NYCU Speech Lab at SemEval-2026 Task 3: Heterogeneous Model Ensemble with Adaptive Weighted Voting for Dimensional Aspect Sentiment Quadruplet Extraction

Hao-Chun Hsieh **Cheng-En Wu** **Yuan-Fu Liao**†
Institute of Artificial Intelligence Innovation
National Yang Ming Chiao Tung University
†yfliao@nycu.edu.tw

Abstract

SemEval-2026 Task 3 (DimABSA) includes Dimensional Aspect Sentiment Quadruplet Extraction (DimASQP), which requires extracting structured tuples—aspect term, aspect category, and opinion term—together with continuous valence–arousal (VA) values from reviews (Yu et al., 2026a). In this work, we participate in Track A, Subtask 3. We describe NYCU Speech Lab's submission for the Chinese Restaurant and Laptop domains. Our system is a post-processing ensemble over heterogeneous architectures: LoRA/QLoRA fine-tuned decoder-only LLMs, a fine-tuned encoder-only model, and (optionally) prompted API-based LLMs. To improve robustness under the continuous F1 (cF1) metric, we use validation-calibrated **weighted voting** for tuple selection and **weighted VA fusion** for numerical aggregation, with strict output validation to enforce task constraints. Experiments on a held-out validation split show consistent gains over single models and clarify the precision–recall trade-offs induced by the voting threshold. On the organizers' released (tentative) test leaderboard snapshot, our submission ranks first in both domains.

1 Introduction

Dimensional Aspect-Based Sentiment Analysis (DimABSA) extends aspect-based sentiment analysis by replacing categorical polarity labels with continuous affective values, typically modeled in a valence–arousal (VA) space (Russell, 1980; Lee et al., 2026). SemEval-2026 Task 3 (Yu et al., 2026b) instantiates this setting and includes Dimensional Aspect Sentiment Quadruplet Extraction (DimASQP), where a system must jointly predict an *aspect span* t, an *aspect category* c, an *opinion span* o, and a *VA* pair (v, a) for each sentence (Yu et al., 2016).

A key difficulty is that the evaluation metric (cF1) couples *discrete extraction* with *continuous regression* (Lee et al., 2026): spurious tuples reduce precision sharply, while small VA errors reduce the contribution of otherwise correct tuples. In our experiments, a single architecture often exhibits a domain-dependent ceiling under this coupling. Decoder-only LLMs better capture implicit aspects/opinions but are prone to format violations and hallucinated tuples, whereas encoder-only models are more stable in boundary detection but less effective on implicit sentiment and long-range dependencies. These complementary error profiles are amplified across domains (Restaurant vs. Laptop), motivating a cross-architecture ensemble.

We therefore propose a **heterogeneous ensemble** that combines: (i) LoRA/QLoRA fine-tuned decoder-only LLMs (Qwen, Gemma, Llama), (ii) a fine-tuned encoder-only RoBERTa model, and (optionally) (iii) prompted API-based LLMs as additional voters. We aggregate tuples with **validation-calibrated weighted voting** and fuse VA values with a **weight-normalized average** while enforcing strict output validation. Our contributions are:

- A heterogeneous ensemble for DimASQP that leverages complementary inductive biases of decoder-only and encoder-only architectures, with optional API-based voters.
- A validation-calibrated weighted voting rule for tuple selection, paired with strict output validation to reduce malformed outputs and out-of-range VA predictions.
- An analysis of hyperparameter calibration (fusion choice and voting threshold) that makes the system configuration self-contained and reproducible.

The code for our system is available on GitHub[1].

