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

Omkar Mahesh Kashyap[1*] , **Padegal Amit**[1*], **Madhav Kashyap**[2]
Ashwini M Joshi[1], **Shylaja SS**[1]
[1]PES University [2]University of Washington
{omkar.m.kashyap, padegalamit, madhavkashyap99}@gmail.com,
{ashwinimjoshi, shylaja.sharath}@pes.edu

Abstract

Aspect-Term Sentiment Analysis (ATSA) aims to predict sentiment polarity for specific aspect terms, a task complicated by conflicting sentiments and limited context in short texts. Existing graph-based approaches rely on predefined pairwise structures to capture different linguistic views. However, this leads to two key limitations: (1) their pairwise formulation often requires multiple graphs to improve expressive capacity, and (2) their reliance on predefined parsers or heuristic graph construction limits adaptability to sentence-specific sentiment composition. We propose HyperATSA, a dynamic hypergraph framework that overcomes these limitations through a single instance-specific hypergraph constructed directly from contextual token representations. Hyperedges are dynamically induced via Hierarchical Agglomerative Clustering (HAC) over token embeddings, where an acceleration-based cutoff identifies sentence-specific semantic groupings and enables adaptive hypergraph construction. Experiments on Lap14, Rest14, and MAMS demonstrate consistent improvements over strong graph-based baselines, suggesting that hypergraph-based relational modeling generalizes effectively to short-text sentiment composition.

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 a subtask of ABSA, 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.

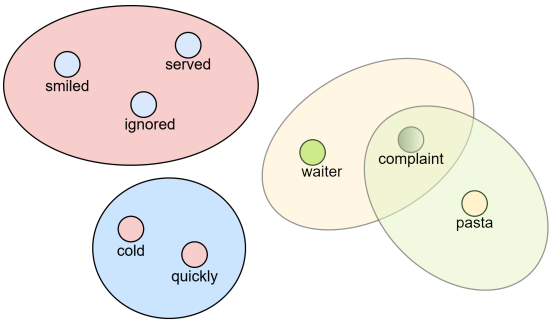

Figure 1: A hypergraph of word interactions showing several semantic clusters based on aspect and sentiment polarity. This illustrates how words are grouped according to meaning and sentiment.

Graph neural networks (GNNs) (Kipf and Welling, 2016) have been widely adopted to capture relevant semantic interactions over dependency graph structures for the ATSA task (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 reliance on pairwise relationships, where interactions are modeled only between two connected tokens at a time. As sentiment polarity in ATSA often emerges from the joint influence of multiple aspect, opinion, and contextual terms, pairwise message passing can struggle to capture compositional sentiment dependencies.

This limitation has motivated multiple-graph architectures (Li et al., 2021; Aziz et al., 2024; Zheng and Li, 2024), which integrate multiple graph structures to represent different syntactic and semantic relational views. However, these approaches also rely on parser- or heuristic-based graph constructions, making them susceptible to noisy or inaccurate relational structures. Combining multiple

[*] Equal Contribution

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop), pages 595–608

graphs also introduces redundancy across views and requires additional encoders and fusion components. This added complexity is particularly ill-suited to short-text ATSA, where limited context favors compact models that generalize without relying on multiple predefined relational structures.

We address these challenges with HyperATSA, a dynamic hypergraph framework that constructs a single hypergraph per input. Unlike pairwise graphs, hyperedges naturally connect arbitrary groups of tokens simultaneously (Zhang et al., 2022a), directly capturing the joint influence of aspect, opinion, and contextual terms that drives sentiment polarity in ATSA (see Figure 1). Rather than relying on external parsers, we construct the hypergraph directly from contextual token representations.

Short texts offer little redundancy, making it critical that relational structure reflects the specific semantic composition of each input. To this end, we construct a data-driven hierarchy over token representations and cut it where semantic groupings become naturally distinct, letting coherent token groups emerge without any predefined structure. This yields instance-specific hypergraphs that generalize well in short-text regimes while being significantly faster to construct. HyperATSA consistently outperforms strong graph-based baselines on Lap14, Rest14, and MAMS, validating that higher-order relational modeling is both a more expressive and more efficient alternative to existing graph-based methods for ATSA.

- We introduce the first hypergraph framework for ATSA, constructing instance-specific aspect-opinion groupings directly from contextual representations without predefined parsers or heuristic graph structures.
- We propose an acceleration-based cutoff with a fallback strategy for adaptive hypergraph construction, enabling robust structure discovery across inputs of varying length and complexity.
- Experiments on Lap14, Rest14, and MAMS demonstrate consistent improvements over strong graph-based baselines, with HyperATSA showing stronger generalization across different data regimes and substantially faster hypergraph construction compared to other clustering alternatives.

2 Related Work

2.1 Graph-Based Methods

Graph-based models are widely used in ATSA to encode syntactic and semantic relationships between tokens. Early approaches employed graph convolutional networks (GCNs) over dependency trees to model syntactic structure (Chen et al., 2019), while later work integrated semantic relations, relational edge types, and aspect-aware interactions to enrich graph representations (Wang et al., 2020; Chen et al., 2022; Bao et al., 2023; Huang and Carley, 2019; Yuan et al., 2020; Tian et al., 2021). Subsequent methods combined attention mechanisms with graph propagation to jointly model contextual and structural dependencies (Xu et al., 2021; Pan et al., 2023; Cui et al., 2023; Zhang et al., 2024), and more recent work explored heterogeneous and multi-graph architectures to capture complementary relational views within text (Zeng et al., 2023; Niu et al., 2022; Li et al., 2021; Aziz et al., 2024; Zheng and Li, 2024).

Despite their differences, these approaches largely rely on predefined pairwise graph structures, often constructed from external parsers or manually designed relational priors. Such formulations can struggle to represent higher-order and overlapping aspect–opinion interactions, particularly in short-text ATSA settings where relational cues are sparse and context-dependent.

2.2 Hypergraph Construction Methods

While hypergraph models have demonstrated strong potential across domains, their use in ATSA remains underexplored. 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 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).

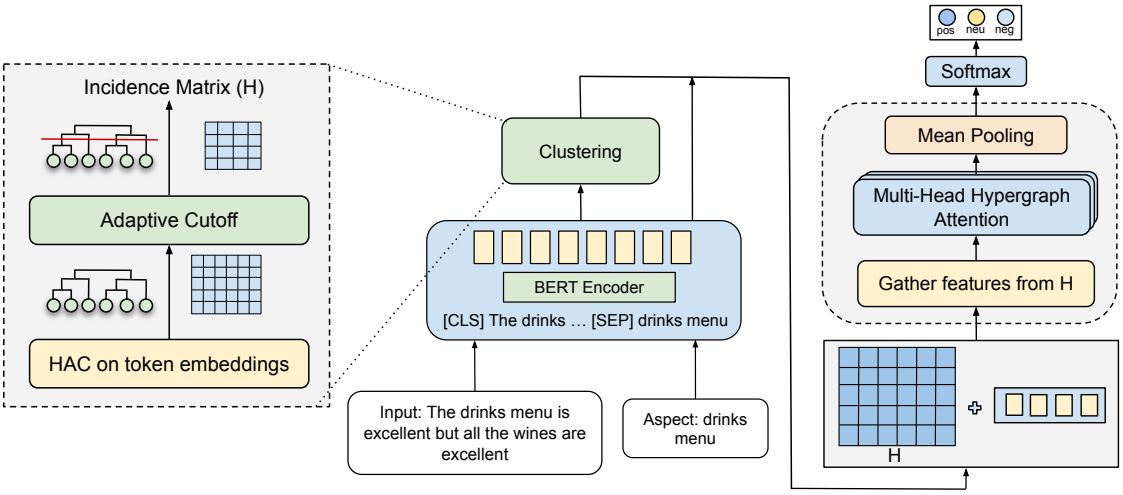

Figure 2: Architecture of HyperATSA.

3 Methodology

In ATSA tasks, a n-word input sentence is represented as $T = \{v_1, v_2, \ldots, v_n\}$, where v_i denotes the i-th word in the sequence. The task involves r aspect terms, each consisting of q words, represented as

$$B = \{b_1^1, b_2^1, \ldots, b_q^1, b_1^2, \ldots, b_q^r\} \subseteq T,$$

where the r-th aspect is denoted as a subset of q tokens $b^{(r)} = \{b_1^r, \ldots, b_q^r\} \subseteq T$

The objective is to learn a function, $g_r : (T, b^{(r)}) \mapsto z$, that maps the sentence T and $b^{(r)}$ to a sentiment label z, indicating the polarity toward that aspect.

3.1 Input Representation

Given a sentence of n tokens, we obtain contextual embeddings using a pretrained encoder:

$$\mathbf{X}^{(0)} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{n \times d}$$

where $\mathbf{x}_i \in \mathbb{R}^d$ is the contextual embedding of token i. These serve as initial node features for subsequent hypergraph construction.

3.2 Hierarchical Clustering of Token Embeddings

To induce semantic groupings, we apply HAC to the set $\{\mathbf{x}_i\}_{i=1}^n$, using a linkage function $\ell(\cdot, \cdot)$. At each merge step $t \in \{1, \ldots, n-1\}$, the two most similar clusters c_1, c_2 are merged, producing a new cluster of size s_t with dissimilarity δ_t.

We represent the sequence of merges as

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

$$z_t = \left(C_1^{(t)}, C_2^{(t)}, \delta_t, s_t\right). \quad (2)$$

Here, each merge step satisfies $C_1^{(t)}, C_2^{(t)} \subseteq T$, and the size of the merged cluster is given by $s_t = \left|C_1^{(t)} \cup C_2^{(t)}\right|$. The sequence $\{\delta_t\}$ encodes the evolution of inter-cluster dissimilarities.

3.3 Adaptive Cutoff: Acceleration-Fallback Criterion

A fixed global merge threshold may fail to account for variations in sentence length and syntactic complexity, potentially resulting in suboptimal clustering. To address this, we define an adaptive cutoff that selects an appropriate dendrogram height for each instance.

Let $\rho \in (0, 1]$ control the fraction of merges considered. The number of merges in this window is:

$$r = \max\{1, \lfloor \rho \cdot (n-1) \rfloor\}. \quad (3)$$

Let $\delta_{\text{recent}} = [\delta_{n-r}, \ldots, \delta_{n-1}]$ denote the sequence of recent merge dissimilarities of length r. For $j = 1, \ldots, r-2$, we define the second-order finite differences as:

$$\kappa_j = \delta_{n-r+j+1} - 2\delta_{n-r+j} + \delta_{n-r+j-1}. \quad (4)$$

The elbow point is then selected as:

$$j^* = \arg\max_j \ \kappa_j. \quad (5)$$

However, if the recent window is too short ($r \leq 3$), or the second-order signal is weak ($\max_j |\kappa_j| \leq$

ϵ), where ε is a small constant, situations common in sparse or noisy data where hierarchical structure is less distinct, we adopt a fallback cutoff:

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

where $\overline{\delta}_{\text{recent}}$ and σ_{recent} denote the mean and standard deviation of the recent dissimilarities, respectively, and λ is a positive scaling parameter that controls how much the variability of recent dissimilarities influences the cutoff threshold.

The final dendrogram cutoff is then:

$$\delta_{\text{elbow}} = \begin{cases} \min\{\delta_{j^*}, \delta_{\text{fallback}}\}, & \text{if } r > 3 \\ & \text{and } \max_j |\kappa_j| > \epsilon \\ \delta_{\text{fallback}}, & \text{otherwise} \end{cases} \tag{7}$$

By cutting the dendrogram at the threshold δ_{elbow}, we obtain a partition of the data into $|E|$ clusters.

We represent these clusters as hyperedges of a hypergraph $\mathcal{G} = (V, E)$, where $V = \{1, \ldots, n\}$ corresponds to tokens, and $E = \{C_e\}_{e=1}^{|E|}$ denotes clusters obtained from the adaptive cutoff. The incidence matrix $H \in \{0,1\}^{|V| \times |E|}$ is defined as:

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

The algorithm is presented in Appendix A

3.4 Hypergraph Neural Network

Information is propagated over the constructed hypergraph using a Hypergraph Attention Network (HyperGAT) (Ding et al., 2020).

Given initial node features $X^{(0)} \in \mathbb{R}^{n \times d}$ and incidence matrix $H \in \{0,1\}^{n \times |E|}$, we compute attention-based aggregation over nodes within each hyperedge.

For each attention head $h \in \{1, \ldots, H\}$, we first project node features into a head-specific subspace:

$$Z^{(h)} = \sigma\left(X^{(0)} W_1^{(h)}\right) \in \mathbb{R}^{n \times d_h}, \tag{8}$$

where $W_1^{(h)} \in \mathbb{R}^{d \times d_h}$ is a learnable projection matrix and $\sigma(\cdot)$ is a non-linear activation function.

We then compute attention coefficients over nodes within each hyperedge. For hyperedge $e \in E$ and node i such that $H_{i,e} = 1$:

$$A_{e,i}^{(h)} = \frac{\exp\left(Z_{i,:}^{(h)} \cdot a^{(h)}\right)}{\sum_{j: H_{j,e}=1} \exp\left(Z_{j,:}^{(h)} \cdot a^{(h)}\right)}, \tag{9}$$

where $a^{(h)} \in \mathbb{R}^{d_h}$ is a learnable attention vector. The softmax is computed over nodes incident to hyperedge e, ensuring normalized contributions within each hyperedge.

Each hyperedge aggregates features from its incident nodes using the learned attention weights:

$$E_{e,:}^{(h)} = \sum_{i=1}^{n} H_{i,e} \cdot A_{e,i}^{(h)} \cdot Z_{i,:}^{(h)}. \tag{10}$$

This yields head-specific hyperedge representations $E^{(h)} \in \mathbb{R}^{|E| \times d_h}$. The outputs from all attention heads are concatenated to obtain the final hyperedge representation:

$$E = \text{Concat}\left(E^{(1)}, \ldots, E^{(H)}\right) \in \mathbb{R}^{|E| \times d'}, \tag{11}$$

where $d' = H \cdot d_h$.

To obtain a graph-level representation, we apply mean pooling over the hyperedge representations and project the resulting embedding into the output space. The output logits are then normalized using the softmax function to produce a probability distribution over the sentiment classes.

3.5 Geometric Interpretation of Acceleration

To better understand the role of acceleration in detecting the elbow point, we treat the recent dissimilarities δ_{recent} as a discrete signal representing merge distances. The first-order differences capture the local slope, while the second-order differences, $\Delta^2 \delta_{\text{recent}}$, quantify the curvature, i.e., how sharply the merge distances deviate from a linear trend. In the context of ATSA, sharp increases in merge distance often indicate transitions from semantically coherent aspect–opinion groupings to broader clusters that combine unrelated contextual information. Peaks in curvature therefore provide a natural signal for identifying the point beyond which cluster merges may dilute aspect-specific sentiment structure. As defined in Eq. 5, we select the index corresponding to the maximum acceleration as the cutoff point.

4 Experimental Details

4.1 Implementation Details

We evaluate HyperATSA on three standard ABSA benchmarks: MAMS (Jiang et al., 2019), SemEval-2014 Restaurants (Rest14), and Laptops (Lap14) (Pontiki et al., 2016), following the data splits of Bai et al. (2020) and the dataset pre-processing

Model	Lap14		Rest14		MAMS	
	Acc(%)	F1(%)	Acc(%)	F1(%)	Acc(%)	F1(%)
R-GAT‡ (Wang et al., 2020)	78.21	74.07	84.64	77.14	83.16	82.42
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	-	-
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	-	-
HyperATSA	**80.46**	**77.42**	**86.76**	**80.64**	**84.56**	83.74

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

Method Variant	ρ	Rest14		Lap14		MAMS	
		Acc. (%)	F1 (%)	Acc. (%)	F1 (%)	Acc. (%)	F1 (%)
$\delta_{\text{elbow}} = \delta_{\text{fallback}}$ (Variance Only)	-	84.07	76.89	79.06	75.84	84.00	83.51
$\delta_{\text{elbow}} = \delta_{n-m+k}$ (Elbow Only)	0.2	80.59	74.61	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
$\delta_{\text{elbow}} = \min(\delta_{n-m+k}, \delta_{\text{fallback}})$ (Without Dynamic Threshold)	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
HyperATSA (Full Model)	Dynamic	**86.76**	**80.64**	**80.46**	**77.42**	**84.56**	**83.74**

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

pipeline of RGAT (Wang et al., 2020). We provide the dataset splits in Table 11. Our model leverages BERT-base as the underlying text encoder, providing contextualized token representations. HAC is performed using Ward's method with Euclidean distance on ℓ_2-normalized token embeddings. 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 λ are treated as trainable parameters. Training is performed using the Adam optimizer (Kingma, 2014) with a learning rate of 10^{-4}, a batch size of 16, and is conducted on a single NVIDIA RTX 4090 GPU.

We have open-sourced our codebase [1].

4.2 Baselines

We benchmark HyperATSA against a comprehensive set of graph-based architectures leveraging syntactic dependencies and GCN variants, including R-GAT (Wang et al., 2020), RGAT (Bai et al., 2020), DMGLT (Fang, 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

[1] https://github.com/OmkarKashyap/HyperATSA

such as YORO (Zheng and Li, 2024), MLEGCN (Aziz et al., 2024) and CaBiLSTM-BERT (He et al., 2025). To ensure fair comparison and reproducibility, we restrict evaluation to publicly available implementations.

5 Results and Analysis

5.1 Results

As shown in Table 1, HyperATSA consistently outperforms strong baselines across all three benchmark datasets. On the Lap14 dataset, it achieves a 5% improvement in F1 score over CaBiLSTM-BERT, a recent 2025 baseline, and outperforms models such as YORO with an average gain of 3% in accuracy. For the Rest14 dataset, HyperATSA obtains the highest accuracy and F1 score, surpassing YORO, MTABSA-BERT and R-GAT by an average margin of 3%. On the larger MAMS dataset, HyperATSA outperforms DMGLT and CHGMAN by an average of 2% in both accuracy and F1, while remaining competitive with YORO in terms of F1 score.

Further inspection of the generated hypergraphs reveals that HyperATSA naturally discovers several meaningful types of hyperedges from the semantic embedding space. In particular, as in Table 5, we observe: (1) *punctuation hyperedges*, which group punctuation tokens together and implicitly separate structural delimiters from semantic content; (2) *n-gram hyperedges*, which capture short locally coherent phrases and compositional token patterns; and (3) *aspect-specific hyperedges*, which connect aspect terms with relevant opinion and contextual words at both coarse- and fine-grained levels. Interestingly, these structures emerge automatically without manually defined syntactic rules or heuristic graph templates, suggesting that the adaptive clustering process organizes tokens according to latent semantic and contextual similarity. We provide additional qualitative examples and analysis of these hyperedge patterns in Section B

5.2 Comparison with LLM Baselines

We begin by benchmarking HyperATSA against large language models to assess whether explicit structural modeling provides benefits beyond prompting-based inference. Table 3 reports a comparison between HyperATSA and several large language model (LLM) baselines on the Lap14 and Rest14 datasets, where LLM performances are taken from (Chen et al., 2024; Wang et al., 2024), following their experimental setup and prompt design. For consistency, we evaluate HyperATSA on the same task formulation, dataset splits, and input preprocessing.

On average across both datasets, HyperATSA consistently surpasses all LLM baselines. Compared to the strongest proprietary model - ChatGPT (few-shot) - HyperATSA achieves a mean improvement of +2.22% in Accuracy and +2.60% in F1. Relative to the best-performing open-source model, Alpaca-13B, the gains are even more substantial, with +3.63% higher Accuracy and +4.89% higher F1 on average.

These results suggest a fundamental limitation of autoregressive LMs in aspect-based sentiment analysis. While they capture general semantics, they struggle with fine-grained, aspect-conditioned reasoning, particularly in sentences with multiple interacting opinion targets. In contrast, HyperATSA explicitly models token-level relational structure via hyperedges, enabling more precise attribution of sentiment to aspects.

Model	Lap14		Rest14	
	Acc(%)	F1(%)	Acc(%)	F1(%)
ChatGPT (zero-shot)‡	77.64	72.30	82.39	73.64
ChatGPT (few-shot)‡	78.15	75.79	84.62	76.08
LLaMa-7B*	75.37	71.63	80.04	70.97
LLaMa-13B*	75.82	70.89	81.78	72.89
Alpaca-7B*	74.58	71.18	80.38	71.36
Alpaca-13B*	77.03	72.48	82.93	75.81
HyperATSA	**80.46**	**77.42**	**86.76**	**80.64**

Table 3: Accuracy and F1 scores (%) of HyperATSA compared with LLM variants on Lap14 and Rest14 datasets. The results with ‡, * are retrieved from (Chen et al., 2024) and (Wang et al., 2024) respectively.

5.3 Encoder Analysis

Most prior ATSA approaches adopt BERT-based encoders, making BERT a natural backbone for evaluating HyperATSA under commonly used experimental settings. We additionally explore RoBERTa to examine how the proposed hypergraph formulation behaves under stronger contextual representations. This results in four configurations: BERT, BERT+HAC (Ours), RoBERTa, and RoBERTa+HAC (Ours). For the encoder-only baselines (BERT and RoBERTa), the contextualized representations are directly passed through a feed-forward network and classification layer without additional structural modeling.

Across both backbones, incorporating the hypergraph consistently improves performance. Under a BERT backbone, the hypergraph yields an improvement of approximately 2–3% in both accuracy and F1 score, as shown in Table 4. Similar gains are observed with RoBERTa, where the hypergraph-enhanced variant consistently outperforms the corresponding encoder-only baseline.

We further observe that replacing BERT with RoBERTa leads to stronger overall performance, with RoBERTa+HAC (Ours) achieving the best results across datasets. Interestingly, BERT+HAC (Ours) also surpasses encoder-only RoBERTa in several settings, suggesting that the proposed hypergraph formulation contributes meaningful relational modeling beyond improvements obtained solely from stronger pretrained representations.

Overall, these results demonstrate that the proposed hypergraph mechanism remains effective across different encoder backbones while benefiting from improvements in the underlying contextual representations.

Model	Lap14		Rest14	
	Acc(%)	F1(%)	Acc(%)	F1(%)
BERT	76.24	72.53	83.89	77.90
BERT + HAC (Ours)	**80.46**	**77.42**	**86.76**	**80.64**
RoBERTa	77.81	75.87	84.29	80.32
RoBERTa + HAC (Ours)	**83.44**	**80.60**	**87.76**	**82.76**

Table 4: Backbone-controlled comparison across datasets.

5.4 Generalization Gap

Prior work in ATSA often relies on multi-graph architectures to capture different semantic views and relational structures. However, the increased structural complexity and parameterization of such models can lead to overfitting, particularly in short-text settings with limited training data. To evaluate the generalization ability of HyperATSA, we measure the generalization gap (Δ_{gen}), defined as the difference between training and test loss, across varying amounts of training data on the Rest14 and Lap14 datasets. We compare HyperATSA against YORO, a multi-graph model, and RGAT, a graph-based baseline.

As shown in Figure 3, HyperATSA consistently achieves lower or comparable generalization gaps across most training settings. In particular, the model maintains stable generalization even under

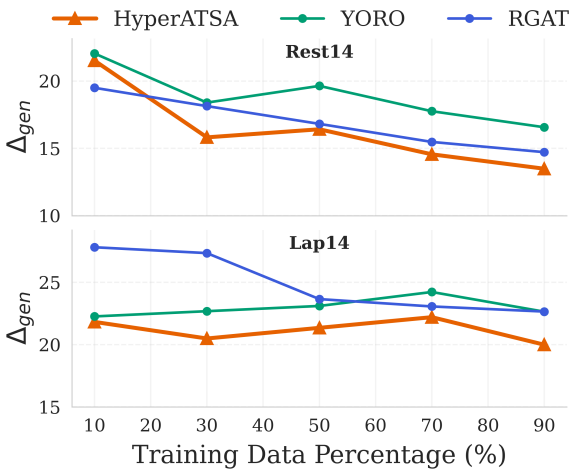

Figure 3: Evaluation of HyperATSA against multi-graph-based models on the Rest14 and MAMS datasets in terms of Generalization Gap (Δ_{gen}), defined as the difference between train loss and test loss. Lower values indicate better generalization.

reduced training data, whereas competing methods exhibit larger gaps and higher sensitivity to training size. On Rest14, HyperATSA achieves the lowest generalization gap across nearly all training percentages. Similar trends are observed on Lap14, where HyperATSA remains competitive and stable as training data increases. We provide further analysis in Section C

These results suggest that the adaptive hypergraph structure enables more effective modeling of context-specific relationships while reducing overfitting, especially in limited-data regimes. Overall, the findings indicate that HyperATSA provides a more robust and generalizable alternative to existing graph-based approaches for ATSA.

5.5 Clustering Quality Analysis

To further disentangle whether the observed gains arise from hypergraph modeling or merely from the underlying contextual embeddings, we evaluate alternative hypergraph construction strategies. Specifically, we compare our method against (i) a Random hypergraph, where nodes and hyperedges are formed without structural guidance, and (ii) a KNN-KMeans hybrid, a widely adopted hypergraph construction approach that combines local neighborhood structure with global centroid-based clustering.

We evaluate cluster quality using the Silhouette Score (Rousseeuw, 1987) and Davies–Bouldin Score (Davies and Bouldin, 1979), which measure intra-cluster cohesion and inter-cluster sep-

No.	Sentence	Aspect	Hyperedges from HyperATSA
1	To be completely fair, the only redeeming factor was the food, which was above average, but couldn't make up for all the other deficiencies of Teodora	food	['.', '.', '.', '.', "'", '.'], ['to', 'be', 'completely', 'fair'], ['the', 'only', 'red', '##eem', '##ing', 'factor', 'the'], ['make', 'up'], ['for', 'all', 'the', 'other'], ['was', 'food', 'which', 'was', 'above', 'average', 'but', 'couldn', 't', 'def', '##iciencies', 'of', '##ora', 'food'], ['the', 'only', 'red', '##eem', '##ing', 'factor', 'the'],
2	The food is uniformly exceptional, with a very capable kitchen which will proudly whip up whatever you feel like eating, whether it's on the menu or not	kitchen	['.', '.', '.', '.'], ['you', 'feel', 'like'], ['the', 'is', 'uniformly', 'with', 'a', 'very', 'which', 'will'], ['food', 'exceptional', 'capable', 'kitchen', 'proudly', 'whip', 'up', 'eating', 'menu'], ['whatever', 'it', "'", 's', 'on', 'the'], ['whether', 'or', 'not'],

Table 5: Example sentences with aspect-specific hyperedges extracted by HyperATSA. Each bracketed token set ([.]) represents a hyperedge. Special tokens like 'CLS' and 'SEP' are excluded for illustration.

aration. As shown in Table 6, HyperATSA consistently produces more cohesive and better-separated clusters than both baselines. Random hypergraphs fail to capture meaningful structure, while the KNN–KMeans hybrid yields moderate improvements but remains less effective than our hierarchical construction.

Since hyperedges are derived from these clusters, improvements in cluster quality lead to more coherent token groupings and cleaner separation between aspects. In contrast, weaker clustering introduces noisy groupings that mix unrelated signals, degrading aspect-level sentiment modeling. This suggests that the performance gains arise from the induced structure, rather than the embeddings alone (Tables 6 and 7).

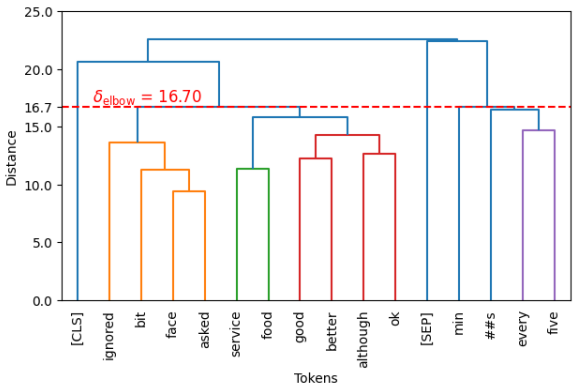

Figure 4: Dendrogram visualization of token-level hierarchical clustering.

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
KMeans	0.31	0.33	0.40	1.05	1.17	1.32
Hierarchical (Ours)	**0.36**	**0.42**	**0.62**	**0.56**	**0.99**	**1.10**

Table 6: Comparison of cluster quality across different hypergraph construction methods.

Model	Lap14		Rest14	
	Acc(%)	F1(%)	Acc(%)	F1(%)
DBSCAN	47.30	27.86	52.87	47.92
GMM	81.40	78.29	86.12	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
Hierarchical (Ours)	**83.44**	80.60	**87.77**	**82.76**

Table 7: Comparison of clustering methods evaluated by accuracy and F1 score on the Lap14 and Rest14 datasets.

5.6 Multi-Granular Approach of Hypergraph

We also examine whether HyperATSA's gains stem from its ability to adaptively capture multiple levels of granularity. 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 ($\rho = 0.2, 0.5, 0.8$). As seen 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 Efficiency Analysis

The primary computational cost in HyperATSA arises from the hierarchical agglomerative clustering (HAC) step used to construct the hypergraph structure. For a sentence with n tokens, storing the pairwise token similarity matrix requires $O(n^2)$ memory complexity. Depending on the linkage strategy and implementation, the time complexity typically ranges between $O(n^2)$ and $O(n^3)$ in the worst case. However, since ABSA sentences are generally short ($n \leq 50$), the practical computational overhead remains small.

The adaptive cutoff mechanism performs a linear scan over the final merge distances in the dendrogram to identify structural transitions. This operation only requires $O(n)$ time and is negligible compared to the clustering step.

In contrast, elbow-based K-Means methods require repeatedly running K-Means for multiple candidate cluster counts $K \in \{1, \ldots, K_{\max}\}$. A single K-Means run has complexity approximately $O(nkdI)$, where k is the number of clusters, d is the embedding dimension, and I is the number of optimization iterations. Consequently, the total cost scales with the number of evaluated cluster settings, making repeated per-instance clustering computationally less efficient than a single HAC-based adaptive construction procedure.

Method	Lap14 (min:secs)	Rest14 (min:secs)
No HG	9:37	14:39
Random	10:30	20:20
GMM	27:43	44:15
KMeans	55:00	144:20
HyperATSA	11:31	17:36

Table 8: Comparison of times in minutes:seconds to run for different methods on the Lap14 and Rest14 datasets. All experiments were conducted on a single NVIDIA RTX 4090 GPU

Empirically (see Table 8), our adaptive hypergraph construction remains highly efficient compared to other clustering approaches. While it is slightly slower than the *No Hypergraph* ("No HG") setting, which directly applies a linear classification layer over contextualized BERT embeddings without any graph construction, and the *Random* hypergraph setting, where hyperedges are generated from Gaussian random features sampled from $\mathcal{N}(0, 1)$, the additional overhead introduced by the adaptive cutoff computation is minimal, increasing the total runtime by only about 1–2 minutes overall. In contrast, Gaussian Mixture Models (GMM) and K-Means take significantly longer because they repeatedly compute distances and refine clusters, resulting in much higher runtimes. They exceed ours by over 3x and 8x respectively.

This demonstrates that HyperATSA balances adaptivity and efficiency while introducing meaningful structural modeling at negligible extra cost. It remains more scalable than conventional clustering approaches.

6 Conclusion

In this paper, we introduce HyperATSA, a novel hypergraph construction methodology for ATSA 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 HyperATSA achieves state-of-the-art performance among graph-based approaches, highlighting its effectiveness in capturing nuanced multi-node interactions for fine-grained sentiment reasoning.

Limitations

Multiple-graph models offer interpretable edge semantics grounded in syntactic or semantic roles, while hypergraphs, despite capturing richer higher-order contextual relationships, provide less explicit relational interpretability, making fine-grained error analysis more challenging. Additionally, although we introduced minor architectural adaptations to the base HGNN framework, it was not originally designed for ATSA, which may limit its effectiveness in certain ATSA-specific scenarios.

References

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.

Yanjiang Chen, Kai Zhang, Feng Hu, Xianquan Wang, Ruikang Li, and Qi Liu. 2024. Dynamic multi-granularity attribution network for aspect-based sentiment analysis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10920–10931.

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.

DL Davies and DW Bouldin. 1979. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(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*.

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.

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.

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, Dimitrios Galanis, Harris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, and 1 others. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 19–30.

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.

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.

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 Algorithm of HyperATSA

Algorithm 1 describes the acceleration-based elbow strategy used for hypergraph construction.

Algorithm 1 Acceleration-Based Elbow Method for Hypergraph Construction

Input: Linkage matrix $Z \in \mathbb{R}^{(n-1)\times 4}$
Hyperparameters: $\rho \in (0, 1]$, $\lambda > 0$, $\epsilon > 0$
Output: δ_{elbow}

$r \leftarrow \max(1, \lfloor \rho(n-1) \rfloor)$
$\delta_{\text{recent}} \leftarrow [\delta_{n-r}, \ldots, \delta_{n-1}]$
if $|\delta_{\text{recent}}| > 3$ **then**
  for $j \leftarrow n - r$ **to** $n - 3$ **do**
    $\kappa_j \leftarrow \delta_{j+2} - 2\delta_{j+1} + \delta_j$
  end for
  $j^\star \leftarrow \arg\max_j \kappa_j$
  $\delta^{(1)}_{\text{elbow}} \leftarrow \delta_{j^\star}$
end if
$\bar{\delta}_{\text{recent}} \leftarrow \text{mean}(\delta_{\text{recent}})$
$\sigma_{\text{recent}} \leftarrow \text{std}(\delta_{\text{recent}})$
$\delta_{\text{fallback}} \leftarrow \bar{\delta}_{\text{recent}} + \lambda\sigma_{\text{recent}}$
if $|\delta_{\text{recent}}| > 3$ **and** $\max_j |\kappa_j| > \epsilon$ **then**
  $\delta_{\text{elbow}} \leftarrow \min(\delta^{(1)}_{\text{elbow}}, \delta_{\text{fallback}})$
else
  $\delta_{\text{elbow}} \leftarrow \delta_{\text{fallback}}$
end if
return δ_{elbow}

B Intuition of Hypergraphs for ATSA

Table 5 provides qualitative insight into the types of hyperedges automatically discovered by HyperATSA. First, we observe *punctuation hyperedges*, where punctuation symbols such as commas and periods are consistently clustered together into separate groups. This behavior suggests that the model learns to isolate structural delimiters from semantic content, preventing punctuation tokens from interfering with aspect-level semantic aggregation.

Second, the model forms *n-gram hyperedges* that capture short locally coherent token patterns and compositional phrases. For example, phrases such as *make up* and *feel like* are grouped into compact hyperedges, indicating that the adaptive clustering process preserves meaningful local contextual structure beyond simple pairwise token relations.

Finally, we observe *aspect-specific hyperedges*, which connect aspect terms with sentiment-bearing opinion and contextual words at both fine- and coarse-grained levels. In Example 1, the aspect term *food* is grouped with words such as *above*, *average*, and *deficiencies*, enabling the hypergraph to jointly capture both positive and negative sentiment cues within the same semantic structure. Similarly, in Example 2, the aspect *kitchen* is clustered with related opinion expressions such as *exceptional*, *capable*, and *proudly*. These examples suggest that HyperATSA implicitly organizes tokens according to latent semantic and contextual similarity, without relying on manually designed syntactic rules, dependency parsers, or heuristic graph templates.

C Additional details on Generalization Gap

To further analyze the generalization behavior of HyperATSA, we additionally compare the test loss trends of HyperATSA and RGAT under varying amounts of training data, as shown in Figure 5. Across both Lap14 and MAMS, HyperATSA consistently achieves lower test loss than RGAT for nearly all training percentages. The improvement is particularly noticeable in lower-data regimes, where graph-based models are generally more prone to overfitting and unstable optimization.

On Lap14, HyperATSA maintains a substantially lower test loss as the amount of training data increases, suggesting that the adaptive hypergraph structure enables more stable learning and improved transfer from training to unseen examples. A similar trend is observed on MAMS, where HyperATSA achieves lower or comparable test loss across most settings while remaining more stable as training size varies. These findings complement the generalization gap results in Section 5.4 and further indicate that HyperATSA learns more robust representations under limited supervision.

In addition to generalization performance, we also evaluate model efficiency using the accuracy-per-parameter ratio (Acc/P), reported in Table 10. Despite using approximately the same number of parameters as RGAT, HyperATSA consistently achieves better Acc/P ratios across all datasets. In particular, HyperATSA achieves the highest efficiency on both MAMS and Rest14 while remaining competitive on Lap14 with only a negligible increase in parameter count. Compared to YORO, which employs multiple graph structures and additional fusion mechanisms, HyperATSA achieves both higher accuracy and better parameter efficiency with a simpler adaptive hypergraph construction process.

Overall, these results further support the claim

	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	87.76	82.76
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 9: Clustering results on the Lap14 and Rest14 datasets showing Accuracy and F1 scores (%) under different linkage methods and distance metrics. Results for the Cosine+Ward setting are omitted because Ward linkage is only theoretically valid under Euclidean distance.

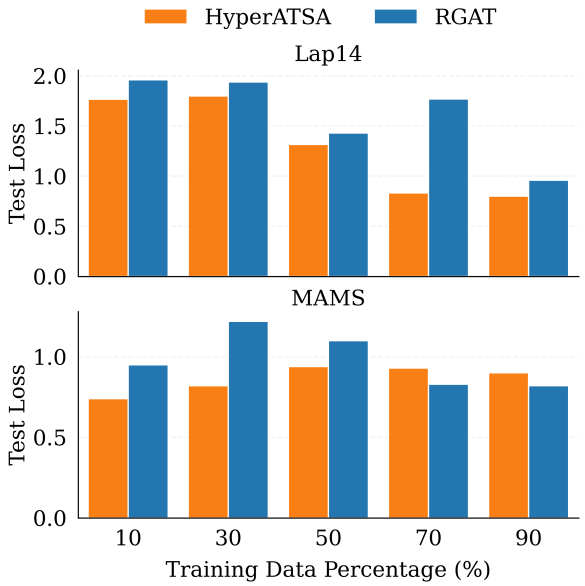

Figure 5: Comparison of test loss between HyperATSA and the graph-based model RGAT on the Lap14 and MAMS datasets.

