InterCLIP-MEP: Interactive CLIP and Memory-Enhanced Predictor for Multi-modal Sarcasm Detection

Anonymous ACL submission

Abstract

Sarcasm in social media, often expressed through text-image combinations, poses challenges for sentiment analysis and intention mining. Current multi-modal sarcasm detection methods have been demonstrated to overly rely on spurious cues within the textual modality, revealing a limited ability to genuinely identify sarcasm through nuanced text-image interactions. To solve this problem, we propose InterCLIP-MEP, which introduces Interactive CLIP (InterCLIP) with an efficient training strategy to extract enriched text-image representations by embedding cross-modal information directly into each encoder. Additionally, we design a Memory-Enhanced Predictor (MEP) with a dynamic dual-channel memory that stores valuable test sample knowledge during inference, acting as a non-parametric classifier for robust sarcasm recognition. Experiments on two benchmarks demonstrate that InterCLIP-MEP achieves state-of-the-art performance, with significant accuracy and F1 score improvements on MMSD and MMSD2.0. *Our code is in the supplementary material.*

1 Introduction

Sarcasm, with its subtlety and complexity, plays a key role in communication by conveying irony, mockery, or hidden meanings (Muecke, 1982; Gibbs and O'Brien, 1991; Gibbs and Colston, 2007). Automatically detecting sarcasm in text has become an important research area, supporting tasks like sentiment analysis and intent mining (Pang et al., 2008; Tsur et al., 2010; Bouazizi and Ohtsuki, 2015). With the rise of social media platforms like Twitter and Reddit, users often use text-image combinations to express their messages. As a result, multi-modal sarcasm detection is increasingly important, posing challenges in understanding the complex relationship between textual and visual cues to identify sarcasm.

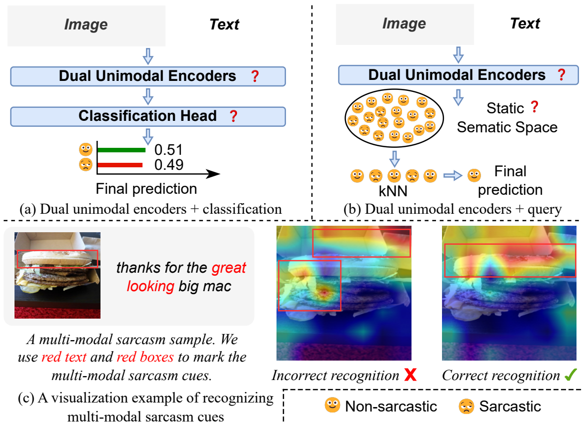

Figure 1: An overview of the shortcomings of existing multi-modal sarcasm detection pipelines. In panels (a) and (b), we present two main multi-modal sarcasm detection pipelines, with shortcomings indicated by a *red question mark*. In panel (c), we visually show an example of multi-modal sarcasm cues correctly or incorrectly recognized in a multi-modal sarcasm sample.

As shown in Figures 1(a) and 1(b), many methods rely on dual unimodal pre-trained encoders, such as ViT(Dosovitskiy et al., 2021) and BERT (Devlin et al., 2019), as the backbone for encoding text-image pairs, followed by specific feature fusion (Xu et al., 2020; Pan et al., 2020; Liang et al., 2021, 2022; Wen et al., 2023; Tian et al., 2023; Wei et al., 2024). However, this approach may not capture multi-modal sarcasm cues as effectively as multi-modal pre-trained models like CLIP (Radford et al., 2021). In Figure 1(a), the use of a learnable classification head to predict labels from fused representations is common but often associated with high predictive entropy and significant uncertainty. Wei et al. (2024) pioneered the construction of a static semantic space using historical training samples, where more robust predictions are obtained during inference through KNN-based querying and voting, as illustrated in Figure 1(b). However, while CLIP has been proven to be effective in serving as a text-image encoder for multi-modal sarcasm detection (Qin et al., