Multi-Perspective Feature Modeling with MLLMs for Multimodal Sarcasm Detection

Anonymous ACL submission

Abstract

Multimodal sarcasm detection (MSD) models often overfit to in-domain data due to a lack of proper understanding of the data, which contains slang or memes in text and includes overly incomprehensible images. And existing methods only focus on inconsistencies in the data while ignoring the diversity of sarcastic expressions. To address this, we propose a novel method which is named as **M**ulti-**P**erspective feature modeling for **M**ultimodal **S**arcasm **D**etection (MPMSD). Specifically, we first use multimodal large language models (MLLMs) to generate relevant knowledge to enhance the understanding of the data. Then, based on the generated data and the original data, MPMSD models diverse types of sarcasm from three perspectives (Knowledge Learning, Incongruity Mining, and Representation Enhancement). Experiments demonstrate that our approach not only outperforms the state-of-the-art (SOTA) but also exhibits strong generalization ability and robust noise resistance.

1 Introduction

Sarcasm is a pervasive linguistic phenomenon in human society, serving as a crucial way to express emotions. With the rapid development of social media, many people opt to convey complex semantics and emotions using text-image combinations, which often contain sarcasm. Consequently, multimodal sarcasm detection plays an essential role in improving the effectiveness of sentiment analysis, question answer systems, and opinion mining.

Previous work (Joshi et al., 2015) demonstrates that inconsistency is a critical factor in sarcasm detection, laying the foundation for extensive subsequent research on inconsistency using textual and visual cues. Many studies model commonalities and inconsistencies between images and text, employing attention-based methods (Wang et al., 2020), graph-based methods (Liang et al., 2021, 2022; Guo et al., 2024), and optimal transport (Prasanna et al., 2023). Studies Li et al. (2021) have emphasized the importance of incorporating extensive world knowledge. Consequently, some studies incorporate object detection methods (Qiao et al., 2023; Hao et al., 2024; Guo et al., 2024), commonsense knowledge graphs (Qiao et al., 2023; Hao et al., 2024), sentiment knowledge graphs (Guo et al., 2024; Liang et al., 2022), and more.

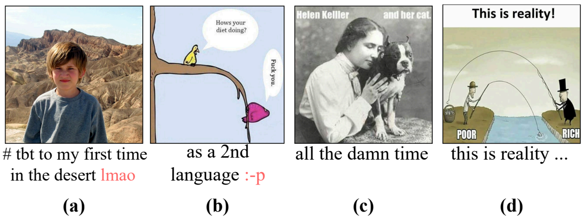

Figure 1: 4 examples of mutlimodel sarcasm. The text in (a) and (b) contains slang and memes, while the images in (c) and (d) are incomprehensible to traditional methods.

Despite these efforts, we observe two challenges in this field: **(1) Insufficient understanding of data.** As illustrated in Figure 1 (a) and (b), "lmao" stands for "laughing my ass off" while ":-p" is an emoticon mimicking a tongue-sticking-out expression, typically conveying a joking tone. Both are critical evidence in sarcasm detection. The incongruity between the dog and the label "cat" in (c), along with the implication that the wealth of the rich derives from the poor in (d), constitute the sarcasm exemplified. These images' semantics are difficult to be understood by traditional methods. **(2) A single model struggles to adapt to diverse sarcastic expressions.** Prior work has primarily focused on inconsistency, while we find that sarcastic expressions are diverse. Inspired by Leggitt and Gibbs (2000), we categorize sarcasm into six types (Inconsistency, Joke, Implicit Irony, Understatement, Overstatement, and Rhetorical Question), as illustrated in Figure 2. A single model is inadequate for detecting such diverse data.

To tackle the above issues, we introduce MPMSD, a novel multi-perspective feature modeling method. First, we prompt MLLMs to generate information about text and image as external knowledge, thereby mitigating the challenge **(1)**. Based on generated and original data, we design two innovative submodels (Knowledge Learning and Representation Enhancement) using the cross-attention and mask contrastive learning, which are respectively used to learn useful information and enhance data representation. Additionally, we identify an issue, misalignment between nodes and adjacency matrix, in prior work that utilized graph neural networks (GNN) for incongruity in the data, and redesign it for MPMSD. We employ these three submodels to capture diverse sarcastic features from distinct perspectives, thereby addressing the challenge **(2)**. Here, I attempt to explain why multiple submodels can better addressing the challenge **(2)**. As shown in Figure 2, for inconsistency among 6 types, the key to detecting sarcasm lies in linking the 'delicious' in the text with the 'bad food' in the image. This may only require token-level feature modeling, so the submodel (Incongruity Mining) using GNN performs better for such data. For another example, joke, both the text and the image contain 'LOL,' which stands for 'Laugh Out Loud.' This is key to identifying sarcasm, but recognizing this requires external knowledge, so the submodel (Knowledge Learning) performs better for such data. Different feature modeling approaches do affect the final prediction for different types of sarcasm. Our contributions are as follows:

- We are the first to point out that poor performance is related to insufficient data understanding. Therefore, our method combines the generated data of MLLMs for the first time.
- We notice five other types of sarcasm, which are not focused on in previous work. And we find that different submodels have varying abilities to extract features for different types.
- We propose two innovative submodels and redesign a problematic submodel, and these submodels are used to jointly model features from distinct perspectives.
- Experiments show that MPMSD significantly outperforms state-of-the-art methods, exhibiting strong generalization ability and robust noise resistance.

Figure 2: 6 types of sarcasm. The samples in the yellow region refer to types previously unnoticed by researchers. **Implicit Irony** refers to a surface expression of support that actually conveys opposition or sarcasm.

2 Related Work

Multimodal Sarcasm detection. Schifanella et al. (2016) initially investigated this task by combining textual and visual features. Subsequently, Cai et al. (2019) developed a hierarchical fusion model to integrate information across different modalities and introduced a new public dataset, MMSD. Later studies explored modeling inconsistencies between text and images using a decomposition and relation network (Xu et al., 2020), BERT-based models with modified attention mechanisms (Pan et al., 2020a; Wang et al., 2020), and graph neural networks (Liang et al., 2021, 2022; Qiao et al., 2023; Guo et al., 2024; Zhang et al., 2025).

Knowledge Enhanced Sarcasm Detection. Xu et al. (2020) focused on identifying adjective-noun pairs (ANPs) from visual data to capture inconsistencies between textual content and visual cues. Subsequently, studies by Liang et al. (2022), Qiao et al. (2023), and Hao et al. (2024) employed object detection frameworks, leveraging the semantic labels of detected entities to narrow the gap between modalities. In particular, both Qiao et al. (2023) and Hao et al. (2024) made use of ConceptNet5 to explore the semantic connections between visual and textual elements. Furthermore, Liang et al. (2022), Liu et al. (2022), and Guo et al. (2024) incorporated SenticNet (Cambria et al., 2020) to analyze sentiment polarity in text as an additional signal for sarcasm detection. While most prior work (Guo et al., 2024; Jia et al., 2024) leveraged LLMs primarily for data augmentation or mit-

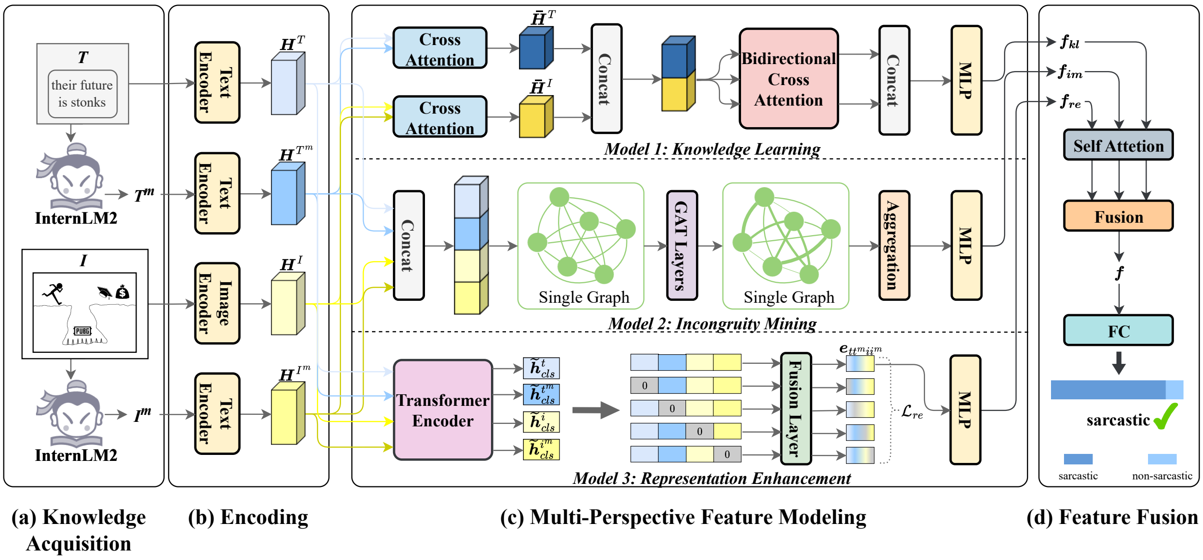

Figure 3: The overall architecture of MPMSD primarily comprises 4 modules: (a) Knowledge Acquisition, (b) Encoding, (c) Multi-Perspective Feature Modeling, and (d) Feature Fusion.

igating modality bias. In our approach, we employ a multimodal large language model to extract information such as slang and popular memes from text and enhance the understanding of images.

3 Methodology

Task Definition. Given an image-text pair $<I_i, T_i>$ where I_i is the i-th image input and T_i is the i-th text input, the task's objective is to determine whether the image-text pair implies any sarcasm. For simplicity, we temporarily omit the subscript i in the following section.

As illustrated in Figure 3, the architecture of our model is primarily composed of 4 modules: Knowledge Acquisition, Encoding, Multi-Perspective Feature Modeling, and Feature Fusion. Next, we provide a detailed description of these modules.

3.1 Knowledge Acquisition

There are a large amount of internet slang and memes in text or images, some of which require substantial world knowledge for accurate semantic interpretation. Therefore, we employ MLLMs to extract knowledge from both textual and visual modalities. Specifically, we respectively use two templates ($Question1$, 2) to acquire knowledge from InternLM2-vl-7b (Dong et al., 2024), and then obtain T^m and I^m:

> *Question1*: Please understand the meaning of the text and try to be concise while ensuring the correctness of your answers.
> Text: ***T***

> *Question2*: Please describe the content in the image and try to be concise while ensuring the correctness of your answers.
> Image: ***I***

3.2 Encoding

We adopt CLIP (Radford et al., 2021) as our encoder, as it encodes textual and visual data into a shared semantic space, which is beneficial for subsequent feature modeling.

CLIP contains a text encoder $\mathbb{T}$ and a visual encoder $\mathbb{V}$. For $\boldsymbol{T}$ and $\boldsymbol{I}$, their representations $\boldsymbol{H}^T$ and $\boldsymbol{H}^I$ are output through $\mathbb{T}$ and $\mathbb{V}$:

$$\boldsymbol{H}^T = (\boldsymbol{h}_1^t, \dots, \boldsymbol{h}_n^t, \boldsymbol{h}_{cls}^t) = \mathbb{T}(\boldsymbol{T}), \qquad (1)$$

$$\boldsymbol{H}^I = (\boldsymbol{h}_{cls}^i, \boldsymbol{h}_1^i, \dots, \boldsymbol{h}_m^i) = \mathbb{V}(\boldsymbol{I}). \qquad (2)$$

For generated textual modal data T^m and I^m, we obtain their representations $\boldsymbol{H}^{T^m}$ and $\boldsymbol{H}^{I^m}$ by Eq. (1). Unlike all previous work, the parameters of CLIP are frozen during training, and the reason will be discussed in **Appendix A**.

3.3 Multi-Perspective Feature Modeling

Sarcasm can be expressed in various ways, and previous approaches with a single model to capture diverse features is no longer appropriate. Therefore, we employ three various submodels to model semantic features from different perspectives.

3.3.1 Model 1: Knowledge Learning

In the submodel, we aim for the original data T and I to learn useful information from knowledge-rich T^m and I^m. Concretely, we apply the cross-attention mechanism to query knowledge from generated sequence $\boldsymbol{H}^{T^m}$ using the original data representation sequence $\boldsymbol{H}^T$ as $\boldsymbol{Q}$:

$$\boldsymbol{Q^T} = \boldsymbol{H}^T, \boldsymbol{K^T} = \boldsymbol{H}^{T^m}, \boldsymbol{V^T} = \boldsymbol{H}^{T^m}, \quad (3)$$

$$\bar{\boldsymbol{H}}^T = \text{Cross\_att}(\boldsymbol{Q^T}, \boldsymbol{K^T}, \boldsymbol{V^T}). \quad (4)$$

Similarly, we obtain $\bar{\boldsymbol{H}}^I$ by $\boldsymbol{H}^I$ and $\boldsymbol{H}^{I^m}$.

Then, we design a bidirectional cross-attention mechanism for text and image representations to learn from each other. Specially, it concatenates $\bar{\boldsymbol{H}}^I$ with $\bar{\boldsymbol{H}}^T$ to form $\boldsymbol{Q}$, $\boldsymbol{K}$and $\boldsymbol{V}$, and then feeds them into a cross-attention layer (Eq. (4)) to obtain $\hat{\boldsymbol{F}} = (\hat{\boldsymbol{h}}^i_{\text{cls}}, \hat{\boldsymbol{h}}^i_1, \ldots, \hat{\boldsymbol{h}}^i_m, \hat{\boldsymbol{h}}^t_1, \ldots, \hat{\boldsymbol{h}}^t_n, \hat{\boldsymbol{h}}^t_{\text{cls}})$. The two [CLS] tokens are concatenated to obtain the final interaction feature $\hat{\boldsymbol{f}}$ of **Model 1**.

To ensure that the final output features of submodels have the same dimensionality, we apply a Multilayer Perceptron (MLP) for dimensional transformation:

$$\boldsymbol{f_{kl}} = \text{MLP}(\hat{\boldsymbol{f}}). \quad (5)$$

3.3.2 Model 2: Incongruity Mining

Text: incredible weather today

Spacy: incredible weather today

Adjacency Matrix: $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$

Tokenization:["inc", "redible", "weather", "today"]

Figure 4: Misalignment between Nodes and Adjacency Matrix.

In the submodel, as previous works (Guo et al., 2024; Hao et al., 2024; Qiao et al., 2023), we use graph neural networks (GNN) to mine incongruity. However, previous works had two issues: 1) Constructing three graphs of text, image, and cross-model aggravates overfitting on in-domain data, and 2) A more fundamental issue is that the constructed adjacency matrix and node representation are not aligned. In these approaches, when constructing the text graph, the initial node representations are derived from token representations output by a text encoder. The adjacency matrix for nodes is generated by extracting dependencies between words in the text using SpaCy[1], as illustrated in Figure 4. However, since the encoder does not necessarily tokenize the text based on words, this can result in a misalignment between nodes and the adjacency matrix.

[1]https://spacy.io/

To address the above issues, we construct a single Graph Attention Network (GAT) (Veličković et al., 2018), and initialize the node representation as $\boldsymbol{G}^0 = \{\boldsymbol{H}^T, \boldsymbol{H}^{T^m}, \boldsymbol{H}^I, \boldsymbol{H}^{I^m}\} = \{\boldsymbol{g}^0_i\}$. Since all nodes are encoded by CLIP in the same semantic space, we directly compute the cosine similarity as edges between nodes. Let $\alpha^l_{i,j}$ be the attention score between i and j, and $\boldsymbol{g}^l_i$ denote the feature of node i in the l-th layer:

$$\alpha^l_{i,j} = \frac{\exp\left(\text{LeakyReLU}\left(\boldsymbol{u}_l[\boldsymbol{W}_l\boldsymbol{g}^l_i \| \boldsymbol{W}_l\boldsymbol{g}^l_j]\right)\right)}{\sum_{k\in\mathcal{N}_i}\exp\left(\text{LeakyReLU}\left(\boldsymbol{u}_l[\boldsymbol{W}_l\boldsymbol{g}^l_i \| \boldsymbol{W}_l\boldsymbol{g}^l_k]\right)\right)}, \quad (6)$$

$$\boldsymbol{g}^{l+1}_i = \alpha^l_{i,i}\boldsymbol{W}_l\boldsymbol{g}^l_i + \sum_{j\in\mathcal{N}_i}\left(\alpha_{i,j}\boldsymbol{W}_l\boldsymbol{g}^l_j\right), \quad (7)$$

where $\mathcal{N}_i$ is neighbor nodes of i. $\|$ denotes concatenation operation. $\boldsymbol{W}_l$ and $\boldsymbol{u}_l$ are learnable parameters. The final representation is $\boldsymbol{G} = \{\boldsymbol{g}_0, ..., \boldsymbol{g}_{|\boldsymbol{G}|}\}$. We obtain the average of node features $\hat{\boldsymbol{g}}$, as the output of **Model 2**. Similar to Eq.5, $\hat{\boldsymbol{g}}$ undergoes a dimensional transformation:

$$\hat{\boldsymbol{g}} = \frac{1}{|\boldsymbol{G}|}\sum_{\boldsymbol{g}_i\in\boldsymbol{G}}(\boldsymbol{g}_i), \quad (8)$$

$$\boldsymbol{f_{im}} = \text{MLP}(\hat{\boldsymbol{g}}). \quad (9)$$

3.3.3 Model 3: Representation Enhancement

In the submodel, we design an unsupervised contrastive learning to enhance data representation. Since CLIP are frozen, we employ a transformer encoder (Vaswani, 2017) to update $\boldsymbol{H^T}$ and obtain new representation $\widetilde{\boldsymbol{H}}^{\boldsymbol{T}}$:

$$\widetilde{\boldsymbol{H}}^{\boldsymbol{T}} = \text{Transformer}(\boldsymbol{H^T}). \quad (10)$$

Similarly, $\widetilde{\boldsymbol{H}}^{\boldsymbol{T^m}}$, $\widetilde{\boldsymbol{H}}^{\boldsymbol{I}}$, and $\widetilde{\boldsymbol{H}}^{\boldsymbol{I^m}}$ are obtained using the same encoder. **Model 3**'s output is $\boldsymbol{e}_{tt^m ii^m}$ by [CLS] tokens of four new representation:

$$\boldsymbol{e}_{tt^m ii^m} = \mathcal{F}(\text{Concat}(\widetilde{\boldsymbol{h}}^t_{cls}, \widetilde{\boldsymbol{h}}^{t^m}_{cls}, \widetilde{\boldsymbol{h}}^i_{cls}, \widetilde{\boldsymbol{h}}^{i^m}_{cls})), \quad (11)$$

$$\boldsymbol{f_{re}} = \text{MLP}(\boldsymbol{e}_{tt^m ii^m}), \quad (12)$$

where $\mathcal{F}$ is a fusion layer which is defined as $\boldsymbol{W}\sigma_{\text{GELU}}(\text{Dropout}(\cdot)) + \boldsymbol{b}$.

We mask one [CLS] token in Eq.11, and this token is replaced with zero vectors. In this way, $\boldsymbol{e}_{0t^m ii^m}$, $\boldsymbol{e}_{t0ii^m}$, $\boldsymbol{e}_{tt^m 0i^m}$ and $\boldsymbol{e}_{tt^m i0}$ are obtained as positively augmented samples. For each positive pair (n,m) in a minibatch during training, we apply the unsupervised contrastive learning loss:

$$\mathcal{L}_{re}^{n,m} = -\log\left(\frac{\exp(sim(\phi(\boldsymbol{e}^n),\phi(\boldsymbol{e}^m))/\tau)}{\sum_k \mathbb{I}_{[k\neq n]}\exp(sim(\phi(\boldsymbol{e}^n),\phi(\boldsymbol{e}^k))/\tau)}\right), \tag{13}$$

where $\boldsymbol{e}^n \in \{\boldsymbol{e}^n_{tt^m ii^m}, \boldsymbol{e}^n_{0t^m ii^m}, \boldsymbol{e}^n_{t0ii^m}, \boldsymbol{e}^n_{tt^m 0i^m}, \boldsymbol{e}^n_{tt^m i0}\}$, n refers to the n-th sample in the mini-batch, $sim(\cdot)$ denotes to the dot product operation, and $\phi(\cdot)$ is a non-linear layer with ReLU, serving as contrastive head. τ represents the temperature, and $\mathbb{I}_{[\cdot]}$ is a indicator function.

Through this masking approach, the distance between positive samples is reduced, compelling the model to learn shared semantic features among T, I, T^m, and I^m, thereby enhancing representations and improving the sarcasm detection performance.

3.4 Feature Fusion

To enable the model to recognize which features are more suitable for the current data instance, we design a Feature Fusion Layer that utilizes a self-attention mechanism. It can adaptively select more important features and fuse them according to their significance.

Specifically, taking $\boldsymbol{f_{kl}}$ as an example, we apply three linear layers to obtain $\boldsymbol{f^q_{kl}}$, $\boldsymbol{f^k_{kl}}$ and $\boldsymbol{f^v_{kl}}$, and then compute the attention weights $\boldsymbol{p_{kl}}$:

$$\boldsymbol{p_{kl}} = \mathrm{softmax}\left(\frac{\boldsymbol{f^q_{kl}}(\boldsymbol{f^k_{kl}})^\top}{\sqrt{d_k}}\right), \tag{14}$$

where d_k is the embedding dimension. The three linear layers are shared among $\boldsymbol{f_{kl}}$, $\boldsymbol{f_{im}}$ and $\boldsymbol{f_{re}}$. Similarly, we can obtain $\boldsymbol{p_{im}}$ and $\boldsymbol{p_{re}}$. The final fused feature is $\boldsymbol{f}$ which is utilized for prediction:

$$\boldsymbol{f} = \boldsymbol{p_{kl}}\boldsymbol{f^v_{kl}} + \boldsymbol{p_{im}}\boldsymbol{f^v_{im}} + \boldsymbol{p_{re}}\boldsymbol{f^v_{re}}, \tag{15}$$

$$\boldsymbol{y} = \mathrm{softmax}(\boldsymbol{W}\boldsymbol{f} + \boldsymbol{b}), \tag{16}$$

where $\boldsymbol{y}$ is output distribution, while $\boldsymbol{W}$ and $\boldsymbol{b}$ are trainable parameters.

3.5 Model Training

We use a standard binary cross-entropy loss for MSD task:

$$\mathcal{L}_{ce} = -(\hat{\boldsymbol{y}}\log\boldsymbol{y} + (1-\hat{\boldsymbol{y}})\log(1-\boldsymbol{y})), \tag{17}$$

where $\hat{\boldsymbol{y}}$ is gold label. The final loss function is defined as the combination of the contrastive learning loss and cross-entropy loss:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda\mathcal{L}_{re}, \tag{18}$$

where λ is the hyperparameter.

MMSD MMSD2.0	Train	Valid	Test
Sentences	19816/19816	2410/2410	2409/2409
Positive	8642/9572	959/1042	959/1037
Negative	11174/10240	1451/1368	1450/1372

Table 1: Statistics of MMSD and MMSD2.0.

4 Experiments

4.1 Experiments Setup

Datasets. We conduct experiments on MMSD (Cai et al., 2019) and MMSD2.0 (Qin et al., 2023). Specifically, MMSD is derived from Twitter. Therefore, tweets with some special hashtags (e.g., sarcasm) are positive examples and those without such hashtags are negative examples. MMSD2.0 is updated from MMSD, which has removed misleading cues. The statistics about datasets are shown in Table 1. **Implementation Details.** We use Adam (Loshchilov, 2017) as the optimizer with a batch size of 32, 10 epochs, and a learning rate of 2e-5. The GAT in Model 2 has 3 layers, with an input dimension of 512, 4 attention heads, and hidden/output dimensions of 128. The transformer encoder in Model 3 follows a 3-layer standard architecture. The temperature τ is 0.2 and λ is 1. All experiments were run on an A100 (80 GB).

4.2 Baselines

We explore two types of baselines: (i) $MLLMs$ $methods$ and (ii) $Multimodal$ $methods$. (i) For $MLLMs$ $methods$, we conduct tests on (1) InternLM2-vl-7b (Dong et al., 2024) and (2) InternLM2-vl-7b with LoRA; (3) GPT-4o (OpenAI, 2024); (4) MMoE (Yu et al., 2024); (5) DemoRetrieval (Tang et al., 2024). (ii) For $Multimodal$ $methods$, (6) Att-BERT (Pan et al., 2020b); (7) InCrossMGs (Liang et al., 2021); (8) CMGCN (Liang et al., 2022); (9) HKE (Liu et al., 2022); (10) MILNet (Qiao et al., 2023); (11) DynRT-Net (Tian et al., 2023); (12) M-CLIP (Qin et al., 2023); (13) M-CLIP+TFCD (Zhu et al., 2024b); (14) DMSD-CL (Jia et al., 2024); (15) G^{2}SAM (Wei et al., 2024); (16) DMMD (Zhu et al., 2024a); (17) ITFNet (Zhang et al., 2025). More information is available in **Appendix B**.

4.3 Main Results

Our method leverages external knowledge, while some methods in $Multimodal$ $methods$ do not. To ensure a fair comparison, we also evaluate **MPMSD** w/o MLLM. Specifically, in Model 1,

Model	MMSD				MMSD2.0			
	Acc. (%)	P (%)	R (%)	F1 (%)	Acc. (%)	P (%)	R (%)	F1 (%)
MLLM methods								
InternLM2-vl-7b (Dong et al., 2024)	66.72	67.42	32.22	43.60	66.29	73.89	33.56	46.15
InternLM2-vl-7b + LoRA	**93.81**	90.35	94.16	92.22	86.05	81.16	88.04	84.46
GPT4o (OpenAI, 2024)	68.05	55.06	**94.92**	69.69	71.03	60.30	**95.11**	73.80
MMoE [Qwen2-0.5b] (ACL2024)	–	–	–	–	82.27	–	–	80.67
DemoRetrieval [LLaVA1.5-7b] (NAACL2024)	89.97	89.26	89.58	89.42	86.43	87.00	86.30	86.43
Multimodal methods								
Att-BERT (EMNLP2020b)	86.05	80.87	85.08	82.92	80.03	76.28	77.82	77.04
InCrossMGs (MM2021)	86.10	81.38	84.36	82.84	–	–	–	–
CMGCN (ACL2021)	86.54	–	–	82.73	79.83	75.82	78.01	76.90
HKE (EMNLP2022)	87.36	81.84	86.48	84.09	76.50	73.48	71.07	72.25
MILNet (AAAI2023)	89.50	85.16	89.16	87.11	–	–	–	–
DynRT-Net (ACL2023)	93.59	**93.06**	93.60	**93.31**	71.40	71.80	72.17	71.34
M-CLIP (ACL2023)	88.33	82.66	88.65	85.55	85.64	80.33	88.24	84.10
M-CLIP+TFCD (IJCAI2024b)	89.57	84.83	89.43	88.13	86.54	82.46	87.95	84.31
DMSD-CL (AAAI2024)	88.95	84.89	87.90	86.37	–	–	–	–
G^2SAM (AAAI2024)	90.48	87.95	89.02	88.48	–	–	–	–
DMMD (COLING2024a)	90.60	86.95	91.04	88.93	–	–	–	–
ITFNet (ACL2025)	92.04	90.21	90.30	90.25	86.73	81.08	88.03	84.41
MPMSD [InternLM2-vl-7b]	93.55	90.04	93.84	91.90	**91.70**	**88.57**	92.67	**90.58**
MPMSD w/o MLLM	91.87	86.75	93.41	89.95	90.49	85.94	93.15	89.40

Table 2: Experimental results on MMSD and MMSD2.0. The best results are highlighted in bold, the second-best results are underlined, and the third-best results are wavy-underlined. [ ] refers to MLLMs used.

$\boldsymbol{H}^T$ and $\boldsymbol{H}^I$ are directly used in the bidirectional cross-attention mechanism. In Model 2, we initialize the node representation as $\boldsymbol{G}^0 = \{\boldsymbol{H}^T, \boldsymbol{H}^I\} = \{\boldsymbol{g}_i^0\}$. In Model 3, the output becomes $\boldsymbol{e}_{ti} = \mathcal{F}(\mathrm{Concat}(\widetilde{\boldsymbol{h}}_{cls}^t, \widetilde{\boldsymbol{h}}_{cls}^i))$, and the contrastive learning is removed.

