Robust Multimodal Sentiment Analysis with Incomplete Modalities via Semantic-aware Completeness based Reconstruction

Han-Jun Choi* **and Byunggill Joe*** **and Saim Shin and Jin Yea Jang†**
Korea Electronics Technology Institute (KETI)
Seongnam, South Korea
{hanjun_c, byunggill, sishin, jinyea.jang}@keti.re.kr

Abstract

Recent multimodal sentiment analysis studies increasingly adopt text-centric fusion approaches to exploit the rich sentiment information inherent in the textual modality. However, these approaches often suffer from performance degradation during inference due to partially missing or noisy data in real-world scenarios, especially when sentiment-related cues are missing. To address this issue, we introduce a new completeness estimation approach that quantifies the degree of sentiment-relevant information preserved in incomplete data to guide the reconstruction of missing semantics. Furthermore, we propose a training strategy that stabilizes multi-task learning while jointly optimizing sentiment prediction and completeness estimation. Extensive experiments and in-depth analyses on three benchmark datasets demonstrate that the proposed approach enables more accurate semantic reconstruction, leading to more precise sentiment prediction.

1 Introduction

Building on the rich sentiment-relevant information in text (Hazarika et al., 2022; Wei et al., 2023), recent Multimodal Sentiment Analysis (MSA) studies have increasingly explored text-centric approaches (Han et al., 2021; Wang et al., 2023a,b; Zhang et al., 2023) that treat text as the dominant modality while considering audio and vision modalities as auxiliary sources. Although these approaches have shown strong performance with complete data, they suffer substantial performance degradation when modalities are partially missing or noisy, particularly when the textual modality is incomplete.

To address this issue, several reconstruction-based approaches have been proposed (Yuan et al., 2021; Zhang et al., 2024a; Zhu et al., 2025). In particular, Zhang et al. (2024a) proposed a missing rate-based completeness estimation method to utilize the missing rate as a reconstruction weight, assuming that a higher missing rate indicates lower completeness, and vice versa. However, we argue that relying solely on the missing rate may fail to accurately capture semantic informativeness.

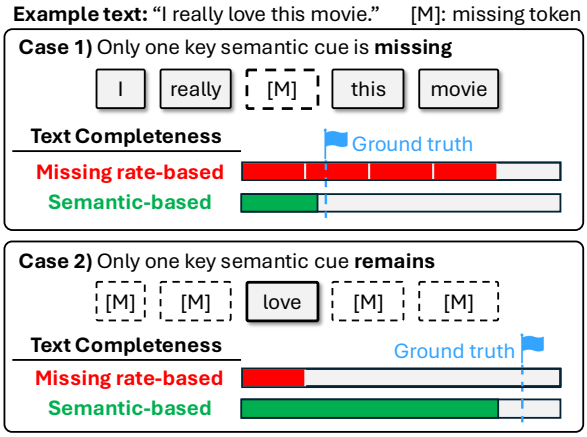


Figure 1: Comparison between missing rate–based and semantic–based completeness. The former is computed as $1 - r$, where $r \in [0, 1)$ denotes the missing rate, while the latter is estimated by our proposed method.

Figure 1 highlights this limitation through two contrasting cases. In Case 1, a missing key sentiment cue [*'love'*] leads to substantial semantic loss despite a low missing rate, causing an overestimation of missing rate-based completeness. On the other hand, in Case 2, it is underestimated even though the sentiment meaning remains intact. Such inaccurate completeness may lead to unreliable reconstruction, which highlights the necessity of a semantic-aware completeness approach that determines completeness based on whether key sentiment cues are missing.

Motivated by the above observations, we propose a semantic-aware completeness estimation approach for more reliable reconstruction of missing textual semantics. Specifically, we first estimate

*Equal contribution.
†Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 339–354
October 24-29, 2026 ©2026 Association for Computational Linguistics

the completeness of incomplete text through a semantic completeness estimator. To supervise the estimator, we introduce Target Probability-based Semantic Completeness (TPSC), which is a pseudo-labeling strategy that generates completeness labels. Meanwhile, the Importance-aware Proxy Feature Generator (IPFG) generates proxy features from the auxiliary modalities by adaptively weighting their contributions for text reconstruction. The estimated completeness is then used as a weighting factor to adaptively combine proxy features and incomplete text, producing reconstructed textual representations. Furthermore, we propose an Alternating Optimization Strategy (AOS) to mitigate gradient conflicts that stem from jointly optimizing the sentiment prediction and completeness estimation tasks on a shared encoder. The contributions of this paper are summarized as follows:

- To the best of our knowledge, we are the first to introduce a semantic-aware completeness estimation approach that quantifies the degree of semantic preservation in incomplete data.
- We propose an optimization strategy that mitigates gradient conflicts in hierarchical multi-task learning, thereby stabilizing training between the completeness estimation and sentiment prediction tasks.
- Extensive experiments on three benchmark datasets with varying missing rates show that our proposed method consistently outperforms 12 competitive baselines, demonstrating the effectiveness of the proposed completeness label in real-world data scenarios.

2 Related Work

Recent studies in MSA have explored various multimodal fusion strategies to integrate textual, acoustic, and visual information (Zadeh et al., 2017; Liu et al., 2018; Tsai et al., 2019; Hazarika et al., 2020; Yu et al., 2021; Sun et al., 2022). These approaches aim to capture cross-modal interactions and leverage complementary information across modalities to improve sentiment prediction. Another line of work focuses on text-centric approaches (Han et al., 2021; Wang et al., 2023a,b; Zhang et al., 2023), which treat text as the dominant modality and utilize audio and visual signals as auxiliary sources to enhance textual representations. However, most existing methods assume that all modalities are fully available during inference, which limits their robustness in real-world scenarios. This issue becomes particularly critical for text-centric approaches, as the loss of important semantic information in text directly undermines the effectiveness of text-centric fusion.

To mitigate the above issue, recent studies have explored reconstruction-based approaches that aim to restore semantic information from missing modalities (Yuan et al., 2021; Lin and Hu, 2023; Zhang et al., 2024a; Zhu et al., 2025; Yang et al., 2025; Shi et al., 2025; Li et al., 2025; Lin et al., 2025). In particular, LNLN (Zhang et al., 2024a) introduces a reconstruction method that restores missing textual semantics using proxy features derived from audio and vision inputs, where the reconstruction weight is estimated from the missing rate of the incomplete text. P-RMF (Zhu et al., 2025) introduces a proxy-driven strategy that dynamically reconstructs incomplete multimodal inputs through cross-modal feature generation and adaptive fusion, further improving robustness under uncertain missing conditions. Additionally, TF-Mamba (Li et al., 2025) proposes a text-enhanced Mamba-based framework that reconstructs missing textual semantics by aligning and enhancing auxiliary modalities through a text-aware modality enhancement module while modeling multimodal dependencies via efficient Mamba-based fusion. While these methods improve robustness to missing modalities, they do not explicitly account for the semantic information loss in incomplete text.

3 Proposed Method

3.1 Problem Definition

Given multimodal inputs from text (t), audio (a), and vision (v) derived from the same utterance, the MSA task aims to predict a continuous sentiment score $y \in \mathbb{R}$.

3.2 Framework Overview

In this work, we introduce a **T**ext **C**ompleteness-based **M**issing **R**econstruction (**TCMR**) framework that estimates textual completeness from incomplete text and reconstructs the missing semantic information by adaptively weighting proxy and incomplete text features based on the estimated completeness.

The overall workflow of TCMR is illustrated in Figure 2. First, each modality input is randomly masked and then embedded into feature representations via modality-specific encoders (①). Next,

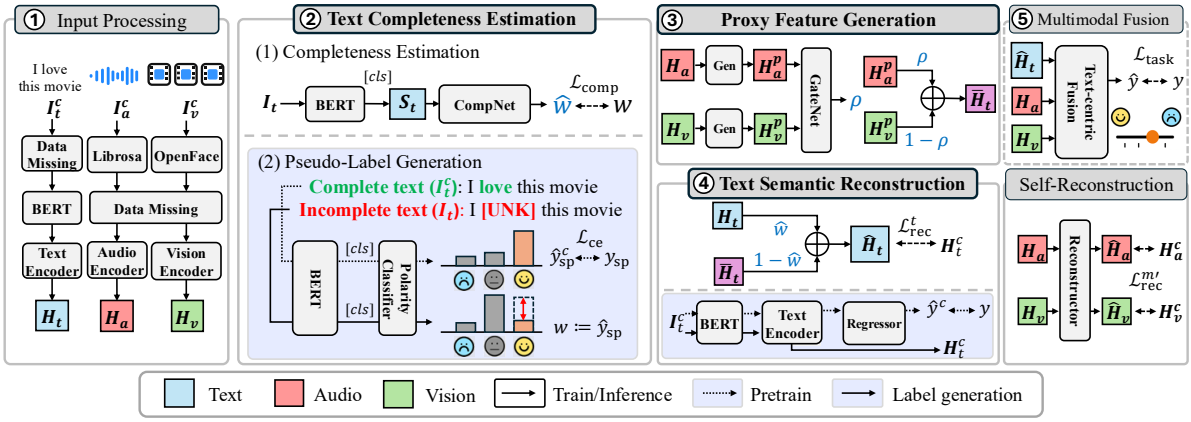


Figure 2: Overview of the proposed framework. Models in the light purple boxes are used only during pretraining or pseudo-labeling and are not part of the TCMR training pipeline.

a completeness estimator predicts how much semantic information remains in incomplete text and reconstructs missing semantics using proxy features derived from auxiliary modalities, weighted by the predicted completeness (②–④). Lastly, a multimodal fusion module performs text-centric fusion to produce the final sentiment score (⑤). Note that the multimodal fusion module is adopted from LNLN, as designing a new fusion module is not the main objective of this work (see Appendix A.4).

3.3 Input Processing

Raw multimodal inputs are processed into high-level embeddings (Figure 2 ①). First, each complete multimodal input I_m^c for modality $m \in \{v, a, t\}$, is processed by modality-specific feature extractors: BERT for text, Librosa for audio, and OpenFace for vision (Devlin et al., 2019; McFee et al., 2015; Baltrušaitis et al., 2016).

Following prior works (Zhang et al., 2024a; Zhu et al., 2025), we adopt a random missing scenario to generate incomplete features $X_m \in \mathbb{R}^{T_m \times d_m}$, where T_m and d_m denote the sequence length and feature dimension. For text, a proportion of tokens in I_t^c are randomly replaced with `[UNK]` according to a missing rate sampled from $[0, 1)$. For audio and vision features, a proportion of the sequence is replaced with zero vectors. Subsequently, the incomplete features are processed by modality-specific Transformer encoders (Vaswani et al., 2017) to obtain the high-level embeddings $H_m \in \mathbb{R}^{T \times d}$:

$$H_m = \text{Transformer}(X_m; \theta_m^{\text{trans}}) \tag{1}$$

3.4 Missing Semantic Reconstruction

In this section, we elaborate on the completeness–guided semantic reconstruction pipeline of TCMR (Figure 2 ②–④), which is the core contribution of our work.

Text Completeness Estimation To estimate the completeness of incomplete text, we design a semantic completeness estimator `CompNet`, which consists of fully connected layers followed by a sigmoid activation (Figure 2 ② –1).

Given the BERT `[CLS]` embedding $S_t \in \mathbb{R}^d$, `CompNet` estimates a completeness weight $\hat{w} \in [0, 1]$:

$$\hat{w} = \text{CompNet}(S_t; \theta^{\text{comp}}), \tag{2}$$

The model is trained to minimize mean squared error between the predicted score $\hat{w}$ and the completeness label w:

$$\mathcal{L}_{\text{comp}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{w}^{(i)} - w^{(i)} \right\|^2, \tag{3}$$

where i denotes the data sample index.

Pseudo-Label Generation In our study, we define textual completeness as the extent to which sentiment-related semantic information is preserved in incomplete text. Following this definition, we introduce a pseudo-labeling strategy to generate completeness labels w for training `CompNet` (Figure 2 ② –2).

To quantify the completeness, we extend the concept of True Class Probability (TCP) (Corbière et al., 2019), defined as the probability assigned to the target class. Our hypothesis is that if a classifier trained on complete text assigns a high TCP to a given complete text, the TCP for the corresponding incomplete text will remain high as long

as sentiment-relevant tokens are preserved. Conversely, the TCP may substantially decrease if key sentiment tokens are missing.

Based on this insight, we design a sentiment polarity classifier that takes only the text modality as input. First, the classifier is trained on the complete text I_t^c:

$$\hat{y}_{\text{sp}} = \text{Classifier}(\text{BERT}^{\text{cls}}(I_t^c); \theta^{\text{cls}}) \quad (4)$$

$$\mathcal{L}_{\text{ce}} = -\sum_{c=1}^{C} y_{sp}^{(c)} \log \hat{y}_{sp}^{(c)} \quad (5)$$

where y_{sp} denotes the polarity class obtained by discretizing the continuous sentiment score according to predefined thresholds. After training, we perform inference on incomplete text using the pretrained classifier, and the probability assigned to the ground-truth polarity class y_{sp} is used as the pseudo completeness label $w \in [0, 1]$:

$$w^{(i)} \triangleq p\left(y_{\text{sp}}^{(i)} \mid \text{BERT}^{\text{cls}}(I_t); \theta_{\text{pre}}^{\text{cls}}\right), \quad (6)$$

where $\theta_{\text{pre}}^{\text{cls}}$ are the pretrained classifier parameters. The definition of sentiment polarity classes and the choice of the optimal number of classes are provided in Appendix B.1 and Appendix B.2, respectively.

Proxy Feature Generation Prior work (Zhang et al., 2024a) generates proxy features without considering the relative importance of auxiliary modalities for reconstruction. However, the contributions of each modality to the reconstruction can vary under random missing conditions.

Therefore, we design an Importance-aware Proxy Feature Generator (IPFG), which consists of Transformer-based modality-specific generators and an MLP-based gating network with a sigmoid:

$$H_{m'}^{p} = \text{Gen}(H_{m'}; \theta_{m'}^{\text{gen}}), \quad (7)$$

$$\rho = \text{Gate}([H_a^p, H_v^p]; \theta^{\text{gate}}), \quad (8)$$

$$\bar{H}_t = \rho \odot H_a^p + (1 - \rho) \odot H_v^p. \quad (9)$$

where $m' \in \{a, v\}$ denotes the auxiliary modalities, $\odot$ denotes element-wise multiplication, and $\rho \in [0, 1]$ is a gating weight that adaptively balances the contributions of the auxiliary modalities.

Semantic Reconstruction The reconstructed text representation $\hat{H}_t \in \mathbb{R}^{T \times d}$ is generated by integrating the incomplete text representation H_t with the generated proxy feature $\bar{H}_t$, guided by the predicted completeness weight $\hat{w}$ from `CompNet`:

$$\hat{H}_t = \hat{w} H_t + (1 - \hat{w}) \bar{H}_t. \quad (10)$$

To encourage the reconstructed features to be similar to the corresponding complete features, we minimize the mean squared error loss $\mathcal{L}_{rec}^{t}$. For the auxiliary modalities, following prior work (Zhang et al., 2024a), we adopt a self-reconstruction method that takes $H_{m'}$ as input and predicts the corresponding complete representation $\hat{H}_{m'}$. The reconstruction losses are defined as:

$$\mathcal{L}_{\text{rec}}^{m} = \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{H}_m^{(i)} - H_m^{c,(i)} \right\|^2, \quad (11)$$

where $H_m^{c,(i)}$ denotes the complete feature of modality m obtained from a model trained with the corresponding complete input.

3.5 Training Objective

TCMR predicts the final sentiment score $\hat{y}$ through the text-centric multimodal fusion module. The model is trained using the following loss:

$$\mathcal{L}_{\text{task}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{y}^{(i)} - y^{(i)} \right\|^2. \quad (12)$$

The overall training objective is defined as:

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{comp}} + \beta \mathcal{L}_{\text{rec}}^{t} + \gamma \mathcal{L}_{\text{rec}}^{m'} + \sigma \mathcal{L}_{\text{task}}. \quad (13)$$

where α, β, γ, and σ are hyperparameters.

3.6 Alternating Optimization Strategy

We found that naively minimizing $\mathcal{L}_{\text{total}}$ in an end-to-end manner often leads to unstable training, where $\mathcal{L}_{\text{comp}}$ is effectively ignored and `CompNet` remains stuck at a poor local minimum. We attribute this behavior to gradient conflict (Yu et al., 2020a). This phenomenon commonly arises in multi-task learning, where the gradients of a dominant task suppress the optimization of relatively weaker tasks. In our case, $\mathcal{L}_{\text{task}}$ can also be minimized through shortcut solutions that bypass learning completeness estimation, causing the optimization of $\mathcal{L}_{\text{comp}}$ to be dominated during training.

To address this issue, we introduce a simple yet effective training strategy called Alternating Optimization Strategy (AOS). We first partition the total objective $\mathcal{L}_{\text{total}}$ into two groups according to their respective roles:

$$\mathcal{L}_{\text{comp}}, \quad \mathcal{L}_{\text{other}} = \beta\, \mathcal{L}_{\text{rec}}^{t} + \gamma\, \mathcal{L}_{\text{rec}}^{m'} + \sigma\, \mathcal{L}_{\text{task}}. \quad (14)$$

Algorithm 1 Alternating Optimization Strategy

Input: dataset $\mathcal{D}$, epochs E, batch size B, weights $\alpha, \beta, \gamma, \sigma$.
Params: $\Theta^{\text{comp}} = \theta^{\text{comp}} \cup \theta^{\text{bert}}$, Θ^{other} for the remaining modules including θ^{bert}.
Opt: Opt_{comp}, $\text{Opt}_{\text{other}}$.

for $e = 1$ to E **do**
  Phase 1: optimize Θ^{comp} to minimize $\mathcal{L}_{\text{comp}}$
  for mini-batch $\mathcal{B} \subset \mathcal{D}$, $|\mathcal{B}| = B$ **do**
    $\mathcal{L}_{\text{comp}} \leftarrow \text{LossComp}(\mathcal{B};\ \Theta^{\text{comp}})$
    $\mathcal{L}_{\text{comp}} \leftarrow \alpha\, \mathcal{L}_{\text{comp}}$
    $\text{Opt}_{\text{comp}}.\text{step}\big(\nabla_{\Theta^{\text{comp}}} \mathcal{L}_{\text{comp}}\big)$
  end for
  Phase 2: optimize Θ^{other} to minimize $\mathcal{L}_{\text{other}}$
  for mini-batch $\mathcal{B} \subset \mathcal{D}$, $|\mathcal{B}| = B$ **do**
    $\mathcal{L}_{\text{task}} \leftarrow \text{LossTask}(\mathcal{B};\ \Theta^{\text{other}})$
    $\mathcal{L}^{t}_{\text{rec}} \leftarrow \text{LossRecText}(\mathcal{B};\ \Theta^{\text{other}})$
    $\mathcal{L}^{m'}_{\text{rec}} \leftarrow \text{LossRecAux}(\mathcal{B};\ \Theta^{\text{other}})$
    $\mathcal{L}_{\text{other}} \leftarrow \beta\, \mathcal{L}^{t}_{\text{rec}} + \gamma\, \mathcal{L}^{m'}_{\text{rec}} + \sigma\, \mathcal{L}_{\text{task}}$
    $\text{Opt}_{\text{other}}.\text{step}(\nabla_{\Theta^{\text{other}}} \mathcal{L}_{\text{other}})$
  end for
end for

As shown in Algorithm 1, AOS performs a two-phase optimization within each training epoch. In Phase 1, we focus on completeness learning by minimizing $\mathcal{L}_{\text{comp}}$, while fixing the remaining parameters and optimizing only Θ^{comp}. In Phase 2, we fix the learned θ^{comp} and optimize Θ^{other} by minimizing $\mathcal{L}_{\text{other}}$. At this stage, the model focuses on reconstruction and sentiment prediction based on the completeness estimation learned in Phase 1. By alternately optimizing the two objectives, AOS stabilizes the training of `CompNet` and enables the remaining modules to be effectively trained according to their respective objectives. Figure 3 illustrates this alternating optimization process, and further experimental analysis of gradient conflicts is provided in Appendix C.

4 Experiments

4.1 Experimental Setup

Dataset and Evaluation Metrics We conduct experiments on three MSA benchmark datasets: MOSI (Zadeh et al., 2016), MOSEI (Bagher Zadeh et al., 2018), and SIMS (Yu et al., 2020b). In MOSI and MOSEI, sentiment scores are annotated with continuous values ranging from -3 (strongly negative) to $+3$ (strongly positive), while SIMS provides sentiment labels on a scale from -1 (negative) to $+1$ (positive). Model performance is evaluated using Acc-2, Acc-5, Acc-7, F1, MAE, and Corr on MOSI and MOSEI, and Acc-2, Acc-3, Acc-5, F1, MAE, and Corr on SIMS. Dataset statistics and metric definitions are provided in Appendix A.1 and Appendix A.3, respectively.

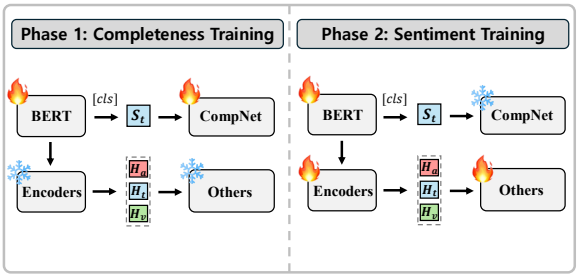


Figure 3: Illustration of the AOS optimization process.

Training and Evaluation Settings Following previous studies (Zhang et al., 2024a; Zhu et al., 2025), we adopt a *partial random missing* scenario. During training, each modality is randomly erased according to a missing rate r sampled from a uniform distribution over the range $[0, 1.0)$. For each of the three seeds, the best model is selected based on its performance at a missing rate of $r = 0.5$ on the validation set. For testing, we conduct experiments across missing rates ranging from 0 to 0.9 with an increment of 0.1, resulting in ten evaluation settings for each seed.

4.2 Main Experimental Results

Baseline Models We select baseline models with publicly available implementations to ensure reproducibility. For a fair comparison, we reproduce all baseline models using publicly available implementations. For MISA, Self-MM, MMIM, CENet, TETFN, TFR-Net, and ALMT, we use the MMSA implementations (Mao et al., 2022). For LNLN[1], P-RMF[2], and TF-Mamba[3], we use their official repositories. In addition, we implement three variants of TCMR with different completeness labeling strategies to analyze their effectiveness:

- **TCMR-TPSC**: estimates completeness using *Target Probability-based Semantic Completeness* (TPSC), as defined in Section 3.4. Note that unless otherwise specified, TCMR refers to TCMR-TPSC for simplicity.

[1] https://github.com/Haoyu-ha/LNLN
[2] https://github.com/aoqzhu/P-RMF
[3] https://github.com/codemous/TF-Mamba

Table 1: Comparison of the overall performance on MOSI and MOSEI datasets.

Method	MOSI						MOSEI					
	Acc-7	Acc-5	Non0 Acc / F1	Has0 Acc / F1	MAE	Corr	Acc-7	Acc-5	Non0 Acc / F1	Has0 Acc / F1	MAE	Corr
MISA	29.03	31.61	68.77 / 68.67	67.94 / 67.72	1.1637	47.57	43.89	44.43	72.22 / 68.21	74.16 / 71.24	0.7346	43.57
Self-MM	30.38	33.69	68.83 / 68.63	68.65 / 69.34	1.1843	47.25	46.45	47.36	73.02 / 71.43	72.66 / 71.82	0.6818	53.68
MMIM	30.67	34.31	70.19 / 69.87	69.59 / 69.15	1.1649	49.01	44.89	45.42	74.29 / 73.19	73.46 / 73.19	0.7032	52.75
CENet	29.49	32.90	69.88 / 69.94	69.43 / 69.39	1.1801	48.24	**47.36**	**48.24**	77.01 / 77.26	75.96 / 76.23	0.6622	58.24
TETFN	29.86	32.56	70.62 / 70.67	69.85 / 69.79	1.1327	49.16	46.78	47.83	77.87 / 77.45	76.11 / 76.37	0.6741	58.31
TFR-Net	28.22	30.31	70.88 / 70.73	70.18 / 69.93	1.1454	50.05	46.08	46.47	75.46 / 74.10	74.18 / 73.61	0.6784	56.24
ALMT	29.57	32.05	71.51 / 71.48	70.52 / 70.39	1.1671	47.57	46.66	47.37	76.83 / 76.41	74.47 / 74.84	0.6749	56.45
LNLN	31.36	34.43	70.00 / 69.99	69.51 / 69.41	1.1450	48.08	46.36	47.14	77.79 / 77.27	76.43 / 76.58	0.6698	58.17
P-RMF	28.91	31.24	69.91 / 69.57	69.28 / 68.84	1.1302	50.07	47.01	47.82	**78.07 / 77.76**	77.18 / 76.91	0.6667	58.45
TF-Mamba	28.85	31.46	71.05 / 71.11	70.73 / 70.69	1.1228	51.97	45.11	46.09	76.11 / 76.17	72.54 / 73.49	0.6891	57.51
TCMR-MRSC	30.39	33.36	70.84 / 70.95	69.81 / 69.82	1.1061	50.86	46.61	47.46	77.65 / 76.74	76.91 / 76.64	0.6673	58.46
TCMR-LDSC	31.63	35.38	70.61 / 70.55	69.83 / 69.58	1.1322	49.07	47.13	47.87	77.52 / 76.97	76.16 / 76.29	0.6641	58.51
TCMR-TPSC	**32.80**	**36.73**	**72.01 / 71.67**	**70.93 / 70.49**	**1.0721**	**52.28**	47.27	48.16	77.99 / 76.99	**77.59 / 77.21**	**0.6614**	**58.63**

Method	Acc-5	Acc-3	Acc-2	F1	MAE	Corr
MISA	32.97	56.05	71.25	66.95	0.5731	0.319
Self-MM	33.57	55.82	69.02	68.38	0.5243	0.363
MMIM	30.33	55.04	69.46	66.74	0.5629	0.301
CENet	21.61	53.48	69.23	56.97	0.6304	0.014
TETFN	**34.91**	56.01	70.85	69.17	0.5439	0.342
TFR-Net	27.21	54.26	69.37	56.82	0.6258	0.091
ALMT	30.99	53.23	70.65	67.34	0.5445	0.311
LNLN	32.17	56.77	71.51	69.21	0.5423	0.345
P-RMF	32.72	57.30	70.58	**70.36**	0.5432	0.361
TF-Mamba	33.05	55.18	69.88	69.24	0.5235	0.359
TCMR-MRSC	33.27	55.46	70.03	68.76	0.5446	0.341
TCMR-TPSC	31.15	**57.45**	**72.81**	68.88	**0.5212**	**0.379**

Table 2: Comparison of the overall performance on the SIMS dataset. Note: TCMR-LDSC is excluded since it is based on an English sentiment lexicon.

- **TCMR-MRSC**: estimates completeness using *Missing Rate-based Semantic Completeness* (MRSC), which measures completeness according to the missing rate of text.

- **TCMR-LDSC**: To better analyze the effectiveness of TPSC, we design *Lexicon-Derived Semantic Completeness* (LDSC). LDSC generates completeness labels based on sentiment scores of individual tokens obtained from SentiWordNet (Baccianella et al., 2010). Specifically, we first compute the sentiment strength of the complete text by accumulating the sentiment scores of all tokens. Then, the same procedure is applied to the incomplete text. As a result, the completeness label is defined as the ratio between the two strengths. Further implementation details are provided in Appendix D.1.

Overall Performance Tables 1–2 present the evaluation results of various models on three MSA benchmark datasets. On the MOSI dataset, TCMR-TPSC achieves state-of-the-art performance across all evaluation metrics. In particular, TCMR-TPSC improves Acc-5 by 6.68% and reduces MAE by 6.37% compared to LNLN. Among the `TCMR-` variants, TCMR-TPSC consistently achieves the best performance across all datasets. This is because TCMR-MRSC struggles to capture the actual semantic loss from the text, as it relies solely on the missing rate. TCMR-LDSC, on the other hand, estimates completeness based on sentiment scores assigned to individual tokens, which does not consider contextual semantics and can lead to inaccurate completeness labels. These results further highlight the importance of semantic-aware completeness estimation. Further analyses with competitive baselines are provided in Appendix E.

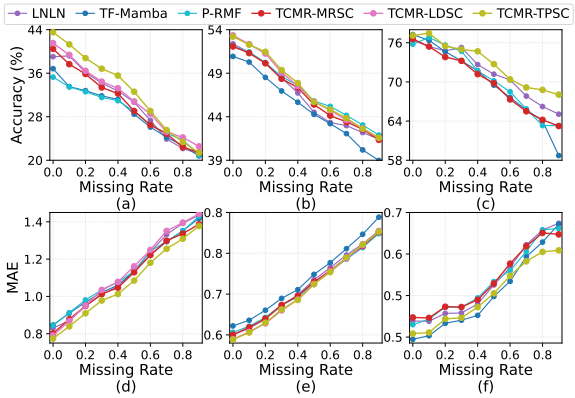


Figure 4: Performance curves under various missing rates. (a)–(c): Accuracy on MOSI (Acc-7), MOSEI (Has0 Acc), and SIMS (Acc-2). (d)–(f): MAE on MOSI, MOSEI, and SIMS.

On the MOSEI and SIMS datasets, TCMR-TPSC consistently shows strong performance

Table 3: A comprehensive ablation study of the proposed TCMR framework on the MOSI and MOSEI datasets.

Method	MOSI						MOSEI					
	Acc-7	Acc-5	Non0 Acc / F1	Has0 Acc / F1	MAE	Corr	Acc-7	Acc-5	Non0 Acc / F1	Has0 Acc / F1	MAE	Corr
w/o AOS												
CompNet-first	27.67	30.65	66.88 / 66.84	66.45 / 66.29	1.1671	43.06	43.33	43.48	65.37 / 64.09	69.14 / 65.16	0.7827	32.61
End2End	29.68	32.07	68.15 / 68.04	67.58 / 67.36	1.1849	49.15	46.82	47.75	77.97 / **77.43**	76.27 / 76.44	0.6619	**58.71**
CompNet-later	32.24	36.17	71.16 / 71.22	70.41 / 70.37	1.0944	50.91	43.11	43.86	73.75 / 71.83	74.58 / 73.47	0.7547	46.85
w/o IPFG	31.31	34.81	69.02 / 68.91	68.65 / 68.42	1.1674	45.89	46.05	47.06	75.92 / 75.80	72.71 / 73.45	0.6816	57.22
TCMR (Full)	**32.80**	**36.73**	**72.01 / 71.67**	**70.93 / 70.49**	**1.0721**	**52.28**	**47.27**	**48.16**	**77.99** / 76.99	**77.59 / 77.21**	**0.6614**	58.63

Table 4: A comprehensive ablation study of the proposed TCMR framework on the SIMS dataset.

Method	Acc-5	Acc-3	Acc-2	F1	MAE	Corr
w/o AOS						
CompNet-first	22.92	53.79	69.82	60.81	0.5764	0.171
End2End	29.26	53.92	68.57	67.71	0.5374	0.321
CompNet-later	29.71	55.49	70.04	68.48	0.5416	0.339
w/o IPFG	30.15	55.57	70.09	66.93	0.5451	0.336
TCMR (Full)	**31.15**	**57.45**	**72.81**	**68.88**	**0.5212**	**0.379**

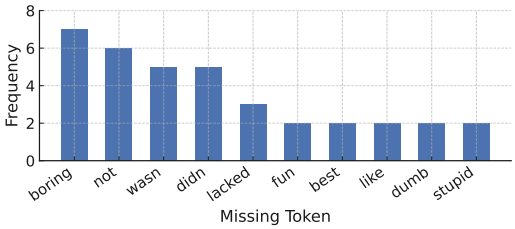
across most evaluation metrics. In particular, on MOSEI, it achieves the best results on Has0 Acc / F1, MAE, and Corr, while remaining competitive on Acc-7, Acc-5, and Non0 Acc / F1. On SIMS, it achieves the best results on Acc-3, Acc-2, MAE, and Corr. While TCMR-TPSC shows slightly lower performance on a few classification metrics compared to CENet and P-RMF, this can be attributed to the discretization involved in computing such metrics. For instance, Acc-7 converts continuous sentiment predictions into discrete classes via rounding. When a prediction lies close to a rounding boundary, it may fall into a different class than the ground truth, leading to misclassification despite a lower regression error. For this reason, regression metrics more accurately reflect the precision of sentiment prediction. A more detailed analysis is provided in Appendix E.1.

To provide a comprehensive comparison with reconstruction-based approaches, we plot performance changes across missing rates in Figure 4. As shown in the figure, TCMR-TPSC consistently outperforms baseline methods, even as the missing rate increases. These results support our motivation that accurately estimating semantic information loss in incomplete text enables effective reconstruction of missing semantics, thereby leading to improved overall performance.

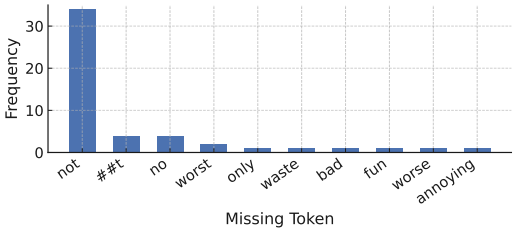
5 Ablation Study

As shown in Table 3 and Table 4, we conduct an ablation study on the MOSI, MOSEI, and SIMS datasets to analyze the contribution of each component. In the w/o AOS setting, we design several alternative multi-task optimization strategies. In `CompNet-first`, we first optimize Θ^{comp}, and then optimize the remaining modules while freezing both the shared BERT encoder and the trained `CompNet`. In contrast, `CompNet-later` first optimizes Θ^{other} using the completeness label w, and then optimizes θ^{comp} while freezing the shared BERT encoder. `End2End` jointly optimizes all model parameters without any staged optimization. In the w/o IPFG setting, we replace IPFG with a simpler proxy feature generator following prior work (Zhang et al., 2024a). Specifically, instead of generating modality-specific proxy features and combining them through a gating mechanism, the audio and vision features are directly used as inputs to a single network that produces the proxy feature.

According to Table 3, `CompNet-first` consistently achieves the lowest performance across all datasets. One possible reason is that freezing the shared BERT encoder after completeness learning prevents it from being further adapted for sentiment prediction, leading to misalignment between the two objectives. This result highlights the necessity of continuously updating the shared BERT encoder across both tasks, while preserving the hierarchical structure from completeness learning to sentiment learning. In contrast, `CompNet-later` achieves relatively competitive performance with `TCMR (Full)` on MOSI, while it performs relatively worse on MOSEI. We attribute this discrepancy to the differences in scale and characteristics between the two datasets. Compared to MOSI, MOSEI contains substantially more samples and covers a wider range of topics, indicating that a broader range of sentiment expression patterns must be learned. As a result, learning completeness estimation by optimizing a



(a) MOSI



(b) MOSEI

Figure 5: Comparison of error-triggering sentiment clue tokens across two benchmark datasets. Each histogram shows the frequency of missing tokens that cause misclassification in the sentiment classifier.

CompNet with relatively few parameters after the shared BERT encoder has already been optimized for ground-truth sentiment labels may be insufficient to capture the diverse patterns present in MOSEI. In this regard, `End2End` shows relatively more competitive performance on MOSEI than on MOSI, which we attribute to MOSEI's greater diversity reducing the influence of the dominant task, thereby naturally alleviating gradient conflict. Nevertheless, its inherent end-to-end nature still results in suboptimal performance compared to `TCMR (Full)`. Removing `IPFG` consistently degrades performance across all datasets. This suggests that it adaptively regulates the contribution of each auxiliary modality for incomplete text reconstruction, effectively filtering redundant information while enhancing complementary semantic cues.

6 In-depth Analysis

Sentiment Token Sensitivity To further examine the effectiveness of the TPSC label, we conduct a sensitivity analysis on sentiment-related tokens. First, we collect samples that are correctly classified by a pretrained sentiment polarity classifier on complete textual inputs. For each sample, we construct a perturbation token set by masking each word token in turn. Then, we count the frequency of tokens whose masking causes a previously correct prediction to become incorrect (detailed procedures are provided in Appendix D.2).

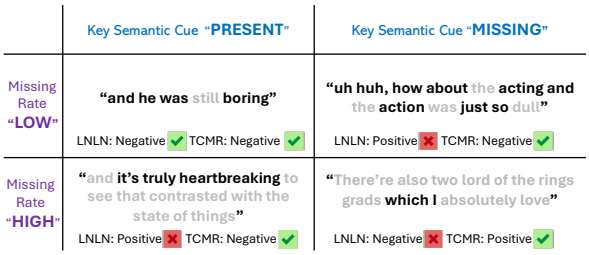


Figure 6: Qualitative comparison between the LNLN and TCMR models under varying missing-text conditions. Each cell presents an example utterance where key semantic cues are either present or missing.

As shown in Figure 5, misclassifications are concentrated on affective tokens such as [*'boring'*], [*'not'*], and [*'fun'*]. This observation supports our hypothesis that TCP indeed reflects the preservation of sentiment-relevant information, as discussed in Section 3.4. Consequently, `CompNet` provides a more accurate estimate of semantic completeness, enabling more reliable reconstruction by maximizing complementary semantic information from auxiliary modalities.

Case Study Figure 6 illustrates how LNLN and TCMR behave under different missing-text conditions. Overall, LNLN fails to accurately reconstruct missing semantics. This stems from overestimating or underestimating the semantic information remaining in incomplete text, leading to incorrect sentiment prediction. In contrast, TCMR produces correct predictions through more accurate completeness estimation. This allows TCMR to adaptively balance its reliance on auxiliary modalities based on the extent of semantic loss, resulting in a more faithful reconstruction. Notably, the bottom-right cell provides an interesting insight. Although LNLN can estimate completeness accurately in this case, due to the high missing rate and the loss of a key sentiment token, it still fails to produce the correct sentiment prediction. This suggests that inaccurate completeness estimation during training may limit LNLN's ability to reconstruct missing semantics. Therefore, accurately estimating the semantic information retained in incomplete text is crucial not only for effective reconstruction but also for stable sentiment prediction.

Comparison of Completeness Labels Table 5 provides a qualitative comparison of different completeness labeling strategies under random missing conditions. Cases 3 and 4 highlight the limitations of MRSC. Since MRSC estimates completeness

Table 5: Qualitative comparison of completeness labels. Each case shows how MRSC, LDSC, and TPSC assign completeness labels depending on whether sentiment-related tokens are missing. In the TPSC column, the value in parentheses denotes $P(y_{sp} \mid \text{complete})$, while the value above it denotes $P(y_{sp} \mid \text{incomplete})$. ✓ and ✗ indicate correct and incorrect completeness estimation, respectively.

Case	Complete Text	Incomplete Text	MRSC	LDSC	TPSC
1	I was happy to see it	I [UNK] happy to see it	0.8333 ✓	0.9888 ✓	0.9526 (0.9580) ✓
2	The soundtrack is good its really good some really great songs	[UNK] soundtrack [UNK] [UNK] [UNK] really [UNK] [UNK] [UNK] [UNK] [UNK]	0.1818 ✓	0.0044 ✓	0.3077 (0.9665) ✓
3	But it was really really awesome	but it was [UNK] [UNK] [UNK]	0.5000 ✗	0.0112 ✓	0.3179 (0.9680) ✓
4	Kids are gonna love the film	[UNK] [UNK] [UNK] love the [UNK]	0.3333 ✗	0.9808 ✓	0.8638 (0.9571) ✓
5	I will admit I'm a big Johnny Depp fan	I will [UNK] I'm [UNK] big Johnny Depp fan	0.7931 ✓	0.0690 ✗	0.9160 (0.9491) ✓
6	and you're better off saving your money uh and maybe renting it when it comes out on uh video so	and [UNK] better [UNK] saving [UNK] money [UNK] [UNK] maybe renting it when it [UNK] out [UNK] uh [UNK] [UNK]	0.5921 ✓	0.9928 ✗	0.4608 (0.6153) ✓
7	I really dig this movie	I [UNK] [UNK] this [UNK]	0.4000 ✗	0.5000 ✗	0.1006 (0.9477) ✓

solely based on the proportion of missing tokens, it cannot capture the actual semantic information loss in the incomplete text. In contrast, LDSC and TPSC more appropriately reflect completeness by considering the semantic importance of sentiment-related tokens. Cases 5–7 reveal the limitations of LDSC. For example, in Cases 5 and 7, words such as [*'fan'*] and [*'dig'*] are used as positive sentiment expressions. However, as SentiWordNet assigns sentiment scores based on the most frequent sense of a word, the assigned score may differ from the sense used in the actual context. Moreover, in Case 6, [*'better'*] is assigned a positive sentiment score in SentiWordNet, leading LDSC to yield a high completeness value even though the overall sentence conveys negative sentiment. In contrast, TPSC produces relatively correct completeness in these challenging cases due to its ability to capture sentiment based on the context of the full sentence rather than individual words.

7 Conclusion

In this work, we reinterpret semantic information loss in incomplete data based on the change in target-class probability between complete and incomplete data to address the performance degradation of text-centric fusion models in MSA caused by noisy or missing data. First, we generate pseudo-labels that provide supervision for training a completeness estimator. We further propose an optimization strategy that mitigates gradient conflicts between the completeness estimation and sentiment prediction tasks, thereby stabilizing hierarchical multi-task learning. We believe the proposed completeness estimation approach can be broadly applied beyond MSA, such as confidence-aware decision making and low-quality data detection or restoration.

Limitations

Although TCMR consistently outperforms existing reconstruction-based MSA models, several limitations remain to be addressed in future work. First, this study only focuses on reconstructing semantic information loss in the textual modality. Extending the TPSC-based completeness estimation to other modalities, such as audio and vision, could improve general multimodal fusion approaches beyond text-centric fusion. Second, TCMR relies on pseudo-labeling, which inherently introduces potential noise in the supervision signal. Future work could mitigate this issue by adopting confidence-aware filtering, using only high-confidence TPSC predictions as pseudo-labels while relying on complementary strategies such as LDSC for less confident samples. For samples where both TPSC and LDSC yield low confidence, human verification could be incorporated to further reduce label noise.

Acknowledgments

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2022-II220608, RS-2022-II220320, and RS-2024-00398115); and the Korea Electronics Technology Institute (KETI) through the project "Development of Modular Humanoid Physical AI Technology for Performing Complex Tasks in Industrial Environments."

References

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

AmirAli Bagher Zadeh, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018. Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2236–2246, Melbourne, Australia. Association for Computational Linguistics.

Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. 2016. OpenFace: An open source facial behavior analysis toolkit. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 794–803. PMLR.

Charles Corbière, Nicolas THOME, Avner Bar-Hen, Matthieu Cord, and Patrick Pérez. 2019. Addressing failure prediction by learning model confidence. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

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

Wei Han, Hui Chen, Alexander Gelbukh, Amir Zadeh, Louis-philippe Morency, and Soujanya Poria. 2021. Bi-bimodal modality fusion for correlation-controlled multimodal sentiment analysis. In *Proceedings of the 2021 International Conference on Multimodal Interaction*, ICMI '21, pages 6–15, New York, NY, USA. Association for Computing Machinery.

Devamanyu Hazarika, Yingting Li, Bo Cheng, Shuai Zhao, Roger Zimmermann, and Soujanya Poria. 2022. Analyzing modality robustness in multimodal sentiment analysis. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 685–696, Seattle, United States. Association for Computational Linguistics.

Devamanyu Hazarika, Roger Zimmermann, and Soujanya Poria. 2020. MISA: Modality-invariant and -specific representations for multimodal sentiment analysis. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, pages 1122–1131, New York, NY, USA. Association for Computing Machinery.

Xiang Li, Xianfu Cheng, Dezhuang Miao, Xiaoming Zhang, and Zhoujun Li. 2025. TF-Mamba: Text-enhanced fusion mamba with missing modalities for robust multimodal sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 11252–11267, Suzhou, China. Association for Computational Linguistics.

Ronghao Lin, Qiaolin He, Sijie Mai, Ying Zeng, Aolin Xiong, Li Huang, Yap-Peng Tan, and Haifeng Hu. 2025. CyIN: Cyclic informative latent space for bridging complete and incomplete multimodal learning. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pages 17605–17642. Curran Associates, Inc.

Ronghao Lin and Haifeng Hu. 2023. MissModal: Increasing robustness to missing modality in multimodal sentiment analysis. *Transactions of the Association for Computational Linguistics*, 11:1686–1702.

Zhun Liu, Ying Shen, Varun Bharadhwaj Lakshminarasimhan, Paul Pu Liang, AmirAli Bagher Zadeh, and Louis-Philippe Morency. 2018. Efficient low-rank multimodal fusion with modality-specific factors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2247–2256, Melbourne, Australia. Association for Computational Linguistics.

Huisheng Mao, Ziqi Yuan, Hua Xu, Wenmeng Yu, Yihe Liu, and Kai Gao. 2022. M-SENA: An integrated platform for multimodal sentiment analysis. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 204–213, Dublin, Ireland. Association for Computational Linguistics.

