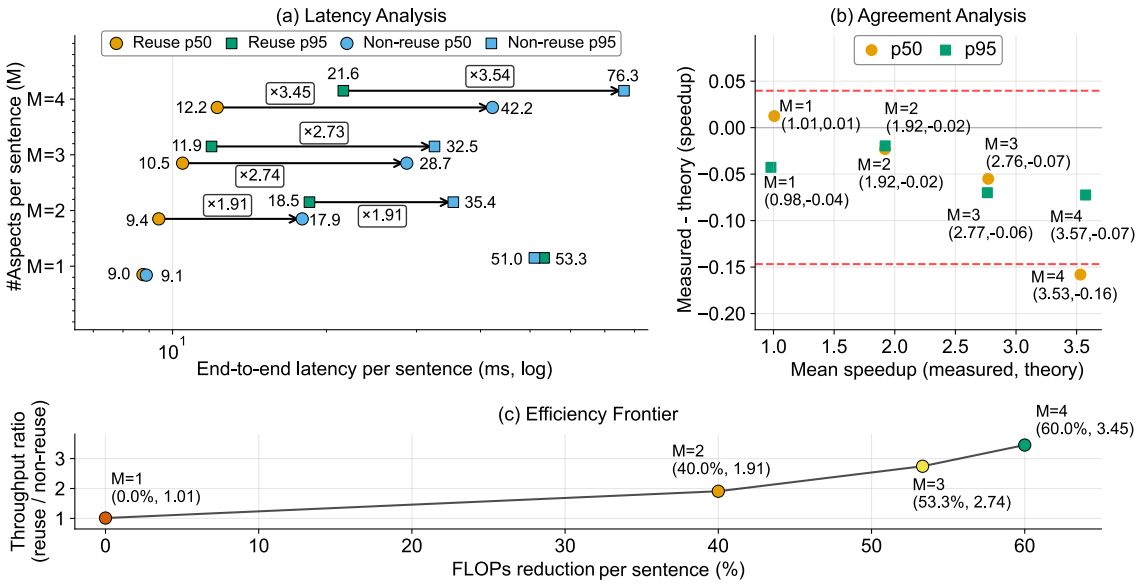


Figure 2: Single-pass reuse under multi-aspect inference ($M$ aspects per sentence), measured with DeBERTa-v3-base on an NVIDIA RTX 5090 GPU. (a) Reuse reduces p50/p95 latency compared to non-reuse, with larger gains as $M$ increases. (b) Measured speedups match the cost-model predictions; red dashed lines mark the 95% agreement bounds. (c) FLOPs reduction leads to higher throughput, forming a clear efficiency frontier as $M$ grows.

dependently allocate depth based on its semantic demands. In DABS, the expensive encoder pass and reusable depth substrate are computed once per sentence, and additional aspects incur only the marginal cost of lightweight depth allocation and evidence selection. In contrast, non-reuse baselines re-encode the sentence for each aspect, repeatedly paying the full cost of deep representation regardless of aspect complexity. To isolate the fixed construction overhead, Table 13 in Appendix reports component-wise metrics under a single-aspect setting ($M$=1). As $M$ increases, this fixed cost is rapidly amortized, leading to the significant latency and throughput gains illustrated in Figure 2.

**End-to-end latency.** Beyond the reuse/non-reuse comparison, we evaluate end-to-end latency against KGAN-BERT, a representative structure-aware baseline with a complete runnable pipeline. Table 5 shows that DABS consistently reduces p95/p99 latency under offered load, indicating that the benefit of single-pass reuse is not limited to the internal amortization analysis in Figure 2, but also yields end-to-end gains against a competitive baseline.

Figure 2(a) shows this effect in terms of latency as the number of aspects $M$ increases. While non-reuse models exhibit near-linear growth due to repeated deep encoding, DABS shows substantially flatter latency curves. At $M$=4, DABS substantially reduces both median and tail latency over the non-reuse baseline. This shows that depth-related computation is largely paid once, and subsequent aspects primarily incur the cost of selective depth querying. Figure 2(b) compares the observed speedup against the theoretical amortization predicted by the cost model. We observe close agreement between measured and cost-model speedup. Deviations remain small for most settings but increase at higher $M$, with the largest gap in $p_{50}$ at $M$=4. Tail-latency ($p_{95}$) deviations also increase from $M \geq 3$ but remain comparatively stable.

Finally, Figure 2(c) summarizes the resulting cost-throughput tradeoff. As $M$ increases, the proportion of computation attributed to shared depth construction dominates, leading to increasing

Table 5: End-to-end tail latency (ms) under offered load, comparing DABS with KGAN-BERT. Lower is better.

| Offered QPS | KGAN-BERT | | DABS (Ours) | | p95 Speedup | p99 Speedup |
|---|---|---|---|---|---|---|
| | p95 | p99 | p95 | p99 | | |
| 60 | 269.48 | 328.95 | 131.58 | 176.21 | 2.05× | 1.87× |
| 100 | 286.43 | 345.73 | 188.10 | 203.73 | 1.52× | 1.70× |

Table 6: Multilingual transfer on SemEval-2016 ABSA benchmarks (French/Russian/Spanish). Results are *mean*±*std* over 3 seeds in Acc. and MF1.

| Methods | French | | Russian | | Spanish | |
|---|---|---|---|---|---|---|
| | Acc | MF1 | Acc | MF1 | Acc | MF1 |
| Baseline | $85.64_{\pm 0.62}$ | $74.67_{\pm 0.63}$ | $84.50_{\pm 0.89}$ | $73.29_{\pm 0.68}$ | $88.95_{\pm 0.57}$ | $71.73_{\pm 1.94}$ |
| DORA-only | $85.03_{\pm 0.32}$ | $73.97_{\pm 0.16}$ | $84.88_{\pm 0.37}$ | $72.87_{\pm 0.30}$ | $88.29_{\pm 0.86}$ | $71.25_{\pm 2.52}$ |
| ACBS-only | $88.56_{\pm 0.87}$ | $79.17_{\pm 1.68}$ | $88.62_{\pm 0.64}$ | $77.75_{\pm 1.83}$ | $92.71_{\pm 0.67}$ | $77.97_{\pm 1.10}$ |
| **DABS** | $\mathbf{89.28}_{\pm 0.89}$ | $\mathbf{80.94}_{\pm 1.34}$ | $\mathbf{89.39}_{\pm 0.32}$ | $\mathbf{79.89}_{\pm 0.14}$ | $\mathbf{93.14}_{\pm 0.86}$ | $\mathbf{80.05}_{\pm 1.44}$ |

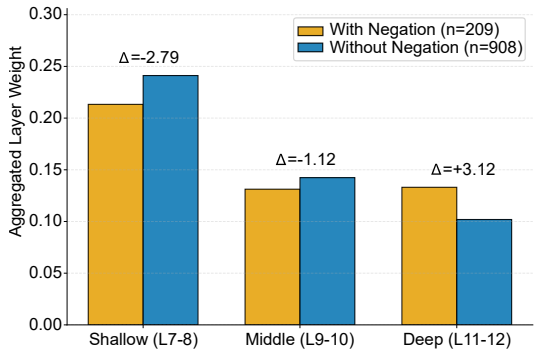


Figure 3: Depth allocation under negation. Aggregated depth-region weights for sentences with vs. without negation, showing a shift toward deeper layers.

Table 7: Sensitivity to the depth budget. Values are reported as Acc, MF1, and p50 latency.

| Dataset | Metric | K=2 | K=4 | K=6 | K=8 |
|---|---|---|---|---|---|
| Lap14 | Acc | $83.20_{\pm 0.92}$ | $83.36_{\pm 0.73}$ | $\mathbf{84.41}_{\pm 0.15}$ | $83.88_{\pm 0.24}$ |
| | MF1 | $80.27_{\pm 1.02}$ | $80.55_{\pm 1.18}$ | $\mathbf{81.56}_{\pm 0.29}$ | $80.78_{\pm 0.17}$ |
| | p50 | $\mathbf{11.64}_{\pm 0.37}$ | $12.11_{\pm 0.28}$ | $12.51_{\pm 0.17}$ | $12.95_{\pm 0.86}$ |
| Rest14 | Acc | $89.38_{\pm 0.31}$ | $89.38_{\pm 0.27}$ | $\mathbf{89.76}_{\pm 0.19}$ | $89.47_{\pm 0.70}$ |
| | MF1 | $84.05_{\pm 0.63}$ | $83.89_{\pm 0.80}$ | $\mathbf{84.87}_{\pm 0.47}$ | $84.47_{\pm 1.40}$ |
| | p50 | $\mathbf{12.18}_{\pm 0.43}$ | $12.42_{\pm 0.36}$ | $12.70_{\pm 0.55}$ | $12.92_{\pm 0.67}$ |
| Rest15 | Acc | $88.44_{\pm 0.70}$ | $88.68_{\pm 1.23}$ | $89.18_{\pm 1.20}$ | $\mathbf{89.28}_{\pm 0.75}$ |
| | MF1 | $73.08_{\pm 1.35}$ | $73.68_{\pm 2.08}$ | $74.06_{\pm 1.61}$ | $\mathbf{74.61}_{\pm 1.48}$ |
| | p50 | $\mathbf{12.43}_{\pm 0.52}$ | $12.79_{\pm 0.41}$ | $13.03_{\pm 0.61}$ | $13.31_{\pm 0.58}$ |
| Rest16 | Acc | $93.78_{\pm 1.13}$ | $94.05_{\pm 0.47}$ | $\mathbf{94.87}_{\pm 0.25}$ | $93.84_{\pm 0.50}$ |
| | MF1 | $82.56_{\pm 2.72}$ | $83.53_{\pm 1.91}$ | $\mathbf{84.38}_{\pm 1.65}$ | $82.33_{\pm 2.61}$ |
| | p50 | $\mathbf{12.12}_{\pm 0.24}$ | $12.46_{\pm 0.47}$ | $12.78_{\pm 0.34}$ | $13.09_{\pm 0.65}$ |

FLOPs reduction (up to 60.0% at $M$=4) and higher throughput ratios (up to 3.45×). Together, these results show that DABS separates fixed deep encoding from aspect-specific depth selection, enabling efficient multi-aspect inference without sacrificing fine-grained accuracy.

**Why querying the last $K$ layers.** A setting of $K$=6 balances expressiveness and overhead. The upper Transformer layers typically encode higher-level compositional and discourse interactions, while still retaining sufficient lexical grounding for span-level polarity. Restricting the substrate to the last $K$ layers also makes depth controls (region/layer masking) easier to interpret and keeps the additional DORA parameters and memory footprint bounded. Table 7 also shows that larger budgets bring smaller gains, while p50 latency increases across the four benchmarks. Together, these results support $K$=6 as a stable choice.

### 4.5 Qualitative Analysis

To verify if DABS learns meaningful linguistic structures, we analyze how depth selection shifts in the presence of negation.

**Negation induces deeper reading.** Negation is a canonical compositional phenomenon where correct sentiment depends on combining multiple cues (*e.g.*, "*not*" reversing "*good*") rather than single lexical triggers. As shown in Figure 3, which aggregates data from 1,117 test samples ($n = 209$ negated, $n = 908$ non-negated), the learned depth selector exhibits a distinct shift. Specifically, the model reduces weight on surface-level features ($\Delta = -2.79$ pp in Shallow layers $L_{7\text{-}8}$) and actively recruits deeper abstractions ($\Delta = +3.12$ pp in Deep layers $L_{11\text{-}12}$). This pattern provides direct evidence that the selector does not mix layers arbitrarily. Instead, it functions as a semantic cursor, automatically shifting from **surface-level lexical matching** to **deep compositional logic** only when the linguistic complexity requires it.

**Case Studies.** Figure 4(a) shows that the learned instance embeddings form label-aligned regions with remaining overlap concentrated around neutral and hard examples. Figure 4(b) shows how fusion gates and depth allocation correlate with both correct predictions and failures. For successful cases, Case 1 (negation) is context-focused, while Case 3 (explicit sentiment) becomes aspect-dominant and concentrates on middle layers ($L_{9\text{-}10}$). The neutral-factual Case 2 shifts $\boldsymbol{\alpha}$ toward deeper layers ($L_{11\text{-}12}$), consistent with higher contextual demand. Notably, high-confidence errors (Cases 4-6) expose systematic limits in which the model could mis-handle *implicit/pragmatic negativity* (Case 4), confuse *target-opinion alignment* when multiple nearby candidates exist (Case 5), and fail on *rhetorical polarity* without explicit sentiment cues (Case 6). Overall, the gates and $\boldsymbol{\alpha}$ diagnostically align with compositional needs for explicit sentiment, while errors track implicit or misaligned evidence.

### 4.6 Ablation Study

To validate the contributions of DORA and ACBS, we perform an ablation on MF1 (Table 8). Accuracy ablations follow similar trends and are reported in Appendix F.1 (Table 18).

**Impact of Regularizers.** Removing sparsity causes the largest drop on Rest16 (-2.13 pp), and on average -1.61 confirming that filtering redundant context is crucial for longer reviews. Gate entropy is essential for preventing mode collapse in the fusion layer as removing it degrades performance consistently across all datasets.

**Impact of DORA.** The removal of DepthGRU

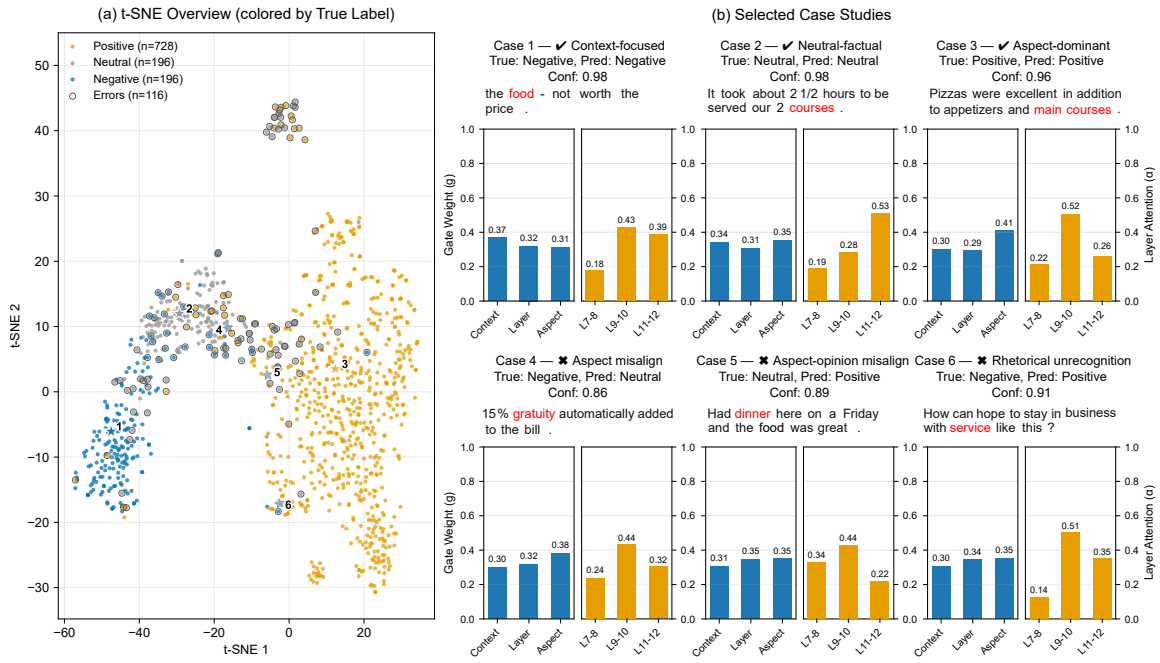


Figure 4: Learned representations and selection behavior. (a) t-SNE of instance embeddings by gold label. (b) Case studies showing fusion-gate weights and depth level $L$.

Table 8: Ablations on $\mathrm{MF1}_{\mathrm{mean}}$. Values are reported as $\mathrm{MF1}^{\Delta}$, where $\Delta$ is the change (pp) from the full model.

| Config | Lap14 $\mathrm{MF1}^{\Delta}$ | Rest14 $\mathrm{MF1}^{\Delta}$ | Rest15 $\mathrm{MF1}^{\Delta}$ | Rest16 $\mathrm{MF1}^{\Delta}$ | Avg $\Delta$ |
|---|---|---|---|---|---|
| **DABS (Full)** | **81.56**$^{0.00}$ | **84.87**$^{0.00}$ | **74.06**$^{0.00}$ | **84.38**$^{0.00}$ | 0.00 |
| *w/o Regularizers* | | | | | |
| - Sparsity | 79.84$^{\downarrow 1.72}$ | 83.68$^{\downarrow 1.19}$ | 72.68$^{\downarrow 1.38}$ | 82.25$^{\downarrow 2.13}$ | $\downarrow$1.61 |
| - Span Masking | 79.77$^{\downarrow 1.79}$ | 82.77$^{\downarrow 2.10}$ | 72.62$^{\downarrow 1.44}$ | 84.37$^{\downarrow 0.01}$ | $\downarrow$1.34 |
| - Gate Entropy | 79.68$^{\downarrow 1.88}$ | 84.14$^{\downarrow 0.73}$ | 73.08$^{\downarrow 0.98}$ | 83.33$^{\downarrow 1.05}$ | $\downarrow$1.16 |
| *w/o DORA* | | | | | |
| - DepthGRU | 80.81$^{\downarrow 0.75}$ | 83.45$^{\downarrow 1.42}$ | 74.46$^{\uparrow 0.40}$ | 80.73$^{\downarrow 3.65}$ | $\downarrow$1.36 |
| - LCP (Pooling) | 79.31$^{\downarrow 2.25}$ | 82.74$^{\downarrow 2.13}$ | 75.29$^{\uparrow 1.23}$ | 80.37$^{\downarrow 4.01}$ | $\downarrow$1.79 |
| *w/o ACBS* | | | | | |
| - Token Sel. | 78.25$^{\downarrow 3.31}$ | 83.14$^{\downarrow 1.73}$ | 71.84$^{\downarrow 2.22}$ | 80.57$^{\downarrow 3.81}$ | $\downarrow$2.77 |
| - Layer Sel. | 79.95$^{\downarrow 1.61}$ | 82.35$^{\downarrow 2.52}$ | 72.63$^{\downarrow 1.43}$ | 83.04$^{\downarrow 1.34}$ | $\downarrow$1.73 |
| - Gated Fusion | 79.06$^{\downarrow 2.50}$ | 81.68$^{\downarrow 3.19}$ | 73.29$^{\downarrow 0.77}$ | 82.02$^{\downarrow 2.36}$ | $\downarrow$2.21 |

results in a 1.36 pp drop on average, validating that multi-level aggregation is necessary to capture semantic abstractions in complex sentences. LCP is similarly critical, without it, the model fails to capture local compositional cues (*e.g.*, negations).

**Impact of ACBS.** This is the most critical module. Removing token selection yields the largest average degradation (-2.77 pp), proving that independent token gating is essential for aggregating distributed sentiment evidence. Layer selection is equally vital, as different aspects require different levels of semantic abstraction (see Section 4.5).

**Impact of DepthGRU.** This ablation isolates DepthGRU within DORA by removing only the cross-layer recurrence while keeping all other components fixed. Table 9 shows that removing DepthGRU decreases Macro-F1 on all stress-test splits. Most notably on Multi-Aspect Conflict (MA-C) and Complex Negation, while Accuracy can remain relatively stable, suggesting DepthGRU primarily improves hard/minority-case discrimination. See Appendix G for completeness.

Table 9: DepthGRU ablation on cross-dataset stress-test splits ($K$=6). Results are over 3 seeds (Acc/MF1).

| **Dataset** | **Setting** | **Acc** | **MF1** |
|---|---|---|---|
| Long Sentences | w DepthGRU | 82.95$_{\pm 0.91}$ | 78.44$_{\pm 0.76}$ |
| | w/o DepthGRU | 82.61$_{\pm 1.57}$ | 76.93$_{\pm 2.14}$ |
| Multi-Aspect Conflict | w DepthGRU | 95.40$_{\pm 1.15}$ | 76.88$_{\pm 10.80}$ |
| | w/o DepthGRU | 94.83$_{\pm 0.57}$ | 71.32$_{\pm 6.29}$ |
| Complex Negation | w DepthGRU | 83.81$_{\pm 2.18}$ | 80.62$_{\pm 1.49}$ |
| | w/o DepthGRU | 82.86$_{\pm 2.86}$ | 77.69$_{\pm 3.36}$ |

## 5 Conclusion

This paper addresses the efficiency-expressiveness tradeoff in ATSA by framing inference as a resource allocation problem. We propose DABS, a single-pass framework that reuses a shared encoder pass while enabling lightweight, aspect-conditioned readout. Across four benchmarks, DABS achieves competitive Accuracy and Macro-F1 while reducing end-to-end computation by up to 60% in multi-aspect settings. These results show that explicitly querying Transformer depth, rather than treating it as a static feature pool, enables efficient reuse and targeted sentiment reasoning.

## Acknowledgment

This work is financially supported by VinUniversity under Grant No. VUNI.2526.AREP.004.

## Limitations

While DABS shows that *single-pass reuse* and *budget-aware depth querying* can improve the efficiency-expressiveness tradeoff in multi-aspect ATSA, its current formulation is constrained by three practical constraints. Specifically, **(i) Amortization-dependent gains.** Since DABS improves efficiency by amortizing a shared encoder pass across aspects, rather than by reducing backbone computation itself, its advantage is most pronounced in true multi-aspect settings and naturally smaller when $M{=}1$; **(ii) Retention cost of the depth substrate.** Keeping the last $K$ layers as a reusable substrate incurs a fixed compute and memory cost that increases with sequence length, model size, and $K$, and finally **(iii) Task scope.** We evaluate DABS only in span-given ATSA with gold aspect spans. Extending the framework to settings with latent, implicit, or jointly extracted aspects is left to future work.

## References

Sangmin Bae, Jongwoo Ko, Hwanjun Song, and Se-Young Yun. 2023. Fast and robust early-exiting framework for autoregressive language models with synchronized parallel decoding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5910–5924, Singapore. Association for Computational Linguistics.

Xiaoyi Bao, Xiaotong Jiang, Zhongqing Wang, Yue Zhang, and Guodong Zhou. 2023a. Opinion tree parsing for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7971–7984, Toronto, Canada. Association for Computational Linguistics.

Xiaoyi Bao, Zhongqing Wang, and Guodong Zhou. 2023b. Exploring graph pre-training for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3623–3634, Singapore. Association for Computational Linguistics.

Aaditya Bodke, Avinoor Singh Kohli, Hemant Subhash Pardeshi, and Prathamesh Bhosale. 2025. PASTEL : Polarity-aware sentiment triplet extraction with LLM-as-a-judge. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25523–25533, Vienna, Austria. Association for Computational Linguistics.

Laura Cabello and Uchenna Akujuobi. 2024. It is simple sometimes: A study on improving aspect-based sentiment analysis performance. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6597–6610, Bangkok, Thailand. Association for Computational Linguistics.

Heyan Chai, Ziyi Yao, Siyu Tang, Ye Wang, Liqiang Nie, Binxing Fang, and Qing Liao. 2023. Aspect-to-scope oriented multi-view contrastive learning for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10902–10913, Singapore. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Xuanwen Ding, Jie Zhou, Liang Dou, Qin Chen, Yuanbin Wu, Arlene Chen, and Liang He. 2024. Boosting large language models with continual learning for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4367–4377, Miami, Florida, USA. Association for Computational Linguistics.

Mostafa Elhoushi, Akshat Shrivastava, Diana Liskovich, Basil Hosmer, Bram Wasti, Liangzhen Lai, Anas Mahmoud, Bilge Acun, Saurabh Agarwal, Ahmed Roman, Ahmed Aly, Beidi Chen, and Carole-Jean Wu. 2024. LayerSkip: Enabling early exit inference and self-speculative decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12622–12642, Bangkok, Thailand. Association for Computational Linguistics.

Rui Fan, Shu Li, Tingting He, and Yu Liu. 2025. Aspect-based sentiment analysis with syntax-opinion-sentiment reasoning chain. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3123–3137, Abu Dhabi, UAE. Association for Computational Linguistics.

Ao Feng, Jiazhi Cai, Zhengjie Gao, and Xiaojie Li. 2023. Aspect-level sentiment classification with fused local and global context. *J. Big Data*, 10(1):176.

Zhibin Gou, Qingyan Guo, and Yujiu Yang. 2023. MvP: Multi-view prompting improves aspect sentiment tuple prediction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4380–4397, Toronto, Canada. Association for Computational Linguistics.

Bo He, Ruoyu Zhao, and Dali Tang. 2025. Cabilstm-bert: Aspect-based sentiment analysis model based

on deep implicit feature extraction. *Knowl. Based Syst.*, 309:112782.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Nils Constantin Hellwig, Jakob Fehle, and Christian Wolff. 2025. Exploring large language models for the generation of synthetic training samples for aspect-based sentiment analysis in low resource settings. *Expert Syst. Appl.*, 261:125514.

ZhongQuan Jian, Jiajian Li, Qingqiang Wu, and Junfeng Yao. 2024. Retrieval contrastive learning for aspect-level sentiment classification. *Inf. Process. Manag.*, 61(1):103539.

Song Jin, Qing He, Yuji Wang, Nisuo Du, and Wenjing Lei. 2025. Aspect-based sentiment analysis with semantic and syntactic enhanced multi-layer fusion model. *Eng. Appl. Artif. Intell.*, 159:111654.

Hongtao Liu, Xin Li, Wanying Lu, Kefei Cheng, and Xueyan Liu. 2024. Graph augmentation networks based on dynamic sentiment knowledge and static external knowledge graphs for aspect-based sentiment analysis. *Expert Syst. Appl.*, 251:123981.

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

Haoran Lv, Junyi Liu, Henan Wang, Yaoming Wang, Jixiang Luo, and Yaxiao Liu. 2023. Efficient hybrid generation framework for aspect-based sentiment analysis. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1007–1018, Dubrovnik, Croatia. Association for Computational Linguistics.

Fukun Ma, Xuming Hu, Aiwei Liu, Yawen Yang, Shuang Li, Philip S. Yu, and Lijie Wen. 2023. Amr-based network for aspect-based sentiment analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 322–337. Association for Computational Linguistics.

Rajdeep Mukherjee, Nithish Kannen, Saurabh Pandey, and Pawan Goyal. 2023. CONTRASTE: Supervised contrastive pre-training with aspect-based prompts for aspect sentiment triplet extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12065–12080, Singapore. Association for Computational Linguistics.

Jihong Ouyang, Zhiyao Yang, Silong Liang, Bing Wang, Yimeng Wang, and Ximing Li. 2024. Aspect-based sentiment analysis with explicit sentiment augmentations. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 18842–18850. AAAI Press.

Jackson Petty, Sjoerd Steenkiste, Ishita Dasgupta, Fei Sha, Dan Garrette, and Tal Linzen. 2024. The impact of depth on compositional generalization in transformer language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7239–7252, Mexico City, Mexico. Association for Computational Linguistics.

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

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015*, pages 486–495. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 27–35. The Association for Computer Linguistics.

Zhiwen Ruan, Yixia Li, He Zhu, Longyue Wang, Weihua Luo, Kaifu Zhang, Yun Chen, and Guanhua Chen. 2025. LayAlign: Enhancing multilingual reasoning in large language models via layer-wise adaptive fusion and alignment strategy. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1481–1495, Albuquerque, New Mexico. Association for Computational Linguistics.

Kevin Scaria, Himanshu Gupta, Siddharth Goyal, Saurabh Sawant, Swaroop Mishra, and Chitta Baral. 2024. InstructABSA: Instruction learning for aspect based sentiment analysis. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human*

*Language Technologies (Volume 2: Short Papers)*, pages 720–736, Mexico City, Mexico. Association for Computational Linguistics.

Yongsik Seo, Sungwon Song, Ryang Heo, Jieyong Kim, and Dongha Lee. 2024. Make compound sentences simple to analyze: Learning to split sentences for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11171–11184, Miami, Florida, USA. Association for Computational Linguistics.

Chuming Shen, Wei Wei, Dong Wang, and Zhong-Hao Wang. 2025. Zero-shot cross-domain aspect-based sentiment analysis via domain-contextualized chain-of-thought reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 4558–4573, Suzhou, China. Association for Computational Linguistics.

Xuefeng Shi, Min Hu, Fuji Ren, Piao Shi, and Satoshi Nakagawa. 2024. Aspect based sentiment analysis with instruction tuning and external knowledge enhanced dependency graph. *Appl. Intell.*, 54(8):6415–6432.

Paul F. Simmering and Paavo Huoviala. 2023. Large language models for aspect-based sentiment analysis. *CoRR*, abs/2310.18025.

Xin Sun, Yongqing Mi, and Hongao Li. 2025. Enhancing aspect sentiment classification with dual-channel graph convolutional network. *ACM Trans. Intell. Syst. Technol.*, 16(3).

Joachim Wagner and Jennifer Foster. 2023. Investigating the saliency of sentiment expressions in aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12751–12769, Toronto, Canada. Association for Computational Linguistics.

Yanyan Wang, Qun Chen, Murtadha H. M. Ahmed, Zhaoqiang Chen, Jing Su, Wei Pan, and Zhanhuai Li. 2023. Supervised gradual machine learning for aspect-term sentiment analysis. *Trans. Assoc. Comput. Linguistics*, 11:723–739.

Zhihao Wang, Bo Zhang, Ru Yang, Chang Guo, and Maozhen Li. 2024. DAGCN: Distance-based and aspect-oriented graph convolutional network for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1863–1876, Mexico City, Mexico. Association for Computational Linguistics.

Xiancai Xu, Jia-Dong Zhang, Lei Xiong, and Zhishang Liu. 2024. iACOS: Advancing implicit sentiment extraction with informative and adaptive negative examples. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4283–4293, Mexico City, Mexico. Association for Computational Linguistics.

Shuo Yin and Guoqiang Zhong. 2024. Textgt: A double-view graph transformer on text for aspect-based sentiment analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19404–19412.

Mao Zhang, Yongxin Zhu, Zhen Liu, Zhimin Bao, Yunfei Wu, Xing Sun, and Linli Xu. 2023a. Span-level aspect-based sentiment analysis via table filling. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9273–9284, Toronto, Canada. Association for Computational Linguistics.

Siyu Zhang, Hongfang Gong, and Lina She. 2023b. An aspect sentiment classification model for graph attention networks incorporating syntactic, semantic, and knowledge. *Knowl. Based Syst.*, 275:110662.

Guangmin Zheng, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2024. Instruction tuning with retrieval-based examples ranking for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4777–4788, Bangkok, Thailand. Association for Computational Linguistics.

Yongqiang Zheng and Xia Li. 2024. You only read once: Constituency-oriented relational graph convolutional network for multi-aspect multi-sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19715–19723.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, Hua Jin, and Dacheng Tao. 2023. Knowledge graph augmented network towards multiview representation learning for aspect-based sentiment analysis. *IEEE Trans. Knowl. Data Eng.*, 35(10):10098–10111.

Linan Zhu, Xiangfan Chen, Xiaolei Guo, Chenwei Zhang, Zhechao Zhu, Zehai Zhou, and Xiangjie Kong. 2024. Pinpointing diffusion grid noise to enhance aspect sentiment quad prediction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3717–3726, Bangkok, Thailand. Association for Computational Linguistics.

## A Dataset Statistics

Section 4.1 describes the datasets and evaluation protocol. For completeness, Table 10 reports the class distributions of the official train/test splits. We include the laptop and restaurant benchmarks from the SemEval-2014 Task 4 (Pontiki et al., 2014), SemEval-2015 Task 12 (Pontiki et al., 2015), and SemEval-2016 Task 5 (Pontiki et al., 2016) challenges. The multilingual datasets (Spanish, French, and Russian) are also sourced from the SemEval-2016 subtasks (Pontiki et al., 2016). All counts are at the *aspect-instance* level (not sentence level), grouped by sentiment label (*positive*, *neutral*, *negative*).

Table 10: Dataset statistics.

| Dataset | Positive | | Neutral | | Negative | |
|---|---|---|---|---|---|---|
| | Train | Test | Train | Test | Train | Test |
| Lap14 | 994 | 341 | 464 | 169 | 870 | 128 |
| Rest14 | 2164 | 728 | 637 | 196 | 807 | 196 |
| Rest15 | 963 | 353 | 36 | 37 | 280 | 207 |
| Rest16 | 1319 | 483 | 72 | 32 | 489 | 135 |
| Spanish | 1368 | 521 | 89 | 34 | 479 | 176 |
| French | 901 | 364 | 116 | 69 | 753 | 285 |
| Russian | 2453 | 670 | 224 | 92 | 481 | 207 |

**Label imbalance and metric sensitivity.** Table 10 highlights a non-trivial label imbalance that is particularly pronounced in the restaurant benchmarks, where the *neutral* class can be extremely small (*e.g.*, Rest15/Rest16). This imbalance makes macro-F1 more sensitive to minority-class recall and can increase variance across random seeds, especially when evaluation subsets further reduce neutral counts. Accordingly, we report both Accuracy (Acc) and Macro-F1 (MF1), as MF1 is more sensitive to minority-class performance. Since the SemEval benchmarks provide only official train/test splits without a standard development set, we train on the official training split and evaluate on the official test split. All statistics are reported at the *aspect-instance* level, meaning a single multi-aspect sentence contributes multiple labeled instances. This is aligned with our multi-query formulation where aspects are treated as parallel queries over shared context.

## B Implementation Details

This appendix summarizes span alignment, hyperparameters, optimization settings, and baseline protocols used in our experiments. Unless otherwise noted, the backbone encoder remains aspect-agnostic so that a single sentence encoding can be reused across all aspects in the same sentence.

### B.1 Tokenization and Span Alignment

We tokenized sentences using the DeBERTa-v3-base tokenizer. SemEval aspect annotations are provided as contiguous spans in the raw sentence. We mapped each gold span to subword indices via offset mappings returned by the tokenizer. When an aspect term was split into multiple subwords, we used the full contiguous subword span and treated it as the aspect interval $[i, j]$ (Section 3.1 in the main paper). Aspect vectors were computed by mean pooling over the enhanced sequence representations $E$ within the aspect span (Section 3.4 in the main paper).

### B.2 Experimental Environment

All experiments were run on an AMD Ryzen 9 9950X CPU and an NVIDIA GeForce RTX 5090 GPU with 32 GB memory. The software stack used PyTorch 2.8.0, CUDA 12.8, and Transformers 4.52.0. We fixed random seeds to $\{42, 123, 456\}$ unless stated otherwise.

### B.3 Model Hyperparameters

We used DeBERTa-v3-base as the default backbone encoder. For depth querying, DORA exposed a depth substrate built from the last $K{=}6$ Transformer layers. In DORA, the Local Context Preservation (LCP) module applied three depthwise-separable convolution branches with kernel sizes $k \in \{1, 3, 5\}$, keeping short-range compositional cues (*e.g.*, negation, degree modifiers) explicit in a reusable substrate.

In ACBS, we applied one multi-head self-attention layer over $E$ to obtain the reusable context $C$. The token selector used a two-layer MLP with shape $2d \to d \to 1$ (GELU), the depth selector used $2d \to d \to K$, and the fusion gate used $3d \to d \to 3$. We set $\varepsilon = 10^{-6}$ in Eq. (7). Temperatures were fixed to $\tau_\alpha = \tau_g = 1.0$ in Eqs. (8) and (9). Unless otherwise noted, the maximum sequence length was 256.

### B.4 Optimization and Regularization

We train on the official training split for up to 30 epochs with a batch size of 32. Because these benchmarks do not provide a standard development

set, for each run we report the best test-set score within the training budget.

Optimization used AdamW with a shared learning rate, selected from {`2e-5`, `3e-5`} depending on the dataset. We used the default linear schedule in Transformers, with dataset-specific warmup ratios. Mixed-precision training (`bf16`) was enabled, and gradient clipping was applied with $\|g\|_2 \leq 1$.

Dropout was set to 0.1 for the encoder and most DORA/ACBS internal modules, while the final classifier used 0.2. For the objective in Eq. (10), we set $\lambda_s = \lambda_m = 10^{-3}$. For the entropy weight $\lambda_{\text{ent}}$, we used a fixed value of $10^{-2}$.

More implementation details and configuration settings are available at https://github.com/panzhzh/acl-dabs.

### B.5 Baseline Protocols

We compared against three baseline groups as described in Section 4.1(in the main paper): (i) structure-aware methods that inject syntactic/-knowledge structure, (ii) fine-tuning paradigms that treat ATSA as supervised classification, and (iii) LLMs evaluated with 5-shot in-context learning.

#### B.5.1 LLM 5-shot In-Context Learning

For each test instance $(s^*, a^*)$, we retrieved $k$=5 demonstrations from the in-domain training split using a convex combination of sentence similarity and aspect-term similarity. We computed cosine similarities between normalized embeddings. Let $u_i = \cos(\phi(s^*), \phi(s_i))$ denote sentence similarity and $v_i = \cos(\phi(a^*), \phi(a_i))$ denote aspect similarity. Candidates were ranked by

$$\text{score}_i = \lambda\, v_i + (1 - \lambda)\, u_i, \quad \lambda = 0.6,$$

and we used the top-$k$ items as demonstrations.

Here, $\phi(\cdot)$ is a frozen embedding function used only for retrieval. To avoid introducing extra models, we used mean-pooled last-layer embeddings from the same DeBERTa-v3-base encoder (without task-specific heads) as $\phi(\cdot)$. Prompts followed a fixed template and required the model to output exactly one label from {`positive`, `neutral`, `negative`}. Decoding used temperature $= 0.1$, top-$p = 1.0$, and max_new_tokens $= 1$. We did not update model parameters for LLM baselines.

The 5-shot prompt template is shown below.

Table 11: Extended effectiveness (mean±std over 3 seeds): MF1/Acc and Δ (pp) relative to the encoder-only baseline. **Bold** indicates the best result per dataset.

| Dataset | Config | MF1 | Δ MF1 | Acc | Δ Acc |
|---|---|---|---|---|---|
| Lap14 | Encoder-only | $75.03_{\pm 0.43}$ | 0.00 | $79.16_{\pm 0.24}$ | 0.00 |
| Lap14 | DORA-only | $75.72_{\pm 0.88}$ | 0.69 | $79.63_{\pm 0.51}$ | 0.47 |
| Lap14 | ACBS-only | $79.97_{\pm 0.73}$ | 4.94 | $82.62_{\pm 0.55}$ | 3.46 |
| Lap14 | Full | $\mathbf{81.56_{\pm 0.29}}$ | 6.53 | $\mathbf{84.41_{\pm 0.16}}$ | 5.25 |
| Rest14 | Encoder-only | $76.33_{\pm 1.24}$ | 0.00 | $84.48_{\pm 0.49}$ | 0.00 |
| Rest14 | DORA-only | $73.18_{\pm 0.23}$ | -3.15 | $83.02_{\pm 0.31}$ | -1.46 |
| Rest14 | ACBS-only | $83.15_{\pm 1.68}$ | 6.82 | $88.81_{\pm 1.01}$ | 4.33 |
| Rest14 | Full | $\mathbf{84.87_{\pm 0.47}}$ | 8.54 | $\mathbf{89.76_{\pm 0.19}}$ | 5.28 |
| Rest15 | Encoder-only | $71.39_{\pm 1.12}$ | 0.00 | $85.85_{\pm 0.91}$ | 0.00 |
| Rest15 | DORA-only | $66.60_{\pm 4.87}$ | -4.79 | $84.56_{\pm 1.23}$ | -1.29 |
| Rest15 | ACBS-only | $72.60_{\pm 0.39}$ | 1.21 | $88.19_{\pm 0.85}$ | 2.34 |
| Rest15 | Full | $\mathbf{74.06_{\pm 1.61}}$ | 2.67 | $\mathbf{89.18_{\pm 1.20}}$ | 3.33 |
| Rest16 | Encoder-only | $75.76_{\pm 2.49}$ | 0.00 | $91.00_{\pm 0.43}$ | 0.00 |
| Rest16 | DORA-only | $72.82_{\pm 1.50}$ | -2.94 | $90.34_{\pm 0.82}$ | -0.66 |
| Rest16 | ACBS-only | $82.25_{\pm 1.43}$ | 6.49 | $94.11_{\pm 0.59}$ | 3.11 |
| Rest16 | Full | $\mathbf{84.38_{\pm 1.65}}$ | 8.62 | $\mathbf{94.87_{\pm 0.25}}$ | 3.87 |

**Aspect-Term Sentiment Prompt (5-shot)**

You are an aspect-term sentiment classifier. Your task is to determine the sentiment polarity of a given aspect term within a sentence.

Follow the rules below strictly:
1. Output must be exactly one label chosen from: `positive`, `neutral`, `negative`.
2. Use lowercase only. Do not output any additional words, punctuation, or whitespace.
3. Base your decision solely on the provided sentence and the marked aspect term.
4. Do not provide explanations or reformulations.

**Examples**

`Sentence: {s1}  Aspect: {a1}  Label: {y1}`
...
`Sentence: {sk}  Aspect: {ak}  Label: {yk}`

**Query**

`Sentence: {s*}  Aspect: {a*}  Label:`

## C Extended Experiments and Reproducibility

### C.1 Cross-dataset Consistency and Discriminability

Table 11 decomposes DABS into its two functional parts: (i) DORA, which constructs an ordered depth substrate intended for reuse and depth-sensitive querying, and (ii) ACBS, which performs aspect-conditioned retrieval over tokens and depths.

Two patterns are consistent across datasets. First, **ACBS-only** provides the majority of the gains over the encoder-only baseline, indicating that aspect-conditioned evidence localization and depth selec-

Table 12: Performance comparison across different backbone encoders (DeBERTa, RoBERTa, BERT).

