From Graphs to Hypergraphs: Enhancing Aspect-Based Sentiment Analysis via Multi-Level Relational Modeling

Anonymous EACL submission

Abstract

Aspect-Based Sentiment Analysis (ABSA) predicts sentiment polarity for specific aspect terms, a task made difficult by conflicting sentiments across aspects and the sparse context of short texts. Prior graph-based approaches model only pairwise dependencies, forcing them to construct multiple graphs for different relational views. These introduce redundancy, parameter overhead, and error propagation during fusion, limiting robustness in short-text, low-resource settings. We present HyperABSA, a dynamic hypergraph framework that induces aspect-opinion structures through sample-specific hierarchical clustering. To construct these hyperedges, we introduce a novel acceleration-fallback cutoff for hierarchical clustering, which adaptively determines the level of granularity. Experiments on three benchmarks (Lap14, Rest14, MAMS) show consistent improvements over strong graph baselines, with substantial gains when paired with RoBERTa backbones. These results position dynamic hypergraph construction as an efficient, powerful alternative for ABSA, with potential extensions to other short-text NLP tasks.

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) is a family of tasks that predict sentiment polarity with respect to specific aspects in text (Zhang et al., 2022b). In this work, we focus on the most widely studied subtask, Aspect Term Sentiment Analysis (ATSA), where given a sentence and one or more marked aspect terms, the goal is to predict the sentiment polarity for each aspect. For instance, in the sentence *"Service is good although a bit in your face , we were asked every five mins if food was ok, but better that than being ignored"*, the aspects "service" and "food" reflect positive and neutral sentiments, respectively. Such fine-grained opinion modeling is central to applications like product reviews, customer feedback, and social media monitoring.

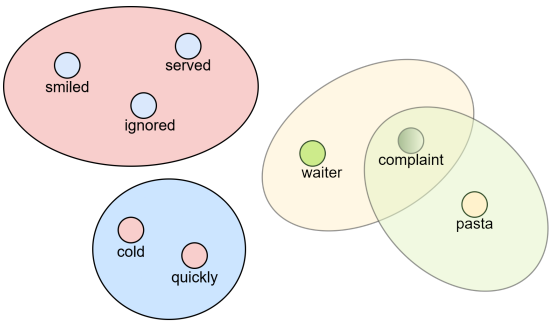

Figure 1: A hypergraph of word interactions showing several semantic clusters based on aspect and sentiment polarity. This illustrates how words are grouped according to meaning and sentiment.

Early work approached ATSA by incorporating dependency trees to capture syntactic relations between aspects and opinion words (Poria et al., 2014; Chen et al., 2022). More recently, graph neural networks (GCNs) have been widely adopted to propagate sentiment information along these structures (Kipf and Welling, 2016; Bai et al., 2020; Liang et al., 2020; Tian et al., 2021; Zhang et al., 2022c). However, a fundamental limitation of these techniques lies in their inherent focus on pairwise relationships, because a single graph can only encode one type of relation at a time and cannot flexibly capture the multiple, overlapping relational views that matter in ABSA.

This gap has led to multi-graph architectures (Li et al., 2021; Aziz et al., 2024; Zheng and Li, 2024), which capture different facets of text, such as syntactic dependencies and semantic relationships, and then attempt to fuse information from these disparate graph sources. While effective in fusing information, these methods introduce redundancy and parameter overhead, require sophisticated fusion mechanisms and are vulnerable to error propagation from parser outputs, particularly

problematic in short texts where parser reliability is low. Such complexity makes them difficult to train robustly, especially in short-text, low-resource ABSA settings where data scarcity already limits generalization.

We address these challenges with HyperABSA, a dynamic hypergraph framework that unifies multiple graph views into a single adaptive structure. Unlike standard graphs that capture only pairwise relations, hypergraphs group tokens into hyperedges, directly modeling aspect–opinion structures and eliminating redundant fusion steps (Zhang et al., 2022a), as seen in Figure 1. Our key contribution is an adaptive hypergraph construction method for ABSA, where each sentence undergoes hierarchical agglomerative clustering (HAC) on token embeddings.The dendrogram is cut using an acceleration-fallback criterion that detects "elbows" in inter-cluster dissimilarities, signaling structural boundaries. When this signal is weak, a variance-sensitive threshold is used instead, ensuring robustness across sentence lengths. While second-order change detection is established in clustering, this is the first application for per-instance hypergraph induction in ABSA, enabling sentence-specific structural modeling. Experiments on Lap14, Rest14, and MAMS show consistent gains over strong graph baselines, achieving state-of-the-art results with RoBERTa backbones.

This paper makes the following contributions to the field of ABSA:

- We introduce HyperABSA, a hypergraph-based framework that replaces multi-graph fusion with a single adaptive structure, grouping aspects and their opinion modifiers into hyperedges instead of relying on separate pairwise graphs.

- We propose a per-instance dynamic hypergraph construction method using hierarchical clustering with an acceleration-fallback cutoff, which adapts the clustering depth to sentence length, unlike prior global or fixed thresholds.

- Our method achieves state-of-the-art results on Laptop14 and Rest14 when paired with RoBERTa, and competitive performance on MAMS. Extensive ablations across encoders, thresholds, and clustering strategies confirm that the adaptive cutoff and hypergraph design drive the improvements.

2 Related Work

Over the years, ABSA has been widely explored using various methodologies.

2.1 Graph Based Methods

Graph-based models are widely used in ABSA to encode syntactic and semantic relationships among tokens. Early approaches employed graph convolutional networks (GCNs) over dependency trees to model syntactic structure(Chen et al., 2019), with later extensions incorporating both syntactic and semantic dependencies to enrich graph representations (Wang et al., 2020; Chen et al., 2022; Bao et al., 2023).

To capture finer-grained relational cues, subsequent work introduced relational GCNs and type-aware architectures that explicitly model aspect-specific and inter-aspect dependencies (Huang and Carley, 2019; Yuan et al., 2020; Tian et al., 2021). While effective, these approaches depend heavily on external syntactic parsers, which can be brittle in short, noisy sentences. Attention mechanisms were later integrated into graph models, with multi-head attention fused with GCNs to jointly encode syntactic and semantic signals (Xu et al., 2021; Pan et al., 2023; Cui et al., 2023; Zhang et al., 2024).

More recent advances have explored heterogeneous graph structures to represent diverse linguistic relations while preserving their functional roles during sentiment propagation (Zeng et al., 2023; Niu et al., 2022). In parallel, multi-graph frameworks construct complementary graph views to simultaneously encode localized aspect cues and global contextual semantics, enhancing sentiment inference across varied discourse structures (Li et al., 2021; Aziz et al., 2024; Zheng and Li, 2024).

2.2 Hypergraph Construction Methods

While hypergraph models have demonstrated strong potential across domains, their use in aspect-based sentiment analysis (ABSA) remains under-explored. Prior work largely advances hypergraph neural architectures (Feng et al., 2019; Zhi, 2024), with comparatively limited focus on text-driven hypergraph construction. Existing methods often connect tokens via nearest-neighbor relations in feature space (Yu et al., 2012; Gao et al., 2022; Nguyen et al., 2020; Dai and Gao, 2023), which can introduce semantically irrelevant nodes. Topic-based strategies, such as LDA (Ding et al., 2020; Turnbull et al., 2024), improve semantic grouping

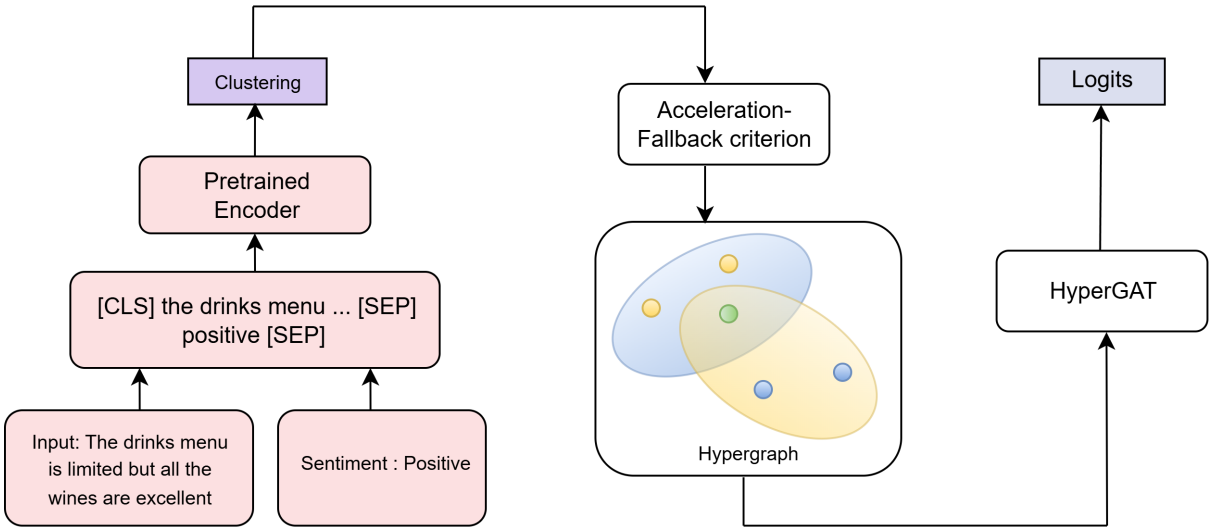

Figure 2: Architecture of HyperABSA.

but depend on latent topic priors, while clustering-based approaches, like K-Means and spectral clustering enhance coherence at the cost of requiring fixed thresholds or cluster counts (Han et al., 1997; Chang et al., 2008; Saito, 2022).

Short-text ABSA presents additional challenges like sparse context, multiple aspects per sentence, and limited supervision, where global construction heuristics often fail. We address these issues with a per-instance adaptive cutoff that dynamically determines clustering depth, ensuring balanced granularity across sentences. Though inspired by classical elbow and change-point detection, our integration of acceleration-based criteria into per-instance hypergraph induction constitutes a novel methodological contribution with clear empirical gains.

3 Methodology

3.1 Hypergraph Construction

Given an input sentence with n tokens, we first encode each token into a contextual representation using a pretrained encoder as seen in Fig 2. Let $H_0 = [\mathbf{h}_1, \ldots, \mathbf{h}_n] \in \mathbb{R}^{n \times d}$, where $\mathbf{h}_i \in \mathbb{R}^d$ is the contextual embedding of token i. These serve as initial node features.

3.2 Hierarchical Clustering of Token Embeddings

To induce higher-order structures, we apply HAC to the token embeddings $\{\mathbf{h}_i\}$, using the linkage function $\ell(\cdot, \cdot)$. HAC iteratively merges the most similar clusters, producing a dendrogram. Each merge step is recorded as $z_t = [c_1, c_2, \delta_t, s_t]$, where c_1, c_2 are the indices of merged clusters, δ_t is the linkage dissimilarity, and s_t is the size of the resulting cluster. Collectively,

$$Z = [z_1, \ldots, z_{n-1}] \in \mathbb{R}^{(n-1)\times 4} \tag{1}$$

is the linkage matrix. The sequence $\{\delta_t\}$ reflects how inter-cluster distances evolve.

3.3 Adaptive Cutoff: Acceleration–Fallback Criterion

A fixed global merge threshold may not adequately account for variations in sentence length and syntactic complexity, potentially leading to suboptimal clustering. Implementing an adaptive cutoff that adjusts per sentence based on recent clustering dynamics could better accommodate these variations.

Let $\rho \in (0, 1]$ control the fraction of merges considered. We compute:

$$r = \max\{1, \lfloor \rho \cdot (n-1) \rfloor\}. \tag{2}$$

and extract the corresponding recent dissimilarities:

$$\boldsymbol{\delta}_{\text{recent}} = [\delta_{n-r}, \ldots, \delta_{n-1}]. \tag{3}$$

If $|\boldsymbol{\delta}_{\text{recent}}| > 3$, we compute second-order finite differences

$$\kappa_j = \delta_{j+2} - 2\delta_{j+1} + \delta_j \quad \text{for } j = n-r, \ldots, n-3. \tag{4}$$

to detect rapid changes in merge compactness. The index

$$j^\star = \arg\max_j \kappa_j \tag{5}$$

marks the most prominent such change and serves as the candidate cutoff point. However, if the recent window is too short or the second-order signal is weak, conditions common in sparse or noisy data where hierarchical structure is less distinct, we adopt a fallback mechanism that dynamically adjusts the cutoff threshold. This variance-based threshold is defined as

$$\delta_{\text{fallback}} = \bar{\delta}_{\text{recent}} + \lambda \cdot \sigma_{\text{recent}}, \quad \lambda > 0. \tag{6}$$

where $\bar{\delta}_{\text{recent}}$ and σ_{recent} denote the mean and standard deviation of the recent dissimilarities, respectively.

The final cutoff is given by:

$$\delta_{\text{elbow}} = \begin{cases} \min\{\delta_{j^\star}, \delta_{\text{fallback}}\}, & |\boldsymbol{\delta}_{\text{recent}}| > 3, \\ \delta_{\text{fallback}}, & \text{otherwise.} \end{cases} \tag{7}$$

We cut the dendrogram at δ_{elbow} to obtain $|E|$ clusters, denoted as $\{C_e\}_{e=1}^{|E|}$ as seen in Fig 3.

We represent these clusters as a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where nodes $\mathcal{V} = 1, \ldots, n$ correspond to tokens, and hyperedges $\mathcal{E} = C_e$ correspond to clusters. The incidence matrix $\mathbf{H} \in 0, 1^{|\mathcal{V}| \times |\mathcal{E}|}$ is defined element-wise as:

$$H_{v,e} = \begin{cases} 1, & \text{if token } v \in C_e, \\ 0, & \text{otherwise.} \end{cases}$$

3.4 Hypergraph Neural Network

To propagate information over the hypergraph structure, we apply a Hypergraph Attention Network inspired by HyperGAT (Ding et al., 2020). Each hyperedge aggregates features from its incident nodes using multi-head attention. Let $X \in \mathbb{R}^{B \times n \times d}$ be the node representations for a batch of size B, where n is the number of nodes and d the feature dimension. For each attention head $h \in \{1, \ldots, H\}$, we compute:

$$A^{(h)} = \text{softmax}\left(H^\top \cdot \sigma\left(X W_1^{(h)}\right)\right) \tag{8}$$

where $H \in \{0, 1\}^{B \times n \times |E|}$ is the batch of incidence matrices, $W_1^{(h)} \in \mathbb{R}^{d \times d}$ is a learnable weight matrix for head h, σ is an activation function and softmax is applied over the node dimension to normalize contributions within each hyperedge.

Each hyperedge then computes a weighted sum of its incident node features as:

$$E_{b,e,:}^{(h)} = \sum_{i=1}^{n} H_{b,i,e} \cdot \alpha_{b,e,i}^{(h)} \cdot X_{b,i,:} \tag{9}$$

where $\alpha_{b,e,i}^{(h)}$ is the attention weight from $A^{(h)}$ for sample b, hyperedge e, and node i.

The outputs from all attention heads are concatenated to form the hyperedge features:

$$E = \text{Concat}\left(E^{(1)}, \ldots, E^{(H)}\right) \in \mathbb{R}^{B \times |E| \times d'},$$

where $d' = H \times d$.

Finally, the hyperedge features are pooled by averaging over all hyperedges and passed through a feed-forward linear projection to obtain logits:

$$\hat{y} = W \cdot \frac{1}{|E|} \sum_{e=1}^{|E|} E_{:,e,:}$$

where $W \in \mathbb{R}^{d' \times C}$ maps the aggregated hyperedge features to C output classes. The algorithm for our approach is presented in Appendix .

3.5 Loss Function

Given the final aspect-specific representation $\mathbf{h}_a$ from the hypergraph network, sentiment classification is performed via a softmax layer that maps $\mathbf{h}_a$ to a categorical distribution over sentiment labels:

$$P(y \mid \mathbf{h}_a) = \text{softmax}(W_c \mathbf{h}_a + b_c), \tag{10}$$

where $W_c \in \mathbb{R}^{|\mathcal{C}| \times d}$ and $b_c \in \mathbb{R}^{|\mathcal{C}|}$ are learnable parameters, and $\mathcal{C}$ denotes the sentiment classes.

The model is trained using cross-entropy loss with ℓ_2 regularization:

$$\mathcal{L} = -\sum_{i=1}^{N} \sum_{c \in \mathcal{C}} \mathbb{I}(y_i = c) \log P(y_i \mid \mathbf{h}_a^{(i)}) + \beta \|\Theta\|_2^2, \tag{11}$$

where Θ includes all trainable parameters and β controls weight decay. This objective promotes confident and accurate predictions while reducing overfitting.

4 Experiments

4.1 Implementation Details

For fair comparison, we follow the preprocessing pipeline of RGAT (Wang et al., 2020). Unlike RGAT, which depends on external syntactic features (POS tags, dependency relations, and head indices), our model uses only raw sentence tokens

Model	Lap14		Rest14		MAMS	
	Acc(%)	F1(%)	Acc(%)	F1(%)	Acc(%)	F1(%)
InterGCN† (Liang et al., 2020)	78.06	73.83	85.45	77.64	82.49	81.95
R-GAT‡ (Wang et al., 2020)	78.21	74.07	84.64	77.14	83.16	82.42
DGEDT‡ (Tang et al., 2020)	79.80	75.60	86.30	80.00	-	-
RGAT‡ (Bai et al., 2020)	80.31	76.38	85.77	79.81	82.96	82.12
DMGLT (Fang, 2022)	78.82	75.56	86.25	79.04	-	-
RMN (Zeng et al., 2022)	77.95	70.839	84.56	79.05	79.97	78.7
CHGMAN‡ (Niu et al., 2022)	78.04	74.46	85.98	79.31	83.23	82.66
MWGCN (Yu and Zhang, 2023)	79.78	76.68	86.36	80.54	-	-
HGCN (Xu et al., 2023)	79.59	76.24	86.45	80.60	-	-
UIKA-BERT§ (Liu et al., 2023)	76.98	72.55	81.65	75.21	-	-
MTABSA-BERT§ (Zhao et al., 2023)	77.43	72.07	83.39	75.09	-	-
MLEGCN (Aziz et al., 2024)	-	69.74	-	79.94	-	-
SSIN (Wang et al., 2024)	79.68	77.04	85.07	78.11	-	-
YORO‡(Zheng and Li, 2024)	77.45	73.21	83.69	76.22	84.21	**83.78**
CaBiLSTM-BERT (He et al., 2025)	77.91	73.04	83.75	75.87	-	-
HyperABSA	**83.44**	**80.6**	**87.77**	**82.757**	**84.56**	83.74

Table 1: Performance of Accuracy and F1 score of HyperABSA with other models. † denotes implementation from (Zheng and Li, 2024), ‡ denotes our implementation, and § denotes implementation from (He et al., 2025) respectively

Method Variant (with Formula)	ρ	Rest14		Lap14		MAMS	
		Acc. (%)	F1 (%)	Acc. (%)	F1 (%)	Acc. (%)	F1 (%)
HyperABSA (Equation 7)	Dynamic	**86.67**	**80.64**	**80.46**	**77.42**	**84.56**	**83.74**
$\delta_{\text{elbow}} = \delta_{fallback}$	-	84.07	76.89	79.06	75.84	84.00	83.51
$\delta_{\text{elbow}} = \delta_{n-m+k}$	0.2	80.59	71.61	79.68	76.30	83.48	82.82
	0.5	83.11	74.75	78.13	74.88	83.48	82.87
	0.8	82.12	74.60	77.03	73.18	83.55	82.90
$\delta_{\text{elbow}} = \min(\delta_{n-m+k}, \delta_{fallback})$	0.2	84.78	77.35	79.22	77.14	84.22	83.46
	0.5	80.95	72.06	78.75	75.95	84.07	83.24
	0.8	84.98	78.24	79.53	76.03	83.70	83.09

Table 2: Ablation study on Rest14, Lap14, and MAMS showing the impact of acceleration formula and proportion (p) on HyperABSA's performance. Formula types are indicated in parentheses within the method name.

and aspect markers. All tokens, including special boundary or aspect delimiters, are passed to both the encoder and the clustering stage for hypergraph construction. Clustering is performed using Ward's method with Euclidean distance on ℓ_2-normalized token embeddings.

We evaluate HyperABSA on three standard ABSA benchmarks: MAMS (Jiang et al., 2019), SemEval-2014 Restaurants (Rest14), and Laptops (Lap14) (Pontiki et al., 2014), following the data splits of Bai et al. (2020). The text encoder is RoBERTa-base, chosen for its superior empirical performance. Dropout with rates in $[0.2, 0.3]$ is applied to both encoder and hypergraph layers, and ℓ_2 regularization ($\beta = 2 \times 10^{-5}$) is used for weight decay. The clustering sparsity parameter ρ and the variance-sensitivity coefficient are treated as trainable parameters. Training employs the Adam optimizer (Kingma, 2014) with a learning rate of 10^{-2}, batch size of 16, and is conducted on a single NVIDIA RTX 4090 GPU.

4.2 Baselines

We benchmark HyperABSA against a comprehensive set of recent ABSA models, encompassing both classical baselines and graph-based architectures. Our comparisons emphasize methods leveraging syntactic dependencies and GCN variants, including InterGCN (Liang et al., 2020), R-GAT (Wang et al., 2020), DGEDT (Tang et al., 2020), RGAT (Bai et al., 2020), DMGLT (Fang, 2022), RMN (Zeng et al., 2022), CHGMAN (Niu et al., 2022), MWGCN (Yu and Zhang, 2023), HGCN (Xu et al., 2023), UIKA-BERT (Liu et al., 2023), MTABSA-BERT (Zhao et al., 2023), SSIN (Wang et al., 2024) and recent multi-graph models such as YORO (Zheng and Li, 2024), and MLEGCN (Aziz et al., 2024). To ensure fair comparison and reproducibility, we restrict evaluation to publicly available implementations.

4.3 Results

As shown in Table 1, HyperABSA consistently outperforms strong baselines across all three benchmark datasets. On the Lap14 dataset, it achieves a 7% improvement in F1 score over CaBiLSTM-BERT, a recent 2025 baseline, and outperforms models such as YORO and RMN with an average gain of 5% in accuracy. For the Rest14 dataset, HyperABSA obtains the highest accuracy and F1 score, surpassing YORO, MTABSA-BERT, R-GAT and RMN by an average margin of 4-5%. On the more challenging MAMS dataset, HyperABSA outperforms InterGCN, DMGLT, and CHGMAN by an average of 2% in both accuracy and F1, while remaining competitive with YORO in terms of F1 score. A comparison between HyperABSA and several LLM-based baselines on the Lap14 and Rest14 datasets is present in

5 Experimental Details

5.1 Comparison with other hypergraph construction methods

Table 3 compares different clustering methods for hypergraph construction on Lap14 and Rest14 datasets. Density-based clustering (DBSCAN) (Ester et al., 1996) performs poorly, likely due to the sparse and uneven distribution of aspect–opinion tokens. In contrast, Gaussian Mixture Models (GMM), KMeans, and random initialization achieve high accuracy and F1, with differences of only about 0.1 in most cases, indicating that hypergraph-based ABSA is robust to small variations in clustering. Notably, Ward linkage with Euclidean distance (our approach) produces results comparable to or slightly better than GMM and KMeans, confirming that variance-aware hierarchical clustering yields stable and semantically meaningful hypergraph structures. RGAT, which does not use clustering, slightly underperforms on Lap14, further highlighting the benefit of our cluster-based hypergraph construction.

Model	**Lap14**		**Rest14**	
	Acc(%)	F1(%)	Acc(%)	F1(%)
DBSCAN	47.30	27.86	52.87	47.92
GMM	81.40	78.29	**88.12**	80.42
KMeans	81.71	**80.67**	85.28	81.26
Random	81.25	78.50	83.39	80.06
No clustering	81.81	78.87	84.29	80.32
RGAT	80.31	76.38	85.77	79.81
HyperABSA	**83.44**	80.60	87.77	**82.76**

Table 3: Performance (%) of different encoders on Lap14 and Rest14 datasets.

5.2 Analysis of stop words and special tokens

To evaluate the effect of token-level preprocessing on model performance, we perform ablations by selectively removing different categories of special tokens. Specifically, we remove stop words identified by the NLTK tokenizer (Bird, 2006) before clustering, and test a variant that excludes tokenizer-specific special tokens such as `[CLS]` and `[SEP]` for BERT, and `<s>` and `</s>` for RoBERTa.

As shown in Table 4, removing NLTK stop words substantially degrades performance across all datasets, highlighting their importance in maintaining syntactic and semantic coherence during hypergraph construction. The best results are achieved when both sentence-level markers and tokenizer-specific special tokens are retained, suggesting that these contextual cues enable the clustering stage to capture clearer structural boundaries. Overall, this analysis demonstrates that token-level contextual markers play a critical role in effective hypergraph induction and sentiment classification.

	Lap14		**Rest14**		**MAMS**	
	Acc(%)	F1(%)	Acc(%)	F1(%)	Acc(%)	F1(%)
No special tokens	80.47	78.46	86.05	79.58	82.39	80.45
No stop words	75.16	72.06	80.79	74.04	75.16	72.06
Including both	**83.44**	**80.60**	**87.76**	**82.76**	**82.51**	**82.12**

Table 4: Ablation study on token-level preprocessing in HYPERABSA

5.3 Embedding Analysis

We evaluate the impact of different text encoders on HyperABSA to assess the robustness of our framework across backbone architectures. As shown in Table 5, RoBERTa consistently outperforms BERT and XLNet across all three datasets (Lap14, Rest14, and MAMS) in both accuracy and F1 score. The improvements are particularly pronounced on the Rest14 dataset, indicating that RoBERTa's stronger contextual representations enhance the quality of token embeddings used for hypergraph construction.

Model	Lap14		Rest14		MAMS	
	Acc(%)	F1(%)	Acc(%)	F1(%)	Acc(%)	F1(%)
RoBERTa	**83.44**	**81**	**87.76**	**83**	**83.18**	**83**
BERT	79.69	77	86.67	80	84.56	83
XLNet	76.88	74	79.69	77	79.69	77

Table 5: Performance (%) of different encoders on ABSA datasets.

In contrast, XLNet underperforms due to its permutation-based pretraining, which appears less suited for short, aspect-focused sentences typical in ABSA. These results confirm that HyperABSA benefits from high-quality encoder representations but remains flexible and effective across different pretrained backbones.

5.4 Linkage and Distance ablation

We analyze the effect of different linkage criteria and distance metrics used in hierarchical clustering for hypergraph construction. As shown in Table 6, Ward linkage with Euclidean distance achieves the highest accuracy and F1 across both Lap14 and Rest14 datasets, outperforming Single, Complete, and Average linkage methods. Ward's variance-minimizing criterion produces more coherent and balanced clusters, yielding interpretable and sentiment-aligned hyperedges. Single and Complete linkage tend to produce fragmented or overly rigid clusters, while Average linkage overly smooths boundaries, obscuring fine-grained sentiment cues. Although other distance metrics such as cosine perform competitively under Single, Complete, or Average linkage, Euclidean distance remains consistently superior due to its alignment with ℓ_2-normalized token embeddings. Overall, these results confirm that variance-aware hierarchical clustering with Euclidean geometry (Ward + Euclidean) yields the most stable and semantically meaningful hypergraph structures for ABSA.

5.5 Cluster Quality Analysis

We evaluate the effectiveness of our hypergraph construction method by comparing it against (i) a Random hypergraph, where nodes and hyperedges are formed without structural guidance, and (ii) a KNN-KMeans hybrid hypergraph that combines local and global relational signals via K-Nearest Neighbors and K-Means clustering. Cluster quality is assessed using the Silhouette Score (Rousseeuw, 1987), which captures intra-cluster cohesion and inter-cluster separation (higher is better), and the Davis-Bouldin Score (Davies and Bouldin, 1979), which quantifies average inter-cluster similarity (lower is better), across different training epochs.

As shown in Table 7, HyperABSA consistently outperforms both baselines. Random hypergraphs, due to their stochastic construction, fail to produce meaningful clusters, often resulting in negative Silhouette Scores. The KNN-KMeans hybrid, while incorporating local and global structural cues, still falls short of HyperABSA in clustering quality. These results highlight that our method preserves both structural and semantic coherence across training epochs. Results regarding the complexity of the construction process in Sec: Appendix.

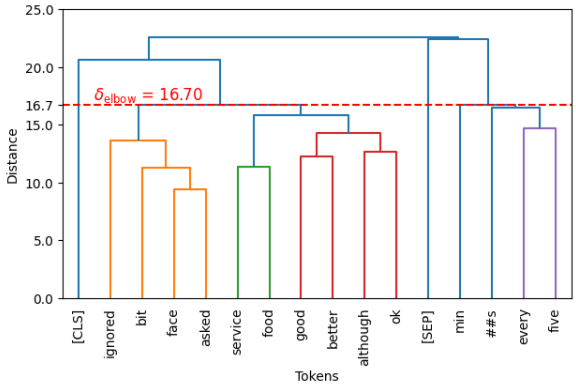

Figure 3: Dendrogram visualization of token-level hierarchical clustering.

5.6 Multi Granular Approach of Hypergraph

To explore whether a dynamically constructed hypergraph can serve as a viable alternative to manually designed multi-graph architectures for multi-granular reasoning, we conduct a series of comparative experiments. We compare our dynamic hypergraph approach with several fixed-granularity baselines, including models with only fallback connections (coarse granularity), and acceleration paths with static thresholds (ρ = 0.2, 0.5, 0.8). As seen

	Single				Complete				Average				Ward			
	Lap14		Rest14		Lap14		Rest14		Lap14		Rest14		Lap14		Rest14	
Metric	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**
Euclidean	81.45	78.85	86.51	80.52	82.56	79.96	86.68	81.65	82.34	79.35	83.44	80.60	**83.44**	**80.60**	**87.77**	**82.757**
Cosine	81.88	79.49	84.10	79.89	82.44	78.85	83.85	78.95	81.23	79.87	84.20	80.67	**82.67**	**79.94**	-	-

Table 6: Clustering results (Accuracy and F1 %) for Lap14 and Rest14 datasets under different linkage methods and distance metrics.

Model	**Silhouette Score**			**Davies-Bouldin Score**		
	Min	**Mean**	**Max**	**Min**	**Mean**	**Max**
Random	-0.24	-0.23	-0.22	1.51	1.59	1.64
KNN-KMeans	0.31	0.33	0.40	1.05	1.17	1.32
HyperABSA	**0.36**	**0.42**	**0.62**	**0.56**	**0.99**	**1.10**

Table 7: Comparison of cluster quality across different hypergraph construction methods.

in Table 2, across datasets, these fixed strategies yield lower or inconsistent performance, indicating their inability to capture the optimal granularity across samples. In contrast, our model dynamically selects both the threshold and the graph construction strategy per instance, effectively adapting to sample-specific views. These findings support our broader claim, that automatically identifying an appropriate granularity per instance can offer a strong alternative to using multiple graphs for capturing the different granularities.

5.7 Generalization Gap

To rigorously evaluate the generalization capabilities of HyperABSA, we investigate the *generalization gap*, defined as the difference between training and test accuracy and loss, across varying proportions of training data. Unlike prior works that rely on multiple graph constructions capturing distinct semantic views at the cost of increased complexity, our model leverages a single dynamic hypergraph, hypothesized to better balance representation richness and overfitting risk. We conduct controlled experiments on two widely-used benchmarks, Lap14 and Rest14, comparing HyperABSA against two strong baselines: YORO and RGAT. Each experiment is repeated over multiple random seeds, and average metrics are reported to ensure statistical reliability.

Figure 4 illustrates that HyperABSA consistently demonstrates significantly smaller generalization gaps across almost all training data fractions. Notably, HyperABSA performs reliably even when trained on less than 50% of the data, whereas both

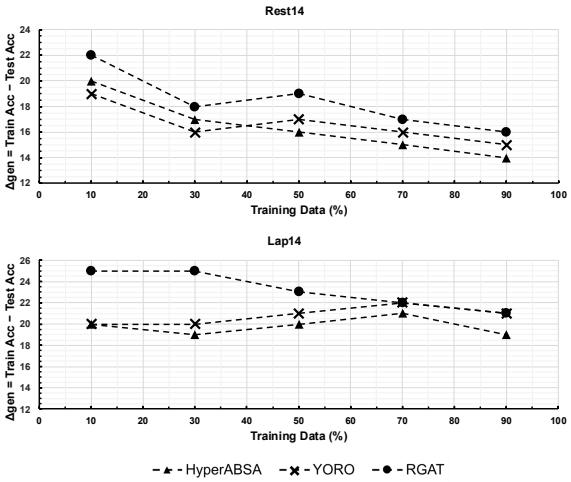

Figure 4: Evaluation of HyperABSA against multi-graph-based models on the Rest14 and MAMS datasets in terms of generalization gap.

baselines require 50–70% of the data to achieve comparable generalization. These trends suggest that the dynamic structure of HyperABSA enables it to capture fine-grained, context-sensitive token relationships without overfitting, even in low-resource settings.

6 Conclusion

In this paper, we introduce HyperABSA, a novel hypergraph construction methodology for ABSA that dynamically forms hyperedges via adaptive hierarchical clustering. Our approach addresses the challenge of overfitting in short-text scenarios by leveraging an efficient, acceleration-based thresholding mechanism, ensuring that hyperedges capture meaningful multi-node interactions while preventing excessive fragmentation or over-merging. Comprehensive evaluations on Lap14, Rest14, and MAMS datasets demonstrate that HyperABSA achieves state-of-the-art performance among graph-based approaches, highlighting its effectiveness in capturing nuanced multi-node interactions for fine-grained sentiment reasoning.

7 Limitations

Multi-graph models offer interpretable edge semantics grounded in syntactic or semantic roles, while hypergraphs, though rich in context, lack this clarity, posing challenges for interpretability and fine-grained error analysis. Our approach is computationally complex compared to conventional single-graph baselines, making it susceptible to overfitting, particularly on low-resource datasets such as Lap14, where aspect-opinion annotations are sparse and domain-specific vocabularies limit generalization. Although we introduced minor architectural adjustments to the base HGNN framework, it was not designed for ABSA. This mismatch added to the modeling complexity and may have hindered performance in ABSA-specific scenarios.

References

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, Austin, Texas. Association for Computational Linguistics.

Kamran Aziz, Donghong Ji, Prasun Chakrabarti, Tulika Chakrabarti, Muhammad Shahid Iqbal, and Rashid Abbasi. 2024. Unifying aspect-based sentiment analysis bert and multi-layered graph convolutional networks for comprehensive sentiment dissection. *Scientific Reports*, 14(1):14646.

Xuefeng Bai, Pengbo Liu, and Yue Zhang. 2020. Investigating typed syntactic dependencies for targeted sentiment classification using graph attention neural network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:503–514.

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

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 interactive presentation sessions*, pages 69–72.

Yuchou Chang, Dah-Jye Lee, James Archibald, and Yi Hong. 2008. Unsupervised clustering using hyperclique pattern constraints. In *2008 19th International Conference on Pattern Recognition*, pages 1–4. IEEE.

Chenhua Chen, Zhiyang Teng, Zhongqing Wang, and Yue Zhang. 2022. Discrete opinion tree induction for aspect-based sentiment analysis. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2051–2064.

Junjie Chen, Hongxu Hou, Yatu Ji, Jing Gao, and Tiangang Bai. 2019. Graph-based attention networks for aspect level sentiment analysis. In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 1188–1194. IEEE.

Xiaodong Cui, Wenbiao Tao, and Xiaohui Cui. 2023. Affective-knowledge-enhanced graph convolutional networks for aspect-based sentiment analysis with multi-head attention. *Applied Sciences*, 13(7):4458.

Qionghai Dai and Yue Gao. 2023. *Hypergraph Computation*. Springer Nature.

David L Davies and Donald W Bouldin. 1979. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227.

Kaize Ding, Jianling Wang, Jundong Li, Dingcheng Li, and Huan Liu. 2020. Be more with less: Hypergraph attention networks for inductive text classification. *arXiv preprint arXiv:2011.00387*.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231.

Chuan Fang. 2022. Dependencymerge guided latent tree structure for aspect-based sentiment analysis. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. 2019. Hypergraph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3558–3565.

Xiang Gao, Fan Zhou, Kedi Xu, Xiang Tian, and Yaowu Chen. 2022. A parallel algorithm for maximal cliques enumeration to improve hypergraph construction. *Journal of Computational Science*, 65:101905.

James Goodman, Andreas Vlachos, and Jason Naradowsky. 2016. Noise reduction and targeted exploration in imitation learning for Abstract Meaning Representation parsing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–11, Berlin, Germany. Association for Computational Linguistics.

Euihong Han, George Karypis, Vipin Kumar, and Bamshad Mobasher. 1997. Clustering based on association rule hypergraphs. *IEEE Transactions on Knowledge and Data Engineering*, 9(2):335–348.

Mary Harper. 2014. Learning from 26 languages: Program management and science in the babel program. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, page 1, Dublin, Ireland. Dublin

City University and 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. *Knowledge-Based Systems*, 309:112782.

Binxuan Huang and Kathleen M Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. *arXiv preprint arXiv:1909.02606*.

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.

Diederik P Kingma. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

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

Bin Liang, Rongdi Yin, Lin Gui, Jiachen Du, and Ruifeng Xu. 2020. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis. In *Proceedings of the 28th international conference on computational linguistics*, pages 150–161.

Juhua Liu, Qihuang Zhong, Liang Ding, Hua Jin, Bo Du, and Dacheng Tao. 2023. Unified instance and knowledge alignment pretraining for aspect-based sentiment analysis. *IEEE/ACM transactions on audio, speech, and language processing*, 31:2629–2642.

Dong Quan Ngoc Nguyen, Lin Xing, and Lizhen Lin. 2020. Community detection, pattern recognition, and hypergraph-based learning: approaches using metric geometry and persistent homology. In *Fuzzy Systems and Data Mining VI*, pages 457–473. IOS Press.

Hao Niu, Yun Xiong, Jian Gao, Zhongchen Miao, Xiaosu Wang, Hongrun Ren, Yao Zhang, and Yangyong Zhu. 2022. Composition-based heterogeneous graph multi-channel attention network for multi-aspect multi-sentiment classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6827–6836.

Yunhui Pan, Dongyao Li, Zhouhao Dai, and Peng Cui. 2023. Aspect-based sentiment analysis using dual probability graph convolutional networks (dp-gcn) integrating multi-scale information. In *International Conference on Neural Information Processing*, pages 495–512. Springer.

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

Soujanya Poria, Nir Ofek, Alexander Gelbukh, Amir Hussain, and Lior Rokach. 2014. Dependency tree-based rules for concept-level aspect-based sentiment analysis. In *Semantic Web Evaluation Challenge: SemWebEval 2014 at ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers*, pages 41–47. Springer.

Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Shota Saito. 2022. Hypergraph modeling via spectral embedding connection: Hypergraph cut, weighted kernel k-means, and heat kernel. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8141–8149.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 6578–6588.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021. Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 2910–2922.

Kathryn Turnbull, Simón Lunagómez, Christopher Nemeth, and Edoardo Airoldi. 2024. Latent space modeling of hypergraph data. *Journal of the American Statistical Association*, 119(548):2634–2646.

Haoyu Wang, Xihe Qiu, and Xiaoyu Tan. 2024. Multivariate graph neural networks on enhancing syntactic and semantic for aspect-based sentiment analysis. *Applied Intelligence*, 54(22):11672–11689.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. *arXiv preprint arXiv:2004.12362*.

Guangtao Xu, Peiyu Liu, Zhenfang Zhu, Jie Liu, and Fuyong Xu. 2021. Attention-enhanced graph convolutional networks for aspect-based sentiment classification with multi-head attention. *Applied Sciences*, 11(8):3640.

Lvxiaowei Xu, Xiaoxuan Pang, Jianwang Wu, Ming Cai, and Jiawei Peng. 2023. Learn from structural scope: Improving aspect-level sentiment analysis with hybrid graph convolutional networks. *Neurocomputing*, 518:373–383.

Bengong Yu and Shuwen Zhang. 2023. A novel weight-oriented graph convolutional network for aspect-based sentiment analysis. *The Journal of Supercomputing*, 79(1):947–972.

Jun Yu, Dacheng Tao, and Meng Wang. 2012. Adaptive hypergraph learning and its application in image classification. *IEEE Transactions on Image Processing*, 21(7):3262–3272.

Li Yuan, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2020. Graph attention network with memory fusion for aspect-level sentiment analysis. In *Proceedings of the 1st conference of the asia-pacific chapter of the association for computational linguistics and the 10th international joint conference on natural language processing*, pages 27–36.

Jiandian Zeng, Tianyi Liu, Weijia Jia, and Jiantao Zhou. 2022. Relation construction for aspect-level sentiment classification. *Information Sciences*, 586:209–223.

Yufei Zeng, Zhixin Li, Zhenbin Chen, and Huifang Ma. 2023. Aspect-level sentiment analysis based on semantic heterogeneous graph convolutional network. *Frontiers of Computer Science*, 17(6):176340.

Fan Zhang, Wenbin Zheng, and Yujie Yang. 2024. Graph convolutional network with syntactic dependency for aspect-based sentiment analysis. *International Journal of Computational Intelligence Systems*, 17(1):37.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2022a. Hegel: Hypergraph transformer for long document summarization. *arXiv preprint arXiv:2210.04126*.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022b. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Transactions on Knowledge and Data Engineering*, 35(11):11019–11038.

Zheng Zhang, Zili Zhou, and Yanna Wang. 2022c. Ssegcn: Syntactic and semantic enhanced graph convolutional network for aspect-based sentiment analysis. In *Proceedings of the 2022 conference of the North American Chapter of the association for computational linguistics: human language technologies*, pages 4916–4925.

Guoshuai Zhao, Yiling Luo, Qiang Chen, and Xueming Qian. 2023. Aspect-based sentiment analysis via multitask learning for online reviews. *Knowledge-Based Systems*, 264:110326.

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.

Xinke Zhi. 2024. A review of hypergraph neural networks. *EAI Endorsed Transactions on e-Learning*, 10.

A Example Appendix

This is a section in the appendix.