Model	**MAMS**		**Rest14**		**Lap14**	
	P	**Acc/P**	**P**	**Acc/P**	**P**	**Acc/P**
RGAT	1.10	75.41	1.10	77.97	**1.10**	73.00
YORO	1.15	73.22	1.15	72.77	1.15	67.37
HyperATSA	**1.10**	**76.87**	**1.10**	**78.87**	1.11	**73.14**

Table 10: Model efficiency comparison based on parameter count and accuracy-per-parameter (Acc/P). Parameters are reported in hundreds of millions.

that HyperATSA provides a more robust and parameter-efficient alternative to existing graph-based and multi-graph approaches for ATSA, while maintaining strong generalization across varying data regimes.

D Linkage and Distance Analysis

We analyze the effect of different linkage criteria and distance metrics used in hierarchical clustering for hypergraph construction. As shown in Table 9, 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 ATSA.

E Sensitivity Analysis of λ and ρ

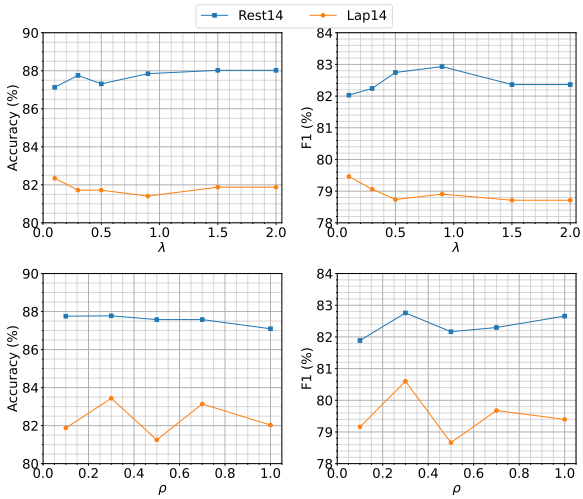

Figure 6: Sensitivity analysis of the adaptive cutoff parameters λ and ρ. The top row shows how Accuracy and F1 (%) vary with λ, and the bottom row illustrates the effect of ρ.

To evaluate the robustness of the adaptive cut-

off mechanism in our hypergraph construction, we conduct a sensitivity analysis over the parameters λ and ρ. The parameter λ appears in Equation 6, where it scales the standard deviation of recent merge dissimilarities. The observed trend in the top row of Figure 6 reveals that as λ increases from 0 to approximately 0.5-1.0, both Accuracy and F1 score show a mild but consistent improvement before plateauing. This behavior reflects how λ governs the fallback cutoff sensitivity during hierarchical clustering: lower λ values make the model overly aggressive in merging clusters, which can conflate distinct aspect or opinion tokens into a single hyperedge. Such under-segmentation causes loss of fine-grained sentiment distinctions, slightly reducing classification performance. Conversely, when λ becomes too large, the cutoff becomes overly conservative, producing many small clusters that fragment coherent semantic groups, an instance of over-segmentation. The plateau observed beyond $\lambda \approx 1.0$ suggests that the model's adaptive cutoff mechanism effectively self-regulates, where further increases in λ have diminishing influence because the elbow-based criterion dominates in most cases. From this behavior, we can infer that a moderate λ allows the model to achieve a structurally balanced hypergraph, capturing both local token coherence and broader semantic grouping.

The parameter $\rho \in (0, 1]$ controls the fraction of recent merges considered when computing the second-order acceleration signal used to detect the elbow point. It effectively governs the temporal window of the clustering process, determining how local or global the merge compactness trends are when deciding where to cut the dendrogram. The bottom row of Figure 6 illustrates how the model's performance varies with respect to the parameter ρ. The stable performance observed for ρ between 0.2 and 0.4 indicates that the adaptive cutoff achieves its best balance when it integrates both local and global clustering information. At lower ρ values, the window of recent dissimilarities is too narrow, causing the model to rely heavily on fine-grained fluctuations in merge distances. This results in a noisy cutoff that can overemphasize short-range dependencies and generate excessively fragmented hyperedges. In contrast, higher ρ values expand the merge window excessively, smoothing out meaningful local variations and leading to overly coarse clusters that blur sentiment distinctions between aspects.

Dataset	Split	Pos. #	Neu. #	Neg. #
Lap14	Train	994	464	870
	Test	341	169	128
Rest14	Train	2,164	637	807
	Test	728	196	182
MAMS	Train	3,380	5,042	2,764
	Dev	403	604	325
	Test	400	607	329

Table 11: Statistics of datasets used for ATSA. Pos.#, Neu.#, and Neg.# denote the number of samples with positive, neutral, and negative sentiment labels, respectively.

F Effect of Token-Level Preprocessing

	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**	**84.56**	**83.74**

Table 12: Ablation study on the impact of token-level preprocessing strategies using the RoBERTa encoder

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 12, removing stop words substantially degrades performance across all datasets, highlighting their importance in maintaining syntactic and semantic coherence during hypergraph construction. Excluding sentence or tokenizer markers also reduces performance, though less severely. The best results are achieved when both stop words and special tokens are retained, suggesting that these contextual cues enable the clustering stage to capture clearer structural boundaries. Overall, token-level cues, often discarded in standard preprocessing, prove essential for stable hypergraph induction and improved sentiment prediction.

Acknowledgments

This research was supported by the Internal Research Fellowship, PES University.