| Dataset | Config | DeBERTa-base | | | | RoBERTa-base | | | | BERT-base | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MF1 | Δ MF1 | Acc | Δ Acc | MF1 | Δ MF1 | Acc | Δ Acc | MF1 | Δ MF1 | Acc | Δ Acc |
| Lap14 | Encoder-only | 75.03 | 0.00 | 79.16 | 0.00 | 76.85 | 0.00 | 80.00 | 0.00 | 73.88 | 0.00 | 78.43 | 0.00 |
| Lap14 | DORA-only | 75.72 | 0.69 | 79.63 | 0.47 | 77.09 | 0.24 | 81.42 | 1.42 | 74.34 | 0.46 | 77.95 | -0.48 |
| Lap14 | ACBS-only | 79.97 | 4.94 | 82.62 | 3.46 | 79.93 | 3.08 | 82.99 | 2.99 | 75.94 | 2.06 | 79.53 | 1.10 |
| Lap14 | DABS (Full) | 81.56 | 6.53 | 84.41 | 5.25 | 80.84 | 3.99 | 83.62 | 3.62 | 76.94 | 3.06 | 80.31 | 1.88 |
| Res14 | Encoder-only | 76.33 | 0.00 | 84.48 | 0.00 | 72.70 | 0.00 | 81.83 | 0.00 | 70.61 | 0.00 | 80.21 | 0.00 |
| Res14 | DORA-only | 73.18 | -3.15 | 83.02 | -1.46 | 74.37 | 1.67 | 83.17 | 1.34 | 71.56 | 0.95 | 80.30 | 0.09 |
| Res14 | ACBS-only | 83.15 | 6.82 | 88.81 | 4.33 | 80.93 | 8.23 | 87.02 | 5.19 | 77.93 | 7.32 | 84.51 | 4.30 |
| Res14 | DABS (Full) | 84.87 | 8.54 | 89.76 | 5.28 | 81.09 | 8.39 | 87.29 | 5.46 | 80.47 | 9.86 | 86.12 | 5.91 |
| Res15 | Encoder-only | 71.39 | 0.00 | 85.85 | 0.00 | 67.73 | 0.00 | 83.03 | 0.00 | 65.74 | 0.00 | 82.10 | 0.00 |
| Res15 | DORA-only | 66.60 | -4.79 | 84.56 | -1.29 | 66.82 | -0.91 | 83.21 | 0.18 | 63.38 | -2.36 | 81.92 | -0.18 |
| Res15 | ACBS-only | 72.60 | 1.21 | 88.19 | 2.34 | 70.55 | 2.82 | 87.27 | 4.24 | 63.14 | -2.60 | 83.03 | 0.93 |
| Res15 | DABS (Full) | 74.06 | 2.67 | 89.18 | 3.33 | 71.37 | 3.64 | 88.01 | 4.98 | 66.98 | 1.24 | 83.95 | 1.85 |
| Res16 | Encoder-only | 75.76 | 0.00 | 91.00 | 0.00 | 73.79 | 0.00 | 90.51 | 0.00 | 70.98 | 0.00 | 88.87 | 0.00 |
| Res16 | DORA-only | 72.82 | -2.94 | 90.34 | -0.66 | 73.91 | 0.12 | 89.85 | -0.66 | 68.81 | -2.17 | 88.54 | -0.33 |
| Res16 | ACBS-only | 82.25 | 6.49 | 94.11 | 3.11 | 81.74 | 7.95 | 93.62 | 3.11 | 72.84 | 1.86 | 90.02 | 1.15 |
| Res16 | DABS (Full) | 84.38 | 8.62 | 94.87 | 3.87 | 82.26 | 8.47 | 94.11 | 3.60 | 78.79 | 7.81 | 92.47 | 3.60 |

tion are the primary drivers of discriminability. Second, **DORA-only** is not designed to be a standalone classifier. Its role is to restructure representations for reuse. In other words, it is intentionally aspect-agnostic, and its role is to reshape representations into a reusable, depth-indexed substrate rather than to optimize aspect-specific decision boundaries. When combined, DORA enables ACBS to query a structured depth bank with minimal per-aspect cost, yielding the strongest and most stable improvements in the full model.

### C.2 Backbone-Agnostic Robustness

To verify that DABS is not tied to a specific pretrained encoder, we evaluate the same architecture under alternative backbone encoders, replacing DeBERTa-base with RoBERTa-base and BERT-base while keeping DORA/ACBS unchanged. We follow the same training protocol as in the main experiments (same datasets, optimization, and three random seeds), and report MF1/Acc together with improvements over the corresponding encoder-only baseline.

Table 12 shows that DABS yields consistent gains across backbones and datasets. For instance, DABS improves MF1 on Rest14 by +8.39 (RoBERTa-base (Liu et al., 2019)) and +9.86 (BERT-base (Devlin et al., 2019)), and on Rest16 by +8.47 (RoBERTa-base) and +7.81 (BERT-base). Similar improvements are observed on Lap14 and Rest15, indicating that our single-pass, depth-selective querying mechanism is backbone-agnostic. Overall, these results support that the benefits of reusable depth substrates and aspect-conditioned depth allocation arise from the proposed readout formulation rather than encoder-specific artifacts.

### C.3 Overall Efficiency and Resource Overhead

Table 13 provides a component-wise efficiency breakdown under a single-aspect protocol ($M{=}1$). Each measurement corresponds to one *(sentence, aspect)* query and therefore does not benefit from cross-aspect reuse. Under this setting, the full model can be slower than Encoder-only because it pays a fixed overhead to construct and retain the depth substrate (primarily from DORA). This table is included to isolate that fixed cost at $M{=}1$. For the deployment-relevant multi-aspect setting ($M \geq 2$), DABS computes the encoder+DORA substrate once per sentence and reuses it across aspects, yielding the amortized gains shown in Figure 2 (in the main paper).

**Why single-aspect can be slower.** We include the $M{=}1$ setting solely to isolate fixed construction costs, not as a target deployment scenario. Under $M{=}1$, DABS cannot amortize the fixed cost of substrate construction and therefore can be slower than encoder-only. This is expected as DORA introduces additional kernels (local convolutions) and depth-ordered integration (DepthGRU), and ACBS adds lightweight MLP-based gating. We report p50/p95 to separate typical from tail behavior

Table 13: Single-aspect ($M$=1) efficiency breakdown: FLOPs, throughput, latency (p50/p95), parameter counts, and relative change vs. encoder-only. **Bold** marks the best value per column.

| | | | | Latency (ms) | | | Parameters | | Δ vs Encoder (%) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | Config | FLOPs | queries/s | p50 | p95 | p95/p50 | Total | Added | FLOPs | p50 | queries/s |
| Lap14 | Encoder-only | **14.48** | **114.2** | **7.6** | 15.6 | 2.05 | **184.4** | **1.2** | **0.0** | **0.0** | **0.0** |
| Lap14 | DORA-only | 15.48 | 16.8 | 16.8 | 133.8 | 7.96 | 193.0 | 9.8 | 6.9 | 121.1 | -85.3 |
| Lap14 | ACBS-only | 14.58 | 69.9 | 9.0 | **10.0** | **1.11** | 192.1 | 8.9 | 0.7 | 18.4 | -38.8 |
| Lap14 | Full | 15.58 | 23.5 | 35.0 | 135.1 | 3.86 | 196.8 | 13.6 | 7.6 | 360.5 | -79.4 |
| Rest14 | Encoder-only | **13.88** | 42.8 | **8.4** | 56.5 | 6.73 | **184.4** | **1.2** | **0.0** | 0.0 | 0.0 |
| Rest14 | DORA-only | 14.81 | 37.6 | 22.2 | 34.9 | 1.57 | 193.0 | 9.8 | 6.7 | 164.3 | -12.1 |
| Rest14 | ACBS-only | 13.98 | **116.9** | **8.4** | **9.1** | **1.08** | 192.1 | 8.9 | 0.7 | **0.0** | **173.1** |
| Rest14 | Full | 14.90 | 58.3 | 17.6 | 20.4 | 1.16 | 196.8 | 13.6 | 7.3 | 109.5 | 36.2 |
| Rest15 | Encoder-only | **13.37** | **126.1** | **7.4** | **8.2** | 1.11 | **184.4** | **1.2** | **0.0** | **0.0** | **0.0** |
| Rest15 | DORA-only | 14.23 | 39.8 | 19.0 | 36.7 | 1.93 | 193.0 | 9.8 | 6.4 | 156.8 | -68.4 |
| Rest15 | ACBS-only | 13.46 | 116.7 | 8.5 | 8.9 | **1.05** | 192.1 | 8.9 | 0.7 | 14.9 | -7.5 |
| Rest15 | Full | 14.31 | 54.6 | 18.0 | 22.1 | 1.23 | 196.8 | 13.6 | 7.0 | 143.2 | -56.7 |
| Rest16 | Encoder-only | **14.48** | **107.3** | **8.1** | **14.1** | 1.74 | **184.4** | **1.2** | **0.0** | **0.0** | **0.0** |
| Rest16 | DORA-only | 15.48 | 52.5 | 18.2 | 22.3 | 1.23 | 193.0 | 9.8 | 6.9 | 124.7 | -51.1 |
| Rest16 | ACBS-only | 14.58 | 68.5 | 14.3 | 16.4 | **1.15** | 192.1 | 8.9 | 0.7 | 76.5 | -36.2 |
| Rest16 | Full | 15.58 | 24.3 | 23.3 | 131.9 | 5.66 | 196.8 | 13.6 | 7.6 | 187.7 | -77.4 |

Table 14: Seed-wise MF1/Acc for paired tests (Full vs. Enc) under 3 seeds. Δ is reported in percentage points (pp).

| | | MF1 | | | Acc | | |
|---|---|---|---|---|---|---|---|
| Dataset | Seed | Full | Enc | Δ (pp) | Full | Enc | Δ (pp) |
| Lap14 | 42 | 81.22 | 75.36 | 5.86 | 84.25 | 79.21 | 5.04 |
| Lap14 | 123 | 81.73 | 75.18 | 6.55 | 84.41 | 79.37 | 5.04 |
| Lap14 | 456 | 81.71 | 74.55 | 7.16 | 84.57 | 78.90 | 5.67 |
| Rest14 | 42 | 84.57 | 76.43 | 8.14 | 89.70 | 84.24 | 5.46 |
| Rest14 | 123 | 85.42 | 77.52 | 7.90 | 89.97 | 85.05 | 4.92 |
| Rest14 | 456 | 84.63 | 75.05 | 9.58 | 89.62 | 84.15 | 5.47 |
| Rest15 | 42 | 73.39 | 70.10 | 3.29 | 89.11 | 85.42 | 3.69 |
| Rest15 | 123 | 75.90 | 72.10 | 3.80 | 90.41 | 86.90 | 3.51 |
| Rest15 | 456 | 72.89 | 71.97 | 0.92 | 88.01 | 85.24 | 2.77 |
| Rest16 | 42 | 86.11 | 78.35 | 7.76 | 95.09 | 91.49 | 3.60 |
| Rest16 | 123 | 82.83 | 73.37 | 9.46 | 94.60 | 90.83 | 3.77 |
| Rest16 | 456 | 84.19 | 75.57 | 8.62 | 94.93 | 90.67 | 4.26 |

whereby large p95/p50 ratios often reflect sensitivity to system-level variance (*e.g.*, cache state and kernel scheduling) rather than algorithmic instability. The deployment-relevant takeaway is thus not the $M$=1 regime, but the multi-aspect regime where encoder+DORA is computed once per sentence and reused across aspects (Figure 2 in the main paper).

## D Significance Tests and Seed-level Scores

Table 14 reports per-seed MF1/Acc for DABS (Full) and the encoder-only baseline. Across all four datasets, the seed-wise differences were predominantly positive, aligning with the aggregate improvements reported in Table 2 (in the main paper). These aligned scores were used for paired statistical tests in the main paper.

## E Additional Multilingual Analyses

### E.1 Per-seed Multilingual Results

Table 15 lists per-seed results for all configurations on French, Russian, and Spanish. ACBS-only delivered the strongest single-component improvements across languages, while the full model was the most stable overall. This pattern suggests that the aspect-conditioned readout in ACBS transfers well cross-lingually, whereas DORA primarily functions as a reusable depth substrate that supports single-pass reuse and depth-ordered querying in the full system.

### E.2 Depth Controls in the Multilingual Setting

We evaluated whether depth allocation behaved as structured specialization rather than incidental layer mixing by applying inference-time depth controls within the last $K$=6 layers. These controls probe whether performance is sensitive to *which* depth band is available, rather than benefiting from arbitrary layer mixing. If depth allocation is functionally meaningful, restricting access to certain bands should change MF1 in a structured way. We observe that the preferred depth region can vary across languages (Table 16), which is plausible under domain and distribution shifts. Rand-2L in Table 17 serves as a sanity-check baseline: it quantifies the expected variability when keeping a random

Table 15: Per-seed multilingual results (Acc/MF1) across 3 seeds for French, Russian, and Spanish.

| | French | | | | | | Russian | | | | | | Spanish | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc (%) | | | MF1 (%) | | | Acc (%) | | | MF1 (%) | | | Acc (%) | | | MF1 (%) | | |
| **Config (seed)** | 42 | 123 | 456 | 42 | 123 | 456 | 42 | 123 | 456 | 42 | 123 | 456 | 42 | 123 | 456 | 42 | 123 | 456 |
| Baseline | 85.08 | 86.31 | 85.54 | 74.01 | 74.73 | 75.27 | 83.93 | 84.04 | 85.52 | 72.76 | 73.06 | 74.06 | 88.86 | 88.43 | 89.56 | 72.05 | 69.65 | 73.50 |
| DORA-only | 84.77 | 85.38 | 84.92 | 74.11 | 74.00 | 73.80 | 85.10 | 84.46 | 85.10 | 73.16 | 72.56 | 72.88 | 87.73 | 89.28 | 87.87 | 68.42 | 73.28 | 72.04 |
| ACBS-only | 89.54 | 88.31 | 87.85 | 80.97 | 77.65 | 78.88 | 88.69 | 87.95 | 89.22 | 78.22 | 75.74 | 79.29 | 92.95 | 91.96 | 93.23 | 78.18 | 76.78 | 78.95 |
| Full | 90.31 | 88.77 | 88.77 | 82.18 | 79.51 | 81.12 | 89.11 | 89.75 | 89.32 | 79.95 | 79.99 | 79.72 | 94.08 | 92.95 | 92.38 | 81.70 | 79.39 | 79.07 |

Table 16: Depth-region masking on multilingual benchmarks using best checkpoints. Each condition retains one 2-layer band within the last $K$=6 layers. $\Delta$ is the best-worst MF1 gap (%).

| Lang | Config | Base MF1 | Shallow ($L_{7\text{-}8}$) | Middle ($L_{9\text{-}10}$) | Deep ($L_{11\text{-}12}$) | $\Delta$(Best-Worst) | Best Region |
|---|---|---|---|---|---|---|---|
| Spanish | ACBS-only | 78.95 | 78.82 | 79.21 | 79.85 | 1.03 | Deep |
| Spanish | Full | 81.70 | 81.29 | 81.37 | 81.70 | 0.41 | Deep |
| French | ACBS-only | 80.97 | 80.97 | 80.94 | 80.05 | 0.92 | Shallow |
| French | Full | 82.18 | 82.32 | 81.67 | 81.42 | 0.90 | Shallow |
| Russian | ACBS-only | 79.29 | 78.77 | 78.78 | 79.40 | 0.63 | Deep |
| Russian | Full | 79.99 | 80.23 | 80.02 | 80.73 | 0.71 | Deep |

contiguous band, making best/worst single-layer effects easier to interpret.

#### E.2.1 Region Masking

Table 16 reports MF1 under shallow/middle/deep region masking, where each condition retained a contiguous 2-layer band within the last $K$ layers. The best-worst gap $\Delta$ indicates that performance depended on which depth band remained, and that the preferred band varied by language and configuration.

#### E.2.2 Single-layer Controls

Table 17 further breaks this down by retaining a single layer at a time, and compares against a randomized contiguous 2-layer baseline (Rand-2L). The consistent best/worst layer differences provide additional evidence that certain layers serve as stronger evidence sources than others, supporting the functional relevance of depth allocation under multilingual transfer.

## F Additional Ablations

### F.1 Accuracy Ablations

Table 18 reports $\mathrm{Acc}_{\mathrm{mean}}$ under the same ablation settings as the MF1 ablations in Table 8. The accuracy trends closely mirrored MF1: removing ACBS components (especially token selection and gated fusion) produced the most consistent degradation, while removing DORA components led to smaller but still noticeable drops. The alignment across metrics suggests that the gains are not artifacts of a particular evaluation measure.

## G Cross-Dataset Stress-Test Splits and Depth-Order Controls

This appendix reports controlled analyses on three cross-dataset stress-test splits designed to emphasize sentence-level difficulty patterns. The goal is to test whether the ordered depth integration in DORA behaves as a functional mechanism, rather than a cosmetic architectural choice.

### G.1 Split Construction

We pooled training splits from SemEval datasets (Lap14, Rest14, Rest15, Rest16) into a single training set, and pooled their test splits into a single test set. From each pool, we constructed three stress-test subsets using explicit filtering rules:

- **Long Sentences.** We computed subword length using the DeBERTa tokenizer and retained sentences at or above the P90 length threshold of the corresponding pool.
- **Multi-Aspect Conflict.** We retained sentences containing at least one positive aspect and at least one negative aspect within the same sentence.
- **Complex Negation.** We retained sentences that contained a negation cue (*e.g.*, *no*, *not*, *never*, *n't*, *without*) and had subword length greater than 40.

Unless stated otherwise, results in this appendix used $K$=6 and report mean and standard deviation over $N$=3 seeds.