As shown in Table 2, in *MLLMs methods*, untrained InternLM2-vl-7b and GPT4o exhibit poor performance. This indicates that sarcasm detection requires task-specific training, and that current LLMs cannot be directly applied to sarcasm detection without fine-tuning. Notably, GPT-4o achieves a recall of around 95%, indicating its strong ability to capture nearly all instances of sarcasm. Overall, many models exhibit substantial performance gaps between the MMSD and MMSD2.0. The most pronounced gap is observed in DynRT-Net, which achieves 93.59% accuracy and 93.31% F1 score on the MMSD dataset, but only 71.40% accuracy and 71.43% F1 on MMSD2.0. We argue that strong performance on the lower-quality MMSD but poor on the higher-quality MMSD2.0 indicates that the model does not possess genuine sarcasm detection capabilities, but rather tends to overfit to in-domain data, resulting in limited generalization, which will be further discussed in the following parts.

For MMSD, MPMSD achieves the third-best performance, with slightly lower accuracy and F1 score compared to InternLM2-vl-7b + LoRA and DynRT-Net. Upon removing external knowledge, MPMSD w/o MLLM decreases by 1.68% in ACC and 1.95% in F1. Nevertheless, MPMSD w/o MLLM still outperforms all other baselines aside from InternLM2-vl-7b + LoRA and DynRT-Net. In the following Sec.(§4.4), we will demonstrate that the superior performance does not reflect stronger sarcasm detection capabilities than MPMSD but rather results from overfitting to spurious cues present in MMSD.

For MMSD2.0, none of the previous methods exceed 87% in both accuracy and F1 score. However, our method achieves a 4.97% increase in accuracy and a 4.15% increase in F1 score compared to the previous best result. Even without external knowledge, MPMSD w/o MLLM still achieves improvements of 3.76% in accuracy and 2.97% in F1 score. Since MMSD2.0 is an advanced version of MMSD, we believe that this clearly demonstrates the necessity of multi-perspective modeling for capturing sarcastic features.

Model	MMSD for RedEval				MMSD2.0 for RedEval			
	Acc. (%)	P (%)	R (%)	F1 (%)	Acc. (%)	P (%)	R (%)	F1 (%)
DynRT-Net (ACL2023)*	93.59	93.06	93.60	93.31	71.40	71.80	72.17	71.34
	58.57	47.06	49.25	41.35	74.80	75.58	76.69	74.66
M-CLIP (ACL2023)*	88.33	82.66	88.65	85.55	85.64	80.33	88.24	84.10
	76.29	75.67	73.70	74.30	80.98	80.85	82.62	80.73
InternLM2-vl-7b + LoRA	93.81	90.35	94.16	92.22	86.05	81.16	88.04	84.46
	61.25	53.33	12.15	19.79	86.75	80.75	87.09	83.80
DemoRetrieval (NAACL2024)*	89.97	89.26	89.58	89.42	86.43	87.00	86.30	86.43
	59.16	49.70	48.67	41.47	83.47	83.12	82.30	82.83
MPMSD	93.55	90.04	93.84	91.90	91.70	88.57	92.67	90.58
	92.13	**95.14**	**84.30**	**89.40**	**92.83**	**84.58**	**100.00**	**91.65**
MPMSD w/o MLLM	91.87	86.75	93.41	89.95	90.49	85.94	93.15	89.40
	86.55	**86.11**	**78.48**	**82.12**	**90.87**	**85.23**	**95.27**	**89.69**

Table 3: The values of the light-colored font are from Table 2, indicating the results tested on the test set of MMSD and MMSD2.0. The values with a yellow background are the results of the trained model tested on RedEval, representing the model's generalization performance. $*$ denotes the experimental results from (Tang et al., 2024).

4.4 Generalization Ability

Previous works (Qin et al., 2023; Tang et al., 2024; Farabi et al., 2024) showed that many models rely on spurious cues and overfit on specific in-domain data. To evaluate the generalization, models are tested on an out-of-domain dataset, RedEval.

RedEval was proposed by (Tang et al., 2024) with the aim of evaluating the generalization of existing models. More detailed information about RedEval can be found in **Appendix D**. We select baselines for comparison based on two criteria: 1) the models perform well on MMSD or MMSD2.0, or 2) their code is publicly available. Based on these criteria, we choose GPT4o, InternLM2-vl-7b, InternLM2-vl-7b + LoRA, DemoRetrieval, DynRT-Net, and M-CLIP for comparison.

	RedEval			
	Acc. (%)	P (%)	R (%)	F1 (%)
GPT-4o	74.37	61.10	96.17	74.73
InternLM2-vl-7b	74.10	64.64	75.44	69.63

Table 4: The results of two MLLMs on RedEval.

As shown in Table 4, for untrained MLLMs, both InternLM2-vl-7b and GPT4o exhibit poor performance at RedEval. As shown Table 3, compared to MMSD2.0, we observe that most baselines trained on MMSD suffer substantial performance drops. Notably, DynRT-Net experiences a decline of over 30% in accuracy and more than 50% in F1 score. This indicates that MMSD contains spurious cues and that models are prone to overfitting to low-quality and in-domain data. In contrast, our model, MPMSD w/o MLLM, only shows a modest drop of 5.32% in accuracy and 7.83% in F1, demonstrating multi-perspective modeling's robustness to noise. The full MPMSD model further improves performance, with results on RedEval closely matching those on MMSD, suggesting that external knowledge generated by MLLMs in our method can more effectively mitigate noise impacts and enhance model robustness.

4.5 Submodels for Six Categories

Categories	f_{kl}	f_{im}	f_{rm}	ToTal
Inconsistency	80	89	93	106
Joke	30	24	21	34
Implicit Irony	667	680	717	800
Understatement	6	6	6	6
Overstatement	24	24	21	26
Rhetorical Question	53	59	60	65

Table 5: Submodels' results. f_{kl}, f_{im}, and f_{re} denote Knowledge Learning (Model 1), Incongruity Mining (Model2), and Representation Enhancement (Model 3), respectively.

In this section, we show the detection effects of submodels on different categories of sarcasm. First, we manually classified the samples labeled as 1 in MMSD2.0 test set into six categories. Then, three submodels are trained separately on MMSD2 training set to test samples in test set. As shown in Table 4, "ToTal" denotes the number of samples of a certain category, and f_{kl} denotes the number of sarcastic samples correctly identified by Model 1.

For the category "Inconsistency", Model 3 and Model 2 accurately identify a similar number, both better than Model 1. For "Joke", Model 1 is better than Model 2 and Model 3. However, the quantities of these two types are too small and not representative. For the more numerous category "Implicit Irony", Model 3 correctly identified 47 more than Model 2, and Model 2 correctly identified 13 more than Model 1. Similar situations also exist in other types, except for "Understatement". Overall, submodels exhibited varying sensitivities to different types and different submodels have varying abilities to extract features for different types. Therefore, it is necessary for three submodels to collaboratively detect sarcasm from multiple perspectives.

4.6 Analysis

4.6.1 MPMSD with Different MLLMs

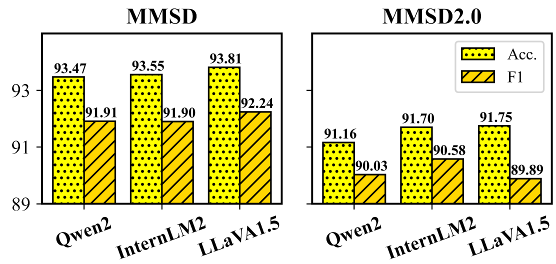

Figure 5: MPMSD with different MLLMs.

To validate that our framework is general and does not rely on a specific MLLM, we test MPMSD using Qwen2-vl-7b (Wang et al., 2024) and LLaVA1.5-7b (Liu et al., 2023). As shown in Figure 5, MPMSD demonstrates minimal performance differences across different MLLMs on two datasets. Due to the instability in MLLM generation and subsequent training process, these differences can be considered negligible, which further demonstrates that our framework is general for many MLLMs. Additionally, some examples generated by InternLM will be shown in **Appendix** E.

4.6.2 The Effectiveness of Submodels

In this section, we evaluate the effectiveness of three submodels. As shown in Table 6, all experiments involving combinations of two or more submodels incorporate Feature Fusion in Sec.(§3.4).

We observe that each submodel individually performs poorly, while their combinations lead to substantial performance gains, highlighting the importance of multi-perspective feature modeling for sarcasm detection. Excitingly, on MMSD2.0, all pairwise combinations surpass the current SOTA, and combining all three submodels yields a significant improvement over the SOTA baseline. Ablation studies in f_{re} and three submodels further show

f_{kl}	f_{im}	f_{re}	MMSD		MMSD2.0	
			Acc.	F1	Acc.	F1
✓	✓	✓	**93.55**	**91.90**	**91.70**	**90.58**
✓			84.91	81.35	81.32	79.66
	✓		85.71	81.75	81.98	80.45
		✓	87.11	84.01	85.18	83.72
		–	86.68	83.53	84.62	82.57
✓	✓		91.74	89.77	89.17	87.90
✓		✓	89.42	86.73	87.63	86.59
	✓	✓	89.38	86.89	87.92	86.65
✓	✓	–	93.21	91.62	91.30	90.21

Table 6: Ablation study of submodels. – refers to the contrastive learning is removed in f_{re}.

that our contrastive learning approach enhances representation learning and contributes positively to sarcasm detection.

4.6.3 The Effectiveness of Feature Fusion

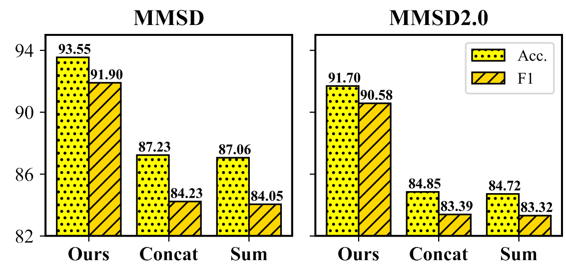

Figure 6: MPMSD with different prediction methods.

To further verify the effectiveness of Feature Fusion, we use different prediction methods. The first approach directly concatenates f_{kl}, f_{im} and f_{re}, followed by a prediction. The second approach independently predicts from f_{kl}, f_{im} and f_{re}, and then aggregates the results by summing probability distributions. As shown in Figure 6, the performance decreases significantly compared to ours. This highlights the importance of dynamically fusing the outputs of three submodels based on output's features during inference, thereby validating the effectiveness of Feature Fusion.

5 Conclusion

In this paper, we observe that previous methods struggle to understand data and overlook the diverse expressions of sarcasm. To this end, we leverage MLLMs to enhance data understanding and employ three distinct submodels to capture sarcasm features from multiple perspectives. Experiments demonstrate that MPMSD not only achieves superior performance but also exhibits strong generalization capabilities and robust noise resistance.

Limitations

Firstly, MPMSD is constrained by the inherent performance of MLLMs. Secondly, compared to other baselines in $Multimodal\ methods$, using MLLMs to generate data demands greater computational resources. Thirdly, in Incongruity Mining, concatenating representations to construct a unified graph consumes substantial GPU memory during training. Fourth, we do not conduct further research and discussion on the six types of sarcasm.

References

Yitao Cai, Huiyu Cai, and Xiaojun Wan. 2019. Multi-modal sarcasm detection in Twitter with hierarchical fusion model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2506–2515, Florence, Italy. Association for Computational Linguistics.

Erik Cambria, Yang Li, Frank Z Xing, Soujanya Poria, and Kenneth Kwok. 2020. Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 105–114.

Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, Wenwei Zhang, Yining Li, Hang Yan, Yang Gao, Xinyue Zhang, Wei Li, Jingwen Li, Kai Chen, Conghui He, Xingcheng Zhang, Yu Qiao, Dahua Lin, and Jiaqi Wang. 2024. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *arXiv preprint arXiv:2401.16420*.

Shafkat Farabi, Tharindu Ranasinghe, Diptesh Kanojia, Yu Kong, and Marcos Zampieri. 2024. A survey of multimodal sarcasm detection. *arXiv preprint arXiv:2410.18882*.

Diandian Guo, Cong Cao, Fangfang Yuan, Yanbing Liu, Guangjie Zeng, Xiaoyan Yu, Hao Peng, and Philip S Yu. 2024. Multi-view incongruity learning for multimodal sarcasm detection. *arXiv preprint arXiv:2412.00756*.

Jiaqi Hao, Junfeng Zhao, and Zhigang Wang. 2024. Multi-modal sarcasm detection via graph convolutional network and dynamic network. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 789–798.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Mengzhao Jia, Can Xie, and Liqiang Jing. 2024. Debiasing multimodal sarcasm detection with contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18354–18362.

Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 2015. Harnessing context incongruity for sarcasm detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 757–762.

John S Leggitt and Raymond W Gibbs. 2000. Emotional reactions to verbal irony. *Discourse processes*, 29(1):1–24.

Jiangnan Li, Hongliang Pan, Zheng Lin, Peng Fu, and Weiping Wang. 2021. Sarcasm detection with commonsense knowledge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3192–3201.

Bin Liang, Chenwei Lou, Xiang Li, Lin Gui, Min Yang, and Ruifeng Xu. 2021. Multi-modal sarcasm detection with interactive in-modal and cross-modal graphs. In *Proceedings of the 29th ACM international conference on multimedia*, pages 4707–4715.

Bin Liang, Chenwei Lou, Xiang Li, Min Yang, Lin Gui, Yulan He, Wenjie Pei, and Ruifeng Xu. 2022. Multi-modal sarcasm detection via cross-modal graph convolutional network. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1767–1777. Association for Computational Linguistics.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning.

Hui Liu, Wenya Wang, and Haoliang Li. 2022. Towards multi-modal sarcasm detection via hierarchical congruity modeling with knowledge enhancement. *arXiv preprint arXiv:2210.03501*.

I Loshchilov. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

OpenAI. 2024. Gpt-4o. Large language model. Accessed on April 15, 2025.

Hongliang Pan, Zheng Lin, Peng Fu, Yatao Qi, and Weiping Wang. 2020a. Modeling intra and inter-modality incongruity for multi-modal sarcasm detection. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1383–1392, Online. Association for Computational Linguistics.