Brian McFee, Colin Raffel, Dawen Liang, Daniel P. W. Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and music signal analysis in

python. In *Proceedings of the 14th Python in Science Conference, SciPy 2015, Austin, Texas, USA, July 6-12, 2015*, pages 18–24. scipy.org.

Piao Shi, Min Hu, Satoshi Nakagawa, Xiangming Zheng, Xuefeng Shi, and Fuji Ren. 2025. Text-guided reconstruction network for sentiment analysis with uncertain missing modalities. *IEEE Transactions on Affective Computing*, 16(3):1825–1838.

Hao Sun, Hongyi Wang, Jiaqing Liu, Yen-Wei Chen, and Lanfen Lin. 2022. CubeMLP: An mlp-based model for multimodal sentiment analysis and depression estimation. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, pages 3722–3729, New York, NY, USA. Association for Computing Machinery.

Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J. Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Multimodal transformer for unaligned multimodal language sequences. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6558–6569, Florence, Italy. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Di Wang, Xutong Guo, Yumin Tian, Jinhui Liu, Li-Huo He, and Xuemei Luo. 2023a. TETFN: A text enhanced transformer fusion network for multimodal sentiment analysis. *Pattern Recognition*, 136:109259.

Di Wang, Shuai Liu, Quan Wang, Yumin Tian, Lihuo He, and Xinbo Gao. 2023b. Cross-modal enhancement network for multimodal sentiment analysis. *IEEE Transactions on Multimedia*, 25:4909–4921.

Yiwei Wei, Shaozu Yuan, Ruosong Yang, Lei Shen, Zhangmeizhi Li, Longbiao Wang, and Meng Chen. 2023. Tackling modality heterogeneity with multi-view calibration network for multimodal sentiment detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5240–5252, Toronto, Canada. Association for Computational Linguistics.

Mingzheng Yang, Kai Zhang, Yuyang Ye, Yanghai Zhang, Runlong Yu, and Min Hou. 2025. Decoupling and reconstructing: A multimodal sentiment analysis framework towards robustness. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 6803–6811. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020a. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 5824–5836. Curran Associates, Inc.

Wenmeng Yu, Hua Xu, Fanyang Meng, Yilin Zhu, Yixiao Ma, Jiele Wu, Jiyun Zou, and Kaicheng Yang. 2020b. CH-SIMS: A Chinese multimodal sentiment analysis dataset with fine-grained annotation of modality. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3718–3727, Online. Association for Computational Linguistics.

Wenmeng Yu, Hua Xu, Ziqi Yuan, and Jiele Wu. 2021. Learning modality-specific representations with self-supervised multi-task learning for multimodal sentiment analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10790–10797.

Ziqi Yuan, Wei Li, Hua Xu, and Wenmeng Yu. 2021. Transformer-based feature reconstruction network for robust multimodal sentiment analysis. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, pages 4400–4407, New York, NY, USA. Association for Computing Machinery.

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor fusion network for multimodal sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1103–1114, Copenhagen, Denmark. Association for Computational Linguistics.

Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages. *IEEE Intelligent Systems*, 31(6):82–88.

Haoyu Zhang, Wenbin Wang, and Tianshu Yu. 2024a. Towards robust multimodal sentiment analysis with incomplete data. In *Advances in Neural Information Processing Systems*, volume 37, pages 55943–55974. Curran Associates, Inc.

Haoyu Zhang, Yu Wang, Guanghao Yin, Kejun Liu, Yuanyuan Liu, and Tianshu Yu. 2023. Learning language-guided adaptive hyper-modality representation for multimodal sentiment analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 756–767, Singapore. Association for Computational Linguistics.

Zhi Zhang, Jiayi Shen, Congfeng Cao, Gaole Dai, Shiji Zhou, Qizhe Zhang, Shanghang Zhang, and Ekaterina Shutova. 2024b. Proactive gradient conflict mitigation in multi-task learning: A sparse training perspective. *arXiv preprint arXiv:2411.18615*.

Aoqiang Zhu, Min Hu, Xiaohua Wang, Jiaoyun Yang, Yiming Tang, and Ning An. 2025. Proxy-driven robust multimodal sentiment analysis with incomplete data. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22123–22138, Vienna, Austria. Association for Computational Linguistics.

A Implementation Details

A.1 Dataset Statistics

Table 6: Statistics of the MOSI, MOSEI, and SIMS datasets used in our experiments.

	MOSI	MOSEI	SIMS
#Samples	2,199	22,856	2,281
Train / Val / Test	1,284 / 229 / 686	16,326 / 1,871 / 4,659	1,368 / 456 / 457
#Speakers	93	1,000+	–
#Topics	89	250+	–
Source	YouTube	YouTube	Movies / TV
Annotation Type	Utterance-level	Utterance-level	Utterance-level
Sentiment Score	–3 to +3	–3 to +3	–1 to +1

Table 6 provides key statistics of the three benchmark datasets used in our experiments. Among them, MOSEI stands out as substantially larger and more diverse in speakers and topics than MOSI and SIMS, resulting in greater variability across acoustic, visual, and textual modalities. Due to this diversity, MOSEI is considered a more challenging benchmark than MOSI and SIMS.

A.2 Model Configuration

The proposed TCMR model was implemented in PyTorch (v2.2.1) with Python 3.11.7 and trained on a workstation with an NVIDIA GeForce RTX 4090 GPU and an AMD Ryzen Threadripper PRO 5975WX CPU. To offer a thorough specification of our architecture design, Table 7 provides the detailed configurations of all modules within TCMR.

A.3 Evaluation Metrics

Following prior work (Zhang et al., 2024a), we adopt the same evaluation metrics. For MOSI and MOSEI, we report both classification metrics (Acc-7, Acc-5, Non0 Acc / F1, Has0 Acc / F1) and regression metrics (MAE, Corr). The classification metrics are computed by discretizing the continuous sentiment scores into predefined sentiment intervals. For SIMS, we report Acc-2, Acc-3, Acc-5, F1, MAE, and Corr. Non0 Acc denotes negative/positive classification (excluding zero-valued samples), whereas Has0 Acc denotes negative/non-negative classification, where zero-valued samples are treated as non-negative.

A.4 Multimodal Fusion Module

The multimodal fusion module follows the design of prior work (Zhang et al., 2024a), and its key components are briefly summarized here. Starting from the reconstructed text representation $\hat{H}_t$, a refinement Transformer encoder applies self-attention, producing layerwise refined features $H_t^{+(i)}$ with $i \in \{1, \ldots, l^{\text{ref}}\}$. Based on $H_t^{+(i)}$, cross-modal attention integrates complementary cues from vision and audio across multiple layers. At each layer, the current refined text feature $H_t^{+(i)}$ serves as the query and attends to H_v and H_a, while the fused representation updates $H_f^{(i)}$ through residual accumulation. In compact form, $H_f^{(i)} = H_f^{(i-1)} + \mathrm{MHA}(H_t^{+(i)}, H_v) + \mathrm{MHA}(H_t^{+(i)}, H_a)$ where H_f^0 is a learnable embedding and $\mathrm{MHA}(Q, K)$ denotes multi-head attention with query Q and key and value from K. A CrossTransformer then models interactions between $[\, H_f^{(l^{\text{ref}})},\ H_t^{+(l^{\text{ref}})} \,]$, and a regression head outputs the final sentiment prediction $\hat{y}$.

B Sentiment Polarity Classes

B.1 Definition of Sentiment Polarity Classes

To utilize TCP, we convert continuous sentiment scores into discrete polarity classes. As the score ranges differ across datasets, the class boundaries are defined separately for SIMS and MOSI/MOSEI.

For the SIMS dataset, polarity classes are defined using fixed interval thresholds. Specifically, the 2-class setting divides the scores into $[-1.0, 0.0]$ and $(0.0, 1.0]$. The 3-class setting introduces a neutral region, resulting in $[-1.0, -0.1]$, $(-0.1, 0.1]$, and $(0.1, 1.0]$. The 5-class setting adopts a finer partition given by $[-1.0, -0.7]$, $(-0.7, -0.1]$, $(-0.1, 0.1]$, $(0.1, 0.7]$, and $(0.7, 1.0]$.

For the MOSI and MOSEI datasets, polarity classes are defined by mapping sentiment scores into discrete levels. The 2-class setting separates negative (< 0) and non-negative (≥ 0) samples. The 3-class, 5-class, and 7-class settings correspond to $\{-1, 0, 1\}$, $\{-2, -1, 0, 1, 2\}$, and $\{-3, -2, -1, 0, 1, 2, 3\}$, respectively.

B.2 Optimal Choice of Class Granularity

We conduct comparative experiments with $k \in \{2, 3, 5, 7\}$ (MOSI/MOSEI) and $k \in \{2, 3, 5\}$ (SIMS) and select k = 3 (MOSI/MOSEI) and k = 2 (SIMS) based on validation performance. As shown in Table 8, this configuration also yields the best test-set performance.

C Gradient Conflict Analysis

In multi-task learning, jointly optimizing multiple losses often leads to gradient conflicts, resulting in biased optimization where certain tasks dominate

Table 7: Network configurations of TCMR. Transformer-based modules list the number of layers, sequence lengths, token counts, input dimensions, attention heads, and hidden dimensions for both MOSI and MOSEI (shown as MOSI / MOSEI). MLP-based modules display the input dimensions and hidden-layer widths used in each component.

Module (MOSI / MOSEI)	Type	Notation	#Layers	Seq. len	Token len	Input dim	#Heads	Hidden dim
Transformer-based modules								
BERT Encoder	Transformer	θ^{bert}	12	50 / 50	50	768 / 768	12	768
Text Encoder	Transformer	θ_t^{trans}	2	49 / 49	8	768 / 768	8	128
Audio Encoder	Transformer	θ_a^{trans}	2	375 / 500	8	5 / 74	8	128
Vision Encoder	Transformer	θ_v^{trans}	2	500 / 500	8	20 / 35	8	128
A2T Generator (audio → text)	Transformer	θ_a^{gen}	2	16 / 16	–	128 / 128	8	128
V2T Generator (vision → text)	Transformer	θ_v^{gen}	2	16 / 16	–	128 / 128	8	128
Text Refinement Transformer	Transformer	θ_t^{ref}	2	8 / 8	–	128 / 128	8	128
CrossTransformer (fusion head)	Transformer	θ^{cross}	2	8 / 8	–	128 / 128	8	128
Reconstructor (Audio)	Transformer	θ_a^{recon}	2	8 / 8	–	128 / 128	8	128
Reconstructor (Vision)	Transformer	θ_v^{recon}	2	8 / 8	–	128 / 128	8	128
MLP-based modules								
Completeness Estimator (CompNet)	MLP	θ^{comp}	6	–	–	768 / 768	–	[768, 768, 1536, 768, 384, 1]
Polarity Classifier (Classifier)	MLP	$\theta_{\text{pre}}^{\text{classifier}}$	2	–	–	768 / 768	–	[384, 3]
Gating Network (GateNet)	MLP	θ^{gate}	2	–	–	256 / 256	–	[128, 1]
Text Regressor (Regressor)	MLP	$\theta^{\text{regressor}}$	1	–	–	128 / 128	–	[1]

Table 8: Performance sensitivity to sentiment class granularity (k) in TPSC label generation.

(a) MOSI dataset

Model	Acc-7	Acc-5	Non0 Acc/F1	Has0 Acc/F1	MAE	Corr
TPSC-2cls	31.45	36.01	70.01 / 69.91	69.46 / 69.25	1.1073	50.66
TPSC-3cls	**32.80**	**36.73**	**72.01 / 71.67**	**70.93 / 70.49**	**1.0721**	**52.28**
TPSC-5cls	30.96	35.12	69.51 / 69.09	68.93 / 68.42	1.1449	49.62
TPSC-7cls	32.16	35.35	70.85 / 70.76	69.74 / 69.54	1.1040	49.28

(b) MOSEI dataset

Model	Acc-7	Acc-5	Non0 Acc/F1	Has0 Acc/F1	MAE	Corr
TPSC-2cls	46.41	47.25	77.76 / 77.32	76.31 / 76.53	0.6677	57.95
TPSC-3cls	**47.27**	**48.16**	**77.99** / 76.99	**77.59 / 77.21**	**0.6614**	58.63
TPSC-5cls	46.94	47.96	77.79 / **77.56**	75.46 / 75.95	0.6638	**59.28**
TPSC-7cls	46.75	47.63	77.74 / 76.74	77.15 / 76.79	0.6667	57.81

(c) SIMS dataset

Model	Acc-5	Acc-3	Acc-2	F1	MAE	Corr
TPSC-2cls	31.15	**57.45**	**72.81**	68.88	**0.5212**	**0.379**
TPSC-3cls	32.43	56.23	71.72	**68.96**	0.5478	0.321
TPSC-5cls	**33.14**	55.45	69.13	68.06	0.5465	0.329

while others are under-optimized. To address this problem, prior work (Chen et al., 2018; Yu et al., 2020a) has proposed methods that mitigate gradient conflicts by adjusting gradient magnitudes or directions during training. For example, Chen et al. (2018) propose GradNorm, a method that automatically balances multiple tasks during training by adjusting the magnitudes of gradients derived from each task-specific loss, ensuring that all tasks learn at a similar rate. Yu et al. (2020a) propose PCGrad, which reduces gradient interference by projecting conflicting gradients onto the normal plane of other task gradients.

To validate the effectiveness of AOS, we conduct

Table 9: Gradient conflict rate (GCR) analysis measured during training. F50/L50 denote the first and last 50% of training epochs, respectively.

Method	GCR (%) ↓			$\mathcal{L}_{comp}$ ↓
	All	F50	L50	
E2E	51.27	51.16	54.36	0.1634
GradNorm	54.84	56.55	53.16	0.1294
PCGrad	52.33	57.43	48.70	0.0340
AOS	48.47	53.67	43.33	0.0055

comparative experiments using existing methods as baselines. Table 9 presents the gradient conflict rates during training and the corresponding convergence behavior of the completeness estimation loss, following the protocol of prior work (Zhang et al., 2024b). In joint training, $\mathcal{L}_{\text{comp}}$ tends to converge to a local optimum, which leads to an increase in GCR (see E2E in Table 9). While GradNorm and PCGrad mitigate gradient conflict, they are still insufficient to escape local optima due to the fundamental limitation of joint training. In contrast, AOS removes interference from other losses during the `CompNet` training phase, thereby effectively mitigating suboptimal convergence of $\mathcal{L}_{\text{comp}}$.

D Analysis of Completeness Labels

D.1 Lexicon-derived Completeness Label

To further validate the effectiveness of the proposed TPSC, we introduce *Lexicon-Derived Semantic Completeness* (LDSC) as a comparison baseline. LDSC measures completeness by explicitly quan-

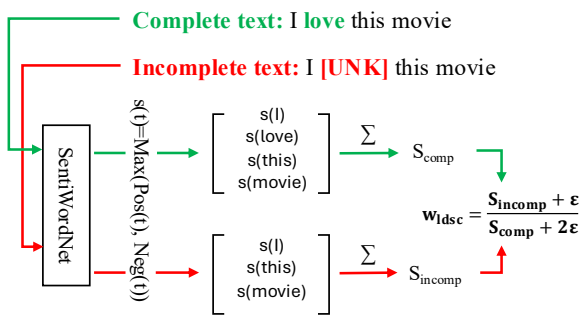
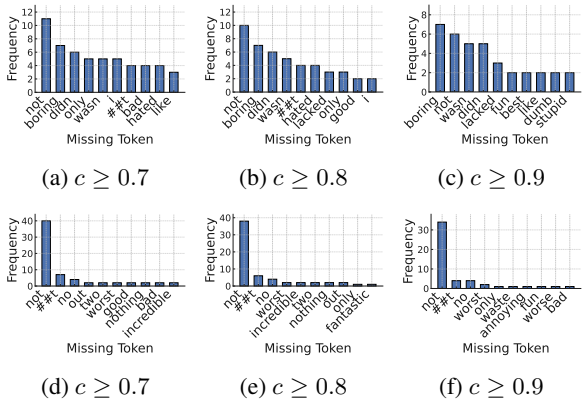


Figure 7: Illustration of LDSC computation



(a) $c \geq 0.7$ (b) $c \geq 0.8$ (c) $c \geq 0.9$

(d) $c \geq 0.7$ (e) $c \geq 0.8$ (f) $c \geq 0.9$

Figure 8: Token-level sensitivity analysis across different confidence thresholds. Subfigures (a)–(c) present results on the MOSI dataset, while (d)–(f) show the corresponding results on the MOSEI dataset.

tifying how much sentiment-related lexical information remains in the incomplete text. Specifically, we adopt SentiWordNet (Baccianella et al., 2010), a widely used lexical resource for sentiment analysis. SentiWordNet provides positive and negative sentiment scores for each word.

Figure 7 illustrates the overall process of generating the LDSC label. First, we define the sentiment intensity score of a token t as:

$$s(t) = \max\big(\mathrm{pos}(t), \mathrm{neg}(t)\big), \tag{15}$$

where $\mathrm{pos}(t)$ and $\mathrm{neg}(t)$ denote the positive and negative sentiment scores of the token. Using this token-level score, we compute the total sentiment-related content of the complete and incomplete text:

$$S_{\text{comp}} = \sum_{t \in T_{\text{comp}}} s(t), \tag{16}$$

$$S_{\text{incomp}} = \sum_{t \in T_{\text{incomp}}} s(t), \tag{17}$$

where T_{comp} and T_{incomp} denote the token sets of the complete and incomplete text, respectively. Finally, LDSC-based completeness is computed as:

$$w_{ldsc} = \frac{S_{\text{incomp}} + \varepsilon}{S_{\text{comp}} + 2\varepsilon}, \tag{18}$$

where ε is a small smoothing constant introduced to avoid numerical instability and degenerate cases in which both S_{comp} and S_{incomp} become zero. Intuitively, when $w_{ldsc} \approx 0$, most key sentiment tokens are missing, whereas $w_{ldsc} \approx 1$ indicates that the sentiment-related information is largely preserved.

D.2 Sentiment Token Sensitivity of TPSC

This subsection provides a more detailed description of the sentiment token sensitivity analysis introduced in Section 6.