Table 19 reports class distributions for the three stress-test splits. Counts are aspect-level instances. The Multi-Aspect Conflict split contained very few

Table 17: Single-layer depth controls on multilingual benchmarks using best checkpoints. Rand-2L keeps a random contiguous 2-layer block (20 trials) and $\Delta$ are relative to Base.

| Lang | Config | Base (Acc/MF1) | Rand-2L MF1 ($\mu \pm \sigma$, $\Delta$) | Best-1L (layer, MF1, $\Delta$) | Worst-1L (layer, MF1, $\Delta$) |
|---|---|---|---|---|---|
| Spanish | ACBS-only | 93.23/78.95 | 79.22±0.37 (↑0.27) | d4: 79.52 (↑0.57) | d1: 78.64 (↓0.31) |
| Spanish | Full | 94.08/81.70 | 81.61±0.16 (↓0.09) | d4: 82.17 (↑0.47) | d6: 81.29 (↓0.41) |
| French | ACBS-only | 89.54/80.97 | 80.63±0.38 (↓0.34) | d1: 81.73 (↑0.76) | d3: 79.54 (↓1.43) |
| French | Full | 90.31/82.18 | 81.79±0.30 (↓0.39) | d2: 82.55 (↑0.37) | d5: 81.42 (↓0.76) |
| Russian | ACBS-only | 89.22/79.29 | 78.96±0.27 (↓0.33) | d6: 79.33 (↑0.04) | d3: 78.65 (↓0.64) |
| Russian | Full | 89.75/79.99 | 80.19±0.19 (↑0.20) | d1: 80.65 (↑0.66) | d3: 79.84 (↓0.15) |

Table 18: Ablations on $\mathrm{Acc}_{\mathrm{mean}}$. Values are reported as $\mathrm{Acc}^{\Delta}$, where $\Delta$ is the change (pp) from the full model.

| Config | Lap14 $\mathrm{Acc}^{\Delta}$ | Rest14 $\mathrm{Acc}^{\Delta}$ | Rest15 $\mathrm{Acc}^{\Delta}$ | Rest16 $\mathrm{Acc}^{\Delta}$ | Avg $\Delta$ |
|---|---|---|---|---|---|
| **DABS (Full)** | **84.41**$^{0.00}$ | **89.76**$^{0.00}$ | **89.18**$^{0.00}$ | **94.87**$^{0.00}$ | 0.00 |
| *w/o Regularizers* | | | | | |
| - Sparsity | 82.94$^{\downarrow 1.47}$ | 89.05$^{\downarrow 0.71}$ | 88.50$^{\downarrow 0.68}$ | 94.33$^{\downarrow 0.54}$ | ↓0.85 |
| - Span Masking | 82.62$^{\downarrow 1.79}$ | 88.72$^{\downarrow 1.04}$ | 88.87$^{\downarrow 0.31}$ | 94.76$^{\downarrow 0.11}$ | ↓0.81 |
| - Gate Entropy | 82.52$^{\downarrow 1.89}$ | 89.49$^{\downarrow 0.27}$ | 89.18$^{0.00}$ | 94.33$^{\downarrow 0.54}$ | ↓0.68 |
| *w/o DORA* | | | | | |
| - DepthGRU | 83.57$^{\downarrow 0.84}$ | 89.05$^{\downarrow 0.71}$ | 88.75$^{\downarrow 0.43}$ | 94.16$^{\downarrow 0.71}$ | ↓0.67 |
| - LCP (Pooling) | 82.83$^{\downarrow 1.58}$ | 88.63$^{\downarrow 1.13}$ | 88.50$^{\downarrow 0.68}$ | 93.84$^{\downarrow 1.03}$ | ↓1.11 |
| *w/o ACBS* | | | | | |
| - Token Sel. | 82.05$^{\downarrow 2.36}$ | 88.84$^{\downarrow 0.92}$ | 88.01$^{\downarrow 1.17}$ | 93.94$^{\downarrow 0.93}$ | ↓1.35 |
| - Layer Sel. | 83.31$^{\downarrow 1.10}$ | 88.27$^{\downarrow 1.49}$ | 87.95$^{\downarrow 1.23}$ | 94.38$^{\downarrow 0.49}$ | ↓1.08 |
| - Gated Fusion | 82.26$^{\downarrow 2.15}$ | 87.94$^{\downarrow 1.82}$ | 88.56$^{\downarrow 0.62}$ | 94.33$^{\downarrow 0.54}$ | ↓1.28 |

Table 19: Class distributions for cross-dataset stress-test splits. Counts are aspect-level instances in train/test.

| **Dataset** | **Positive** | | **Neutral** | | **Negative** | |
|---|---|---|---|---|---|---|
| | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| Long Sentences | 677 | 274 | 282 | 78 | 538 | 156 |
| Multi-Aspect Conflict | 372 | 96 | 38 | 6 | 368 | 96 |
| Complex Negation | 116 | 29 | 44 | 8 | 154 | 40 |

neutral instances, which can increase variance in macro-F1.

## G.2 Layer-Order Ablation ($K$=6)

DORA integrates the last $K$ layers using a depth-ordered recurrence. If the depth order is meaningful, disrupting that order should reduce performance under stress-test conditions. We used a DeBERTa-base backbone for this controlled study. We evaluated three conditions: (i) normal order, (ii) reversed order, and (iii) shuffled order. Table 20 reports mean±std over three seeds, with $\Delta$MF1 computed relative to the normal order within each split.

DepthGRU is explicitly *depth-ordered*. It accumulates representations from shallower to deeper layers, encouraging monotonic refinement from lexical/local cues to more abstract compositional features. Reversing the order forces the recurrence to integrate abstractions before grounding signals, which can blur compositional dependencies and reduce robustness under long-context or negation-heavy conditions. Shuffling disrupts the monotonicity less systematically than full reversal, which is consistent with the smaller average degradation observed under the shuffled condition.

Table 20: Layer-order ablation on cross-dataset stress-test splits ($K$=6). Results are mean±std over 3 seeds. $\Delta$MF1 is relative to the normal order (pp).

| **Dataset** | **Order** | **Acc** | **MF1** | **$\Delta$MF1** |
|---|---|---|---|---|
| Long Sentences | normal | 82.74$_{\pm 1.08}$ | 77.97$_{\pm 0.72}$ | 0.00 |
| | reversed | 82.26$_{\pm 0.60}$ | 76.44$_{\pm 0.21}$ | -1.53 |
| | shuffled | 82.19$_{\pm 1.46}$ | 77.22$_{\pm 1.39}$ | -0.75 |
| Multi-Aspect Conflict | normal | 95.40$_{\pm 1.15}$ | 76.88$_{\pm 10.80}$ | 0.00 |
| | reversed | 95.79$_{\pm 0.88}$ | 74.10$_{\pm 9.55}$ | -2.78 |
| | shuffled | 95.40$_{\pm 1.15}$ | 74.28$_{\pm 9.77}$ | -2.60 |
| Complex Negation | normal | 83.81$_{\pm 2.18}$ | 80.62$_{\pm 1.49}$ | 0.00 |
| | reversed | 83.33$_{\pm 2.97}$ | 79.14$_{\pm 1.19}$ | -1.48 |
| | shuffled | 83.81$_{\pm 2.18}$ | 80.10$_{\pm 1.55}$ | -0.52 |

Table 21: DepthGRU ablation on cross-dataset stress-test splits ($K$=6). Results are mean±std over 3 seeds (Acc/MF1, %).

| **Dataset** | **Setting** | **Acc** | **MF1** |
|---|---|---|---|
| Long Sentences | w DepthGRU | 82.95$_{\pm 0.91}$ | 78.44$_{\pm 0.76}$ |
| | w/o DepthGRU | 82.61$_{\pm 1.57}$ | 76.93$_{\pm 2.14}$ |
| Multi-Aspect Conflict | w DepthGRU | 95.40$_{\pm 1.15}$ | 76.88$_{\pm 10.80}$ |
| | w/o DepthGRU | 94.83$_{\pm 0.57}$ | 71.32$_{\pm 6.29}$ |
| Complex Negation | w DepthGRU | 83.81$_{\pm 2.18}$ | 80.62$_{\pm 1.49}$ |
| | w/o DepthGRU | 82.86$_{\pm 2.86}$ | 77.69$_{\pm 3.36}$ |

## G.3 DepthGRU Ablation

This ablation isolates the contribution of DepthGRU inside DORA by removing the recurrence while keeping the remaining components unchanged. We used a DeBERTa-base backbone with $K$=6 and $N$=3 seeds. Table 21 compares the full

model (Base) against the variant without DepthGRU (w/o DepthGRU).

Removing DepthGRU reduced macro-F1 across all three stress-test splits, with the largest drop on Multi-Aspect Conflict and Complex Negation. Together with the layer-order ablation, these results support that ordered depth integration contributes materially under conditions that emphasize long context, polarity conflict, and negation-driven composition.

We observe that accuracy can remain relatively stable while macro-F1 drops, especially on conflict/negation splits. This pattern suggests that DepthGRU mainly improves *minority-class* and *hard-case* discrimination (which MF1 is sensitive to), rather than only boosting majority predictions. This is consistent with our hypothesis that ordered depth integration provides additional compositional capacity that matters most when superficial lexical cues are insufficient.

## H AI Assistants

AI assistants were used solely to support language editing and clarity, including minor rephrasing and grammatical corrections. All technical content, modeling decisions, experimental design, results, and interpretations were conceived, implemented, and verified by the authors. No AI system was used to generate experimental results, analyze data, or make scientific claims.