Hongliang Pan, Zheng Lin, Peng Fu, Yatao Qi, and Weiping Wang. 2020b. Modeling intra and inter-modality incongruity for multi-modal sarcasm detection. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1383–1392.

MSM Prasanna, SG Shaila, and A Vadivel. 2023. Polarity classification on twitter data for classifying sarcasm using clause pattern for sentiment analysis. *Multimedia Tools and Applications*, 82(21):32789–32825.

Yang Qiao, Liqiang Jing, Xuemeng Song, Xiaolin Chen, Lei Zhu, and Liqiang Nie. 2023. Mutual-enhanced incongruity learning network for multi-modal sarcasm detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 9507–9515.

Libo Qin, Shijue Huang, Qiguang Chen, Chenran Cai, Yudi Zhang, Bin Liang, Wanxiang Che, and Ruifeng Xu. 2023. MMSD2.0: Towards a reliable multi-modal sarcasm detection system. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10834–10845, Toronto, Canada. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Rossano Schifanella, Paloma De Juan, Joel Tetreault, and Liangliang Cao. 2016. Detecting sarcasm in multimodal social platforms. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 1136–1145.

Binghao Tang, Boda Lin, Haolong Yan, and Si Li. 2024. Leveraging generative large language models with visual instruction and demonstration retrieval for multimodal sarcasm detection. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1732–1742.

Yuan Tian, Nan Xu, Ruike Zhang, and Wenji Mao. 2023. Dynamic routing transformer network for multimodal sarcasm detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2468–2480.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Xinyu Wang, Xiaowen Sun, Tan Yang, and Hongbo Wang. 2020. Building a bridge: A method for image-text sarcasm detection without pretraining on image-text data. In *Proceedings of the First International Workshop on Natural Language Processing Beyond Text*, pages 19–29, Online. Association for Computational Linguistics.

Yiwei Wei, Shaozu Yuan, Hengyang Zhou, Longbiao Wang, Zhiling Yan, Ruosong Yang, and Meng Chen. 2024. Gˆ 2sam: Graph-based global semantic awareness method for multimodal sarcasm detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9151–9159.

Nan Xu, Zhixiong Zeng, and Wenji Mao. 2020. Reasoning with multimodal sarcastic tweets via modeling cross-modality contrast and semantic association. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3777–3786, Online. Association for Computational Linguistics.

Haofei Yu, Zhengyang Qi, Lawrence Jang, Russ Salakhutdinov, Louis-Philippe Morency, and Paul Pu Liang. 2024. Mmoe: Enhancing multimodal models with mixtures of multimodal interaction experts. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10006–10030.

Jiecheng Zhang, C.L.Philip Chen, Shuzhen Li, and Tong Zhang. 2025. Incongruity-aware tension field network for multi-modal sarcasm detection. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14499–14508, Vienna, Austria. Association for Computational Linguistics.

Zhihong Zhu, Xuxin Cheng, Guimin Hu, Yaowei Li, Zhiqi Huang, and Yuexian Zou. 2024a. Towards multi-modal sarcasm detection via disentangled multi-grained multi-modal distilling. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16581–16591, Torino, Italia. ELRA and ICCL.

Zhihong Zhu, Xianwei Zhuang, Yunyan Zhang, Derong Xu, Guimin Hu, Xian Wu, and Yefeng Zheng. 2024b. Tfcd: Towards multi-modal sarcasm detection via training-free counterfactual debiasing. In *Proc. of IJCAI*.

A Frozen CLIP's Parameters

In this paper, we utilize a frozen CLIP encoder and then employ three parallel submodels to update distinct representations, with the aim of modeling sarcastic features from three distinct perspectives. However, if CLIP's parameters are not frozen, the updates to the representations would first be processed through a shared CLIP before being further updated by three submodels. We posit that this could result in a degree of feature assimilation, thereby reducing the independence of the representations cultivated by each submodel and consequently diminishing the model's capacity to model from multiple perspectives.

	MMSD		MMSD2.0	
	Acc.	F1	Acc.	F1
frozen	**93.55**	**91.90**	**91.70**	**90.58**
unfrozen	91.07	88.55	89.17	87.48

Table 7: If CLIP is frozen or not.

As shown in Table 7, we conduct experiments on unfrozen CLIP, the results show that the performance of the model decrease significantly.

B Baselines

Our work utilizes MLLMs as an information source to extracting external knowledge. Therefore, we explore two types of baselines: (i) *MLLMs methods* and (ii) *Multimodal methods*.

(i) For *MLLMs methods*, we conduct tests on the following 5 models:

- InternLM2-vl-7b (Dong et al., 2024)
- InternLM2-vl-7b with LoRA: The corresponding prompts and implementation details are provided in **Appendix C**.
- GPT-4o (OpenAI, 2024)
- MMoE (Yu et al., 2024) proposes to train separate expert models (MLLMs) for each type of multimodal data for sarcasm detection.
- DemoRetrieval (Tang et al., 2024) retrieves similar samples, and fine-tuned MLLM generates sarcastic labels based on those similar samples.

(ii) For *Multimodal methods*, we conduct tests on the following 12 models:

- Att-BERT (Pan et al., 2020b) adopts self-attention and co-attention mechanisms to model the intra-modality and inter-modality incongruity respectively.
- InCrossMGs (Liang et al., 2021) utilizes a graph-based model to capture sarcastic relations between image and text
- CMGCN (Liang et al., 2022) proposes a fine-grained cross-modal graph architecture to capture sarcastic clues.
- HKE (Liu et al., 2022) explores atomic-level consistency based on multi-head cross-attention mechanisms and combinatorial-level consistency based on graph neural networks.
- MILNet (Qiao et al., 2023) utilizes the underlying consistency between the two modules to improve performance.
- DynRTNet (Tian et al., 2023) proposes a dynamic routing transformer network to capture the sarcastic clues from images and texts.
- M-CLIP (Qin et al., 2023) utilizes a framework based on CLIP (Radford et al., 2021) from image view, text view, and image-text interactions view for MSD.
- M-CLIP+TFCD (Zhu et al., 2024b) proposes a training-free counterfactual debiasing framework based on M-CLIP.
- DMSD-CL (Jia et al., 2024) proposes a debiasing multimodal sarcasm detection framework with contrastive learning aimed at mitigating the deleterious effects of biased textual factors on robust generalization.
- G^2SAM (Wei et al., 2024) uses global graph-based semantic awareness for multimodal sarcasm detection.
- DMMD (Zhu et al., 2024a) proposes a framework dubbed disentangled multigrained multimodal distilling for multimodal sarcasm detection.
- ITFNet (Zhang et al., 2025) proposes a pioneering inconsistency preference method called incongruity aware tension field network for multimodal sarcasm detection.

C InternLM Evaluation

We use the zero-shot prompt shown in the figure below to evaluate InternLM2-vl-7b. Because of the instability generated by LLMs, manual verification is required during the final evaluation.

> *Zero shot*: A data is a text with a image. Given a data (image: ***I***, text: ***T***), please select the sarcasm label from {0, 1}, your choice is

> *LoRA*: According to image: ***I*** and text: ***T***, please select the sarcasm label from {0,1}.