Analysis Procedure. To evaluate the sensitivity of pseudo completeness labels to key sentiment tokens, we conduct an analysis following a four-step procedure:

- **Step 1: Correct-sample filtering.** A pre-trained sentiment polarity classifier is applied to the complete textual inputs, and only the correctly classified samples are selected for subsequent steps. Note that a sample is included only when the true-class probability $c \in [0, 1]$ exceeds a predefined threshold.
- **Step 2: Token-wise Augmentation via Masking.** For each correctly classified sample, we generate augmented samples by sliding a window over the sequence and removing exactly one token at each position, replacing it with `[UNK]`. For example, the sentence *"I love this movie"* produces four augmented samples:

  (1) `[UNK]` love this movie.
  (2) I `[UNK]` this movie.
  (3) I love `[UNK]` movie.
  (4) I love this `[UNK]`.

- **Step 3: Re-evaluation of augmented samples.** Then, each augmented sample is passed through the same classifier used in Step 1, and we collect the samples in which masking a specific token causes the predicted sentiment to flip to the opposite polarity (e.g., positive $\rightarrow$ negative or negative $\rightarrow$ positive).
- **Step 4: Identification of error-triggering tokens.** Using the polarity-flip samples collected in Step 3, we examine which token in

each flipped sample triggers misclassification and compute how frequently each token leads to a misprediction.

Analysis Results Figure 8 illustrates which tokens trigger polarity flips when masked under different thresholds. In both datasets, we consistently observed that as the threshold increases key sentiment tokens remain at the top while semantically ambiguous tokens naturally disappear. In MOSI, [*'I'*] is a sentiment-neutral token. One possible explanation is that masking it creates an ambiguous context before the verb, which may resemble negation patterns (e.g., "don't") learned during training and occasionally lead to prediction flips. In MOSEI, the token [*'two'*] is also observed among the flip-triggering words. This is because in the original sentence *"I give the movie two out of five stars"* the token [*'two'*] serves as an explicit negative cue. Overall, the consistent emergence of key sentiment tokens at higher confidence thresholds suggests that the text classifier is well calibrated with respect to semantic cues.

D.3 Upper-Bound Analysis

Table 10: Comparison between TCMR and its upper-bound variant TCMR-ub.

Method	Acc-5	Acc-2	MAE	Corr
MOSI Dataset				
TCMR	36.73	70.93	1.0721	52.28
TCMR-ub	41.24	77.57	0.9305	65.03
MOSEI Dataset				
TCMR	48.16	77.59	0.6614	58.63
TCMR-ub	58.72	79.83	0.5697	71.78
SIMS Dataset				
TCMR	31.15	72.81	0.5212	37.91
TCMR-ub	40.39	73.69	0.3991	66.47

To validate the effectiveness of the pseudo-labels, we compare the performance of TCMR-ub, trained with target TPSC labels. Specifically, TCMR-ub is trained by directly using the pseudo-labels instead of training the confidence estimator, enabling us to assess the upper-bound performance of our framework. As shown in Table 10, TCMR-ub consistently outperforms TCMR across all metrics. This indicates that the pseudo-labels provide a meaningful and effective signal for guiding the reconstruction process.

D.4 Failure Cases of TPSC

Dataset	Accuracy (%)	Error Rate (%)
MOSI	74.92	25.08
MOSEI	66.98	33.02
SIMS	79.07	20.93

Table 11: Accuracy of the pretrained sentiment polarity classifier used for TPSC pseudo-labeling on each dataset's complete text. The higher error rate indicates a greater likelihood of noisy pseudo-labels.

Since TPSC is based on pseudo-labeling, incorrect completeness labels may be generated when the classifier misclassifies samples even on complete data. This is a fundamental limitation of pseudo-label-based approaches that do not rely on human annotations. To assess the prevalence of such failure cases, we evaluated the performance of the pretrained classifier on each dataset. Table 11 shows that MOSEI exhibits relatively lower accuracy compared to other datasets, likely due to its higher complexity. This suggests that the noisier pseudo-labels in MOSEI can partially explain its relatively weaker generalization performance.

We further analyzed failure cases and observed that misclassifications mainly occur in ambiguous and sarcastic expressions, as shown in Table 12. These cases are inherently challenging to predict using only the text modality. Despite these limitations, the impact of such mispredictions on completeness estimation is limited. Due to their low TCP, TPSC assigns low completeness scores to misclassified samples, preventing overestimation of completeness and encouraging reliance on proxy features.

E Analyses with competitive baselines

E.1 Metric-Level Comparison

In MSA tasks, both classification and regression performances are evaluated to provide a comprehensive assessment. However, classification metrics may not fully represent the quality of sentiment prediction in certain cases due to discretization in their computation.

Specifically, this discretization process discards fine-grained differences between predictions within the same rounding interval. For example, when computing the classification metrics (e.g., Acc-7, Acc-5), continuous sentiment predictions are first converted into discrete classes via rounding, which

Pattern	Utterance	GT	Pred	TCP
Ambiguous	There are some funny moments	Neutral	Positive	**0.038**
	Um I did enjoy it	Neutral	Positive	**0.027**
Sarcasm	Hi I'm pretty I have a giant smile I'm supposed to know things um walk of screen	Negative	Positive	**0.131**
	He um had all the charm of a narcissist xxx boy the whole film	Negative	Positive	**0.022**

Table 12: Representative failure cases of the pretrained classifier in TPSC pseudo-labeling.

Dataset	TCMR vs	Acc-7	Acc-5	Non0 A/F	Has0 A/F	MAE	Corr	W-L
MOSI	CENet	+3.31	+3.83	+2.13 / +1.73	+1.50 / +1.10	+0.108	+4.04	**8-0**
	P-RMF	+3.89	+5.49	+2.10 / +2.10	+1.65 / +1.65	+0.058	+2.21	**8-0**
MOSEI	CENet	−0.09	−0.08	+0.98 / −0.27	+1.63 / +0.98	+0.001	+0.39	**5-3**
	P-RMF	+0.26	+0.34	−0.08 / −0.77	+0.41 / +0.30	+0.005	+0.18	**6-2**

Dataset	TCMR vs	Acc-5	Acc-3	Acc-2	F1	MAE	Corr	W-L
SIMS	CENet	+9.54	+3.97	+3.58	+11.91	+0.109	+0.365	**6-0**
	P-RMF	−1.57	+0.15	+2.23	−1.48	+0.022	+0.018	**4-2**

Table 13: Detailed comparison between TCMR-TPSC and CENet/P-RMF on MOSI, MOSEI, and SIMS.

may lead to inconsistencies near class boundaries, as follows:

- GT: $1.4 \rightarrow \text{round}(1.4) \rightarrow$ class 1
- Model A: 1.55 (MAE $= 0.15$) $\rightarrow$ class 2
- Model B: 1.10 (MAE $= 0.30$) $\rightarrow$ class 1

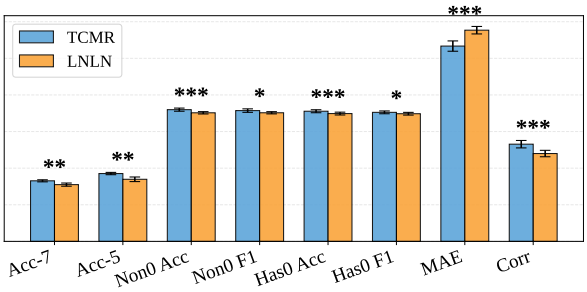
As shown in Table 13, except for a few classification metrics, TCMR-TPSC consistently outperforms CENet and P-RMF across all benchmark datasets. This indicates that TCMR provides more reliable sentiment prediction overall.

E.2 Training Efficiency Analysis

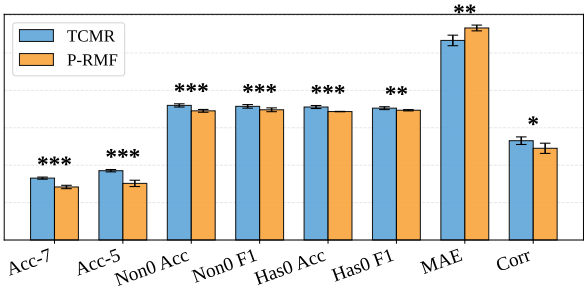
Table 14: Comparison of model complexity and training time per epoch.

Model	# Params (M)	Time / Epoch (s)
LNLN	115.97	15.40
P-RMF	117.31	18.00
TF-Mamba	111.30	11.12
TCMR	119.23	12.48

As shown in Table 14, TCMR achieves the second-fastest training speed, following TF-Mamba. This is because the reconstruction modules used in LNLN and P-RMF involve more computationally intensive operations, whereas TCMR employs a lightweight MLP-based CompNet. As a result, the additional optimization steps introduced by AOS do not significantly increase the overall training cost.



(a) TCMR vs LNLN



(b) TCMR vs P-RMF

Figure 9: Error bars indicate standard deviations. Asterisks denote statistical significance (* $p < 0.1$, ** $p < 0.05$, *** $p < 0.01$).

E.3 Statistical Analysis

To examine the statistical significance of performance differences among models, we conduct comparative experiments on the MOSI dataset with LNLN, P-RMF, and TCMR. As shown in Figure 9, TCMR achieves statistically significant performance improvements across most evaluation metrics.