2 Related Work

LLMs have shifted ABSA toward generative formulations (Yan et al., 2021), but are sensitive to

[1] https://github.com/QuAAAAA/ensemble

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 1568–1574
July 3-4, 2026

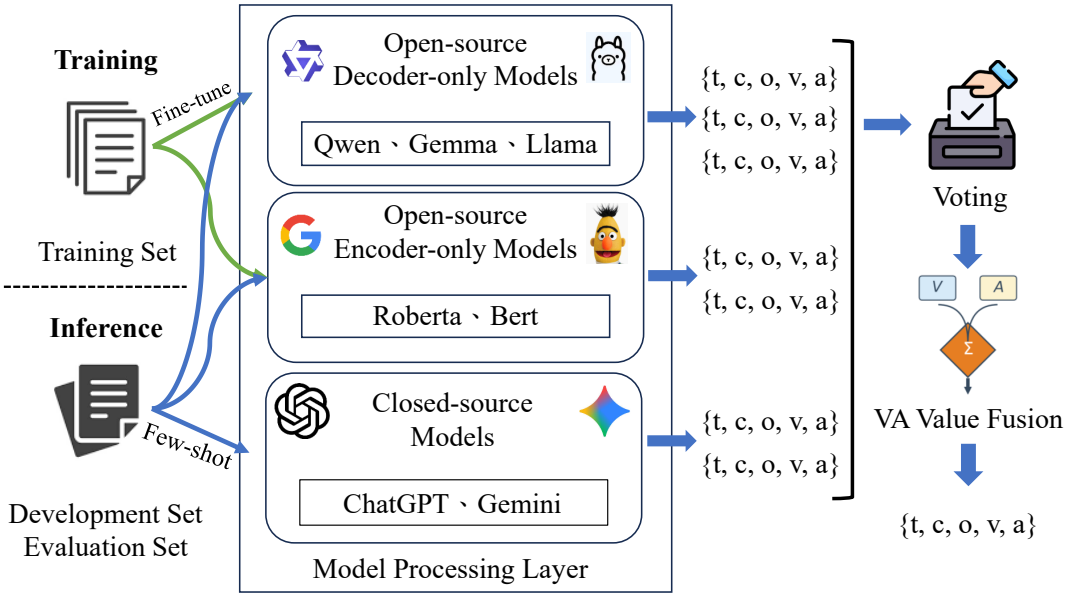

Figure 1: Architectural overview of the NYCU Speech Lab system. Predictions from heterogeneous models are aligned by a tuple key, aggregated via weighted voting, and fused for continuous VA values.

prompting and may hallucinate structured elements or produce invalid formats (Zhang et al., 2023). Prompting strategies such as multi-view prompting (Gou et al., 2023) improve robustness but do not eliminate variance across domains. Ensembling has long been used to improve stability by combining complementary models (Yin et al., 2023; Lin et al., 2024), and weighted fusion often outperforms uniform voting when base models have different reliability profiles.

DimASQP introduces an additional challenge: the evaluation metric couples discrete extraction quality with continuous VA error through cF1 (Lee et al., 2026). Accordingly, an effective system must manage both (i) *tuple correctness* and (ii) *numerical consistency*. Our work focuses on a lightweight post-processing ensemble that unifies these two requirements through validation-calibrated tuple voting and VA fusion.

3 System Overview

Our system is a post-processing ensemble (Figure 1). Each base model outputs a set of DimASQP tuples for an input sentence. We then (i) normalize and validate outputs, (ii) align tuples across models by a unique key, (iii) select tuples via voting, and (iv) fuse the continuous VA values for accepted tuples.

3.1 Task Output and Validation

A DimASQP tuple is represented as (t, c, o, v, a), where t is the extracted aspect term span, c is the aspect category, o is the extracted opinion span, and (v, a) are the valence and arousal values. We define the categorical key $k = (t, c, o)$; voting and metric matching are performed on k, while the associated VA vector is $\mathbf{y}_k = [v, a]$.

Following the DimABSA specification, we apply a lightweight normalization and validation pipeline to ensure that the final submission is always schema-compliant. Table 1 summarizes these checks.

Table 1: Output validity checks applied before alignment and voting.

Stage	Rule	Action
Schema & inventory	Parsed as a 5-tuple (t, c, o, v, a) and c in the task inventory	Discard
VA range	$v, a \in [1, 9]$ (DimABSA spec) (Lee et al., 2026)	Discard
VA rounding	Round v, a to 2 decimals (DimABSA spec) (Lee et al., 2026)	Round
Span text	Trim whitespace and normalize punctuation for t, o	Normalize

After these checks, all tuples in the submission file satisfy the task schema and numeric constraints (0 invalid tuples by construction).

3.2 cF1 Metric

DimASQP is evaluated by the continuous F1 (cF1) metric, which combines exact matching of categorical elements with regression error in the VA space (Lee et al., 2026). A prediction contributes to the true positive count only if its categorical key $k = (t, c, o)$ exactly matches a gold key in the same sentence; otherwise it contributes 0. For a

categorical match, the continuous true positive is defined as

$$\mathrm{cTP}(k) = 1 - \mathrm{dist}(\mathbf{y}_k, \mathbf{y}_k^*), \tag{1}$$

where $\mathbf{y}_k = [v, a]$ and $\mathbf{y}_k^*$ are the predicted and gold VA pairs for the matched key k. The distance is the Euclidean VA error normalized by the maximum possible distance on the $[1, 9]$ scale, so that $\mathrm{dist} \in [0, 1]$ (Lee et al., 2026). Finally, $\mathrm{cPrec.} = \sum \mathrm{cTP}/|\mathcal{P}|$, $\mathrm{cRec.} = \sum \mathrm{cTP}/|\mathcal{G}|$, and cF1 is their harmonic mean, where $\mathcal{P}$ and $\mathcal{G}$ denote the predicted and gold key sets (Lee et al., 2026).

Compared with conventional tuple-level F1, cF1 additionally discounts each matched tuple by its VA error via $\mathrm{cTP} = 1 - \mathrm{dist}$. Thus, two systems with similar exact-match precision/recall can still differ in cF1 if their VA estimates have different accuracy.

3.3 Tuple Alignment via a Unique Key

Because different models may output tuples in different orders or with minor formatting differences, we align tuples using the **unique key** $k = (t, c, o)$ defined in Section 3.1. All candidate tuples from all models are merged into a candidate set indexed by k. For each candidate key, models cast an existence vote indicating whether they predicted that tuple.

3.4 Adaptive Weighted Voting

We use a weighted voting rule where each model's vote is scaled by its validation performance. Let q_i be the validation cF1 score of model i (computed using the official scorer), and let

$$W_i = \frac{q_i}{\sum_{j=1}^{N} q_j}. \tag{2}$$

For a candidate key k, define $\mathbb{I}_i(k) = 1$ if model i predicts that key, else 0. The ensemble support score is

$$\mathrm{Vote}(k) = \sum_{i=1}^{N} W_i \, \mathbb{I}_i(k). \tag{3}$$

We accept k if $\mathrm{Vote}(k) \geq \tau$, where τ is tuned on the validation set via a grid search over $[0, 1]$ to maximize validation cF1 for the chosen ensemble configuration. Section 5.1 reports the validation sweeps used to select τ and other ensemble hyperparameters.

Hard voting baseline. For comparison, we also evaluate a *hard* (unweighted) voting rule that accepts a tuple if it is predicted by at least m models ("m-vote"), which corresponds to uniform weights $W_i = \frac{1}{N}$ and thresholding the integer vote count. This baseline helps isolate the benefit of validation-calibrated weights.

3.5 VA Value Fusion

For each accepted key k, we fuse continuous VA values using a weight-normalized average over the subset of models that predicted that key:

$$v_{\text{final}}(k) = \frac{\sum_{i \in \mathcal{M}(k)} W_i \, v_{i,k}}{\sum_{i \in \mathcal{M}(k)} W_i}, \tag{4}$$

$$a_{\text{final}}(k) = \frac{\sum_{i \in \mathcal{M}(k)} W_i \, a_{i,k}}{\sum_{i \in \mathcal{M}(k)} W_i}, \tag{5}$$

where $\mathcal{M}(k)$ is the set of models predicting key k. Finally, we clip $(v_{\text{final}}, a_{\text{final}})$ to $[1, 9]$ and round to two decimals, matching the task requirement (Lee et al., 2026).

4 Experimental Setup

4.1 Datasets

We participate in the Chinese Restaurant and Laptop datasets of SemEval-2026 Task 3 Track A. Dataset statistics and task definitions are provided by the organizers (Yu et al., 2026b) and the DimABSA dataset paper (Lee et al., 2026). Following the competition protocol, we held out 300 instances from each domain's training set as a validation split for checkpoint selection and for calibrating ensemble weights and thresholds (Section 5.1).

4.2 Model Architectures

We ensemble a diverse set of architectures:

- **Fine-tuned decoder-only LLMs:** instruction-tuned Qwen3, Gemma 3, and Llama 3.1 variants, adapted with parameter-efficient fine-tuning (LoRA/QLoRA) (Hu et al., 2022; Dettmers et al., 2023; Qwen Team, 2025; Gemma Team, 2025; Meta AI, 2024).
- **Fine-tuned encoder-only model:** `chinese-roberta-wwm-ext-large` as a discriminative baseline with stable span boundaries (Cui et al., 2019).
- **API-accessed LLMs (optional):** We optionally include two API-based LLMs as additional voters in some ensemble variants; this may improve recall but reduces full reproducibility (OpenAI, 2025; Google, 2026).

4.3 Implementation Details

We fine-tuned open-source models via Parameter-Efficient Fine-Tuning (PEFT) with LoRA/QLoRA adapters (Hu et al., 2022; Dettmers et al., 2023), limiting trainable parameters to ∼0.5B. Experiments utilized NVIDIA RTX 4090 GPUs, PyTorch (v2.9.1), and HuggingFace Transformers (v4.56.2). Models were trained for 2–5 epochs with an effective batch size of 4, using varied random seeds to enhance ensemble diversity.

Adhering to the competition protocol, a held-out validation split of 300 instances per domain was used with the official scorer (DimABSA Organizers, 2026a) to: (i) select optimal checkpoints by evaluating lowest loss, highest cF1, and the final training step, (ii) compute validation cF1 scores q_i for ensemble weight normalization (Section 3.4), and (iii) calibrate voting thresholds τ and fusion strategies (Section 5.1).

To mitigate formatting errors, we enforce strict output parsing (Section 3.1): malformed tuples, invalid categories, or out-of-range VA values are discarded, with valid VA scores rounded to two decimals (Lee et al., 2026). For API-accessed models, we compare direct sentence-level prompting with a segmented approach that decomposes long sentences into shorter clauses.

5 Results

5.1 Hyperparameter Selection and Calibration

All ensemble hyperparameters were determined on the held-out validation split using the organizers' official scorer (DimABSA Organizers, 2026a). Calibration is performed in a staged manner to match the ensemble pipeline: we first determine the weighted-voting threshold τ that controls which tuple keys are selected, and only then choose the VA fusion rule that aggregates continuous values for the selected keys.

Concretely, we use the validation split to: (i) compute each model's validation cF1 score q_i and derive weights W_i by normalization (Section 3.4), (ii) select the threshold τ for weighted voting (Figures 2–3), and (iii) with τ fixed, select the VA fusion strategy (Table 2).

Stage 1: Threshold sweeps for weighted voting We sweep $\tau \in [0.00, 1.00]$ on the held-out validation split and report the resulting precision/recall/cF1 trade-offs. For the sweep, VA values are fused with a fixed, simple average so that the threshold search primarily reflects the tuple-selection (precision–recall) trade-off. The tuned thresholds (approximately 0.32–0.40 depending on ensemble composition) lie in the "elbow" region where validation cF1 is maximized.

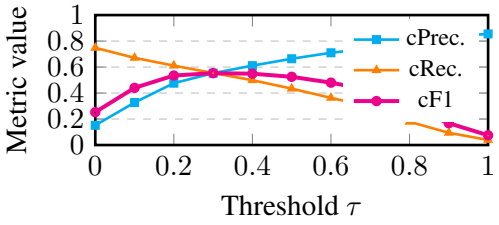

Figure 2: Threshold sweep in the Restaurant domain (validation split).

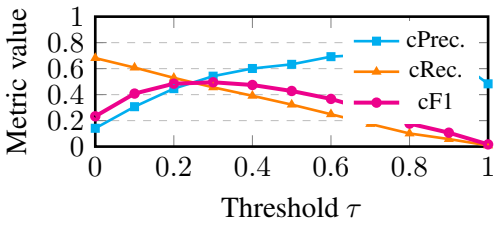

Figure 3: Threshold sweep in the Laptop domain (validation split).

Stage 2: Ablation on VA fusion (with fixed τ) After selecting τ for each ensemble variant, we fix the accepted key set and compare two VA aggregation rules: (1) a weight-normalized average (Section 3.5) and (2) an unweighted average over models predicting the same key. Because fusion affects only the continuous term for already-selected keys, it is conceptually downstream of threshold selection. Results in Table 2 are reported on the validation split and should be interpreted as calibration evidence rather than official test outcomes.

The near-identical performance of Weighted vs. Simple Average in Table 2 results from several factors. Since both strategies operate on the same accepted key set (with τ fixed), variations arise only from subtle shifts in fused VA values. For most aligned tuples, individual model predictions are already numerically close, and our normalized weights W_i are not sufficiently skewed to diverge from a uniform mean. Furthermore, the mandatory $[1, 9]$ clipping and two-decimal rounding absorb minor variances, yielding nearly identical cTP and cF1 outcomes.

Table 2: Ablation results of VA fusion strategies on the validation split. Bold indicates the best performance.

Data	Strategy	cTP	cPrec.	cRec.	cF1
Res	**Weighted**	**1472.98**	**0.5951**	**0.5148**	**0.5521**
	Average	1472.67	0.5950	0.5147	0.5520
Lap	**Weighted**	**828.52**	**0.5568**	**0.4304**	**0.4855**
	Average	828.52	0.5568	0.4304	0.4855

5.2 Experimental Results

5.2.1 Restaurant Domain

Table 3 reports validation performance for representative single models in the Restaurant domain. Among the fine-tuned models, `Qwen3-32B` achieved the strongest cF1, while RoBERTa provided competitive precision with more stable extraction behavior. Table 4 shows that ensembling substantially improves cF1, and that adding closed-source models can further increase recall when used as additional votes.

Table 3: Validation performance of representative models in the Restaurant domain.

Model	Epoch	cF1	cPrec.	cRec.
RoBERTa-wwm-large	3	0.5673	0.5851	0.5505
Llama-3.1	3	0.5413	0.5513	0.5317
Gemma-3	3	0.5501	0.5838	0.5201
Qwen3-14B	3	0.5848	**0.6054**	0.5656
Qwen3-32B	3	**0.5852**	0.5845	**0.5860**
Gemini 3 Pro (prompted)	–	0.3516	0.3330	0.3724
GPT-5 (prompted)	–	0.2745	0.2972	0.2551

Table 4: Validation performance of ensemble configurations in the Restaurant domain.

Ensemble	Threshold τ	cF1	cPrec.	cRec.
Open-source only	0.395	0.6385	**0.6385**	0.6385
+ GPT-5, Gemini	0.3234	**0.6440**	0.6342	**0.6542**

5.2.2 Laptop Domain

Table 5 shows that, in Laptop, the fine-tuned RoBERTa model achieved the strongest single-model cF1, highlighting the advantage of bidirectional attention for span boundary detection. Table 6 demonstrates that our ensemble framework consistently improves cF1, and that weighted voting generally outperforms hard voting by down-weighting weaker models.

Table 5: Validation performance across models in the Laptop domain.

Model	Epoch	cF1	cPrec.	cRec.
RoBERTa-wwm-ext	2	0.3722	0.3988	0.3488
RoBERTa-wwm-ext	3	0.3837	0.3483	**0.4273**
RoBERTa-wwm-large	3	**0.4169**	0.4212	0.4128
Qwen3-14B (4-bit)	4	0.4104	0.4119	0.4089
Qwen3-14B (4-bit)	3	0.3879	0.3732	0.4037
Gemma-3-12B-IT	4	0.3926	0.4284	0.3623
Gemma-3-12B-IT	3	0.4144	**0.4338**	0.3968
Llama-3.1-8B-IT	5	0.3302	0.3241	0.3365
Llama-3.1-8B-IT	4	0.3401	0.3205	0.3623
Qwen3-4B (4-bit)	2	0.2926	0.2865	0.2991
BERT-base-chinese	3	0.4032	0.4144	0.3926
Gemini (prompted)	–	0.2330	0.2179	0.2503
GPT-5 (prompted)	–	0.0841	0.1112	0.0676
Gemini (prompted, segmented)	–	0.1933	0.1771	0.2128
GPT-5 (prompted, segmented)	–	0.0805	0.0799	0.0811

Table 6: Validation performance of ensemble strategies in the Laptop domain.

Strategy		Open-source Only		+ GPT-5 & Gemini	
Voting	**Threshold**	**cF1**	**cPrec.**	**cF1**	**cPrec.**
Hard	2 votes	0.4056	0.3085	0.4534	0.3883
	3 votes	0.4544	0.4073	**0.4810**	0.4714
	4 votes	0.4780	0.4937	0.4738	**0.5926**
Weighted	0.15	0.4058	0.3087	0.4264	0.3305
	0.25	0.4564	0.4128	0.4734	0.4508
	0.35	**0.4816**	**0.5014**	0.4754	0.5437

5.2.3 Official Test Results and Leaderboard Ranking

Beyond validation analyses, the organizers released a *tentative/unofficial* ranking spreadsheet for Subtask 3 (DimABSA Organizers, 2026a,b). In that released snapshot, our submission is listed as rank **1** in both domains (Restaurant: 0.5521; Laptop: 0.4824). Table 9 provides the full ranking list (global view), and Figure 4 summarizes the head of the distribution. Notably, the margins over the 2nd-ranked system are +0.0161 (Restaurant) and +0.0505 (Laptop), suggesting that our heterogeneous ensemble particularly benefits the Laptop domain.

The final ensemble comprises 13 models for Chinese Restaurant and 18 for Laptop, incorporating diverse fine-tuned encoder-only and decoder-only architectures (Table 7, 8). To ensure diversity, checkpoints were selected across 2–5 epochs based on either the lowest validation loss or highest cF1. We also integrated zero-shot API-based models (GPT-5, Gemini-3-Pro-Preview). Weights W_i were primarily derived from validation cF1 scores, though identical architectures in the Laptop domain were assigned consistent weights to demonstrate system robustness to minor manual adjustments.

Table 7: Detailed ensemble members for the **Chinese Restaurant** domain. Weights are mostly derived and normalized based on validation cF1 scores.

Model	Config	Weight
Qwen-3-32B	epoch=3	0.0862
Qwen-3-32B	lowest loss	0.0854
Qwen-3-32B	highest cF1	0.0843
Qwen-3-14B	epoch=3	0.0862
Qwen-3-14B	lowest loss	0.0847
Gemma-3-12B-IT	epoch=3	0.0810
Gemma-3-12B-IT	epoch=4	0.0792
Gemma-3-12B-IT	epoch=5	0.0792
Llama-3.1-8B	epoch=3	0.0797
Llama-3.1-8B	lowest loss	0.0783
RoBERTa-wwm-large	epoch=3	0.0836
Gemini-3-Pro	–	0.0518
GPT-5	–	0.0404
Total (13 Models)		**1.0000**

Table 8: Detailed ensemble members for the **Chinese Laptop** domain. Weights are partially derived and normalized based on validation cF1 scores.

Model	Config	Weight
Qwen-3-14B	epoch=2	0.0593
Qwen-3-14B	epoch=3	0.0593
Qwen-3-14B	epoch=4	0.0593
Qwen-3-4B	epoch=2	0.0593
Qwen-3-4B	epoch=3	0.0593
Qwen-3-4B	epoch=4	0.0593
Gemma-3-12B-IT	epoch=3	0.0593
Gemma-3-12B-IT	epoch=4	0.0593
Llama-3.1-8B	epoch=2	0.0490
Llama-3.1-8B	epoch=3	0.0490
Llama-3.1-8B	epoch=4	0.0490
RoBERTa-wwm-large	epoch=2	0.0623
RoBERTa-wwm-large	epoch=3	0.0623
RoBERTa-wwm-ext	epoch=6	0.0579
RoBERTa-wwm-ext	epoch=7	0.0579
BERT-base-chinese	epoch=7	0.0564
Gemini-3-Pro	–	0.0371
GPT-5	–	0.0445
Total (18 Models)		**1.0000**

Table 9: Organizer-released ranking scores (cF1) for Chinese Restaurant and Laptop in Subtask 3, listed in descending order (DimABSA Organizers, 2026b).

Rank	Restaurant cF1	Laptop cF1
1 (Ours)	**0.5521**	**0.4824**
2	0.5360	0.4319
3	0.5357	0.4316
4	0.5026	0.4016
5	0.4966	0.3968
6	0.4966	0.3836
7	0.4853	0.3745
8	0.4661	0.3703
9	0.4544	0.3478
10	0.4199	0.3139
11	0.3979	0.1996
12	0.3309	0.1900 (baseline)
13	0.2859 (baseline)	0.1885
14	0.1605 (baseline)	0.1124 (baseline)

6 Conclusions

We presented a heterogeneous ensemble framework for SemEval-2026 Task 3 DimASQP. By combining decoder-only LLMs with an encoder-only extractor, and calibrating model weights and voting thresholds on the held-out validation split, our system improves robustness under the cF1 metric by reducing spurious tuples while maintaining recall. Interestingly, our analysis reveals that zero-shot API-based LLMs underperform compared to their fine-tuned open-source counterparts. This is likely due to the lack of domain-specific instruction tuning for structural quadruplet extraction and the inherent difficulty of zero-shot continuous VA regression. We also showed that simple, weight-normalized VA fusion is numerically stable and performs on par with unweighted averaging. Overall, the results support cross-architecture ensembling as a strong and practical baseline for dimensional ABSA tasks where both structured extraction and continuous regression matter.

Acknowledgements

I am deeply grateful to my partner. In our team of two, we carried the weight of this project together; although the path was difficult, I never felt alone because you were there. Beyond the research, the connection we built is what I will cherish most. My sincere thanks also go to Professor Yuan-Fu Liao for his vital guidance and for providing the GPU resources that made this work a reality.

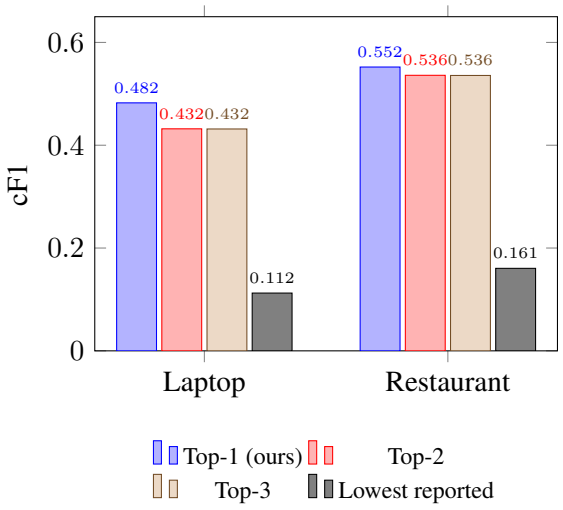

Figure 4: Comparison of official cF1 scores for Subtask 3 (DimASQP). Our ensemble system ranks first in both Chinese Restaurant and Laptop domains, demonstrating a significant performance gain over the Official Baseline (lowest reported). (DimABSA Organizers, 2026b).

References

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2019. Pre-training with whole word masking for chinese BERT. ArXiv:1906.08101.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems*.

DimABSA Organizers. 2026a. SemEval-2026 task 3 (DimABSA) official resources. https://github.com/DimABSA/DimABSA2026. Accessed: 2026-03-01.

DimABSA Organizers. 2026b. Semeval-2026 task 3 track a subtask 3 ranking spreadsheet (tentative). https://docs.google.com/spreadsheets/d/1geyC9aFjAmXFE_7w1LnjGYX4F0y2jttN/edit. Accessed: 2026-03-01.

Gemma Team. 2025. Gemma 3 technical report. ArXiv:2503.19786.

Google. 2026. Gemini 3.1 Pro: A smarter model for your most complex tasks. https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-3-1-pro/. Accessed: 2026-03-01.

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*. ArXiv:2106.09685.

Lung-Hao Lee, Liang-Chih Yu, Natalia Loukashevich, Ilseyar Alimova, Alexander Panchenko, Tzu-Mi Lin, Zhe-Yu Xu, Jian-Yu Zhou, Guangmin Zheng, Jin Wang, Sharanya Awasthi, Jonas Becker, Jan Philip Wahle, Terry Ruas, Shamsuddeen Hassan Muhammad, and Saif M. Mohammad. 2026. Dimabsa: Building multilingual and multidomain datasets for dimensional aspect-based sentiment analysis. *Preprint*, arXiv:2601.23022.

Tzu-Mi Lin, Zhe-Yu Xu, Jian-Yu Zhou, and Lung-Hao Lee. 2024. NYCU-NLP at EXALT 2024: Assembling large language models for cross-lingual emotion and trigger detection. In *Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 505–510, Bangkok, Thailand. Association for Computational Linguistics.

Meta AI. 2024. Introducing Llama 3.1: Our most capable models to date. https://ai.meta.com/blog/meta-llama-3-1/. Accessed: 2026-03-01.

OpenAI. 2025. Introducing GPT-5. https://openai.com/index/introducing-gpt-5/. Accessed: 2026-03-01.

Qwen Team. 2025. Qwen3 technical report. ArXiv:2505.09388.

James A Russell. 1980. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161–1178.

Hang Yan, Junqi Deng, Taiqiang Li, Xipeng Qiu, and Xuanjing Zhang. 2021. A unified generative framework for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 2416–2429.

Zhangyue Yin, Qiushi Sun, Cheng Chang, Qipeng Guo, Junqi Dai, Xuanjing Huang, and Xipeng Qiu. 2023. Exchange-of-thought: Enhancing large language model capabilities through cross-model communication. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Liang-Chih Yu, Jonas Becker, Shamsuddeen Hassan Muhammad, Idris Abdulmumin, Lung-Hao Lee, Ying-Lung Lin, Jin Wang, Jan Philip Wahle, Terry Ruas, Alexander Panchenko, Ilseyar Alimova, Kai-Wei Chang, Lilian Wanzare, Nelson Odhiambo, Bela Gipp, and Saif M. Mohammad. 2026a. SemEval-2026 task 3: Dimensional aspect-based sentiment analysis (DimABSA). In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*, Dublin, Ireland. Association for Computational Linguistics.

Liang-Chih Yu, Jonas Becker, Shamsuddeen Hassan Muhammad, Idris Abdulmumin, Lung-Hao Lee, Ying-Lung Lin, Jin Wang, Jan Philip Wahle, Terry Ruas, Alexander Panchenko, Ilseyar Alimova, Kai-Wei Chang, Lilian Wanzare, Nelson Odhiambo, Bela Gipp, and Saif M. Mohammad. 2026b. SemEval-2026 task 3: Dimensional aspect-based sentiment analysis (DimABSA). In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*. Association for Computational Linguistics.

Liang-Chih Yu, Lung-Hao Lee, Shuai Hao, Jin Wang, Yuncan He, Jun Hu, Jianbo Lai, and Xuejie Zhang. 2016. Building chinese affective resources in valence-arousal dimensions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 540–545. Association for Computational Linguistics.

Wenxuan Zhang, Yue Li, Yifei Deng, Wai Lam Liu, and Lidong Yuan. 2023. Sentiment analysis in the era of large language models: A reality check. *arXiv preprint arXiv:2305.15005*.