For LoRA (Hu et al., 2021), we inject low-rank matrices as adapters into the model. The rank of the low-rank matrices is set to 64, with a scaling factor of 64 for LoRA updates. The batch size is set to 1, and the training runs for 1 epoch, with gradient accumulation performed over 8 steps to simulate a larger effective batch size. The model is trained on a single NVIDIA A100 GPU, utilizing BF16 precision.

D RedEval Dataset

RedEval	Positive	Negative	Total
	395	609	1004

Table 8: Statistics of RedEval.

RedEval was proposed by (Tang et al., 2024) and collected from another social platform, Reddit[2]. They selected image-text data from the “sarcasm” subreddit as positive sarcasm samples, and a certain number of samples from other subreddits. Just like MMSD2.0, the emotion in RedEval has been removed. Tang et al. employ 3 graduate students to ensure the quality of image-text pairs in RedEval aligns with intended meaning of sarcastic labels. Specifically, each image-text pair is reviewed by two annotators, who are tasked with predicting the sarcasm label. Only the samples that receive consistent predictions from both annotators are retained. The statistics about RedEval are shown in Tabel 8.

E InternLM Generation

As illustrated in Figure 7, two samples are from Figure 1. For the first example, InternLM doesn't indicate that "lmao" in text is an abbreviation for "laughing my ass off," but instead suggests that "lmao" means "laughing or joking." For the second example, InternLM explicitly notes that the dog in the image is incorrectly labeled as "cat" in the caption. However, for overly complex images, InternLM with fewer parameters still cannot accurately interpret them, which will be discussed in the **Appendix F**.

	I^m by InternLM	T^m by InternLM
 # tbt to my first time in the desert lmao	The image features a young boy standing in front of a desert landscape. He is wearing a gray t-shirt and has blonde hair. The background shows a rocky mountain range under a clear blue sky.	This text appears to be a casual expression or hashtag, possibly referring to a throwback or reminiscing about a past experience in the desert. The phrase "lmao" suggests that the person might be laughing or joking about the situation.
 all the damn time	The image is a black and white photograph of Helen Keller, an American author, political activist, and lecturer, along with her dog. The text above the image reads "Helen Keller and her cat." However, it appears to be a caption error as the image actually shows Helen Keller with her dog, not a cat.	The phrase "all the damn time" means consistently or constantly, often used to emphasize the frequency or intensity of something.

Figure 7: InternLM's Generation about Images and Text.

F Inaccurate Understanding of Images

We find that for highly complex images, InternLM cannot accurately interpret them. Moreover, if sarcastic semantics are primarily present in such images, MPMSD will exhibit poor performance on this type of data. As illustrated in Figure 8, two four-panel images reveal their sarcastic or humorous implications. However, InternLM failed to accurately understand them. The figure demonstrates inaccurate I^m, with GPT-4o correctly interpreting the content of the images.

G Low-Resource Scenario

Following (Qin et al., 2023), we explore MPMSD's performance in low-resource scenario of MMSD2.0. Specifically, we compared our model with models mentioned in generalization ability (Sec.(§4.4)), except for DynRT-Net, as DynRT-Net performs poorly on MMSD2.0.

As shown in Figure 9, M-CLIP, InternLM2-vl-7b + LoRA, and DemoRetrieval all exhibit slow performance growth with the increase of training data, with an accuracy of around 86% as their per-

[2] https://www.reddit.com/

	Inaccurate I^m by InternLM	**Accurate I^m by GPT4o**
 two types of guys :	The image is a collage of four scenes from the TV show "The Monkees." The main characters, Davy Jones and Micky Dolenz, are seen in various winter settings. In one scene, Davy Jones is wearing a red jacket and saying, "Well here, take my jacket. I love you." Micky Dolenz is also present in this scene. Another scene shows Davy Jones and Micky Dolenz standing next to each other with snow-covered trees in the background. Davy Jones is wearing a blue sweater and says, "Well damn, Jackiel. I can't control the weather!"	The image is a four-panel meme from the TV show That '70s Show. Here's a breakdown: Top-left panel: Donna says, "It's kind of cold." Eric responds by offering his jacket. Top-right panel: Eric says, "Well here, take my jacket. I love you." Bottom-left panel: Jackie says, "I'm cold, too." Hyde is standing next to her. Bottom-right panel: Kelso responds, "Well damn, Jackie! I can't control the weather!" Summary: The meme humorously contrasts Eric's romantic gesture with Kelso's frustration, showing different relationship dynamics.
 like if you agree .. :p	The comic strip consists of four panels, each featuring a character with red eyes and a white face. In the first panel, the character is asking "Hey Christine, wanna dance?" The second panel shows the same character saying "Okay... I'll sit here." In the third panel, the character is encouraging another character to go dancing by saying "Okay go stand... I'll sit here." Finally, in the fourth panel, the character is looking at the other character and saying "Okay go dance... I'll sit here."	The image is a four-panel comic strip: 1. A boy asks a girl (Christine), "Hey Christine... Wanna Dance?" She replies, "Okay..." while getting up from her chair. 2. The boy walks her to the dance floor. 3. The boy then says, "Okay go dance... I'll sit here," indicating her seat. 4. He smugly sits on her chair while she stands, confused. Humor: The boy tricked the girl into giving up her chair rather than actually asking her to dance.

Figure 8: InternLM's and GPT4o's Generation about Images.

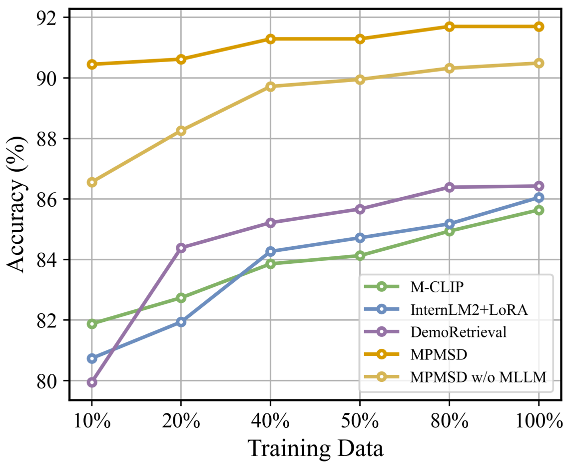

Figure 9: Low-Resource Performance on MMSD2.0.

formance ceiling. However, MPMSD achieves an accuracy of over 90% even under extremely low resource conditions, such as with only 10% of training data. MPMSD w/o MLLM consistently underperforms compared to MPMSD across different proportions of training data, indicating the importance of external knowledge for multimodal sarcasm detection. Furthermore, regardless of the presence of external knowledge, our method achieves significantly better performance than all baselines under the low-resource scenario. This demonstrates another advantage of multi-perspective modeling, namely its high data efficiency.

H Additional Details

In this section, we provide more details about MPMSD: 1) For Knowledge Acquisition (Sec.(§3.1)), generating 1000 samples for the text-understanding and image-understanding components required an average of 0.79 GPU-hours and 1.55 GPU-hours, respectively. 2) For MPMSD, training on the training set consumed 3.32 GPU-hours, while a single inference pass on the validation or test set required 0.0275 GPU-hours. 3) For MPMSD w/o MLLM, training cost 1.26 GPU-hours, and one inference pass on the validation or test set took 0.019 GPU-hours. 4) Parameters: MPMSD≈167M, MPMSD w/o MLLM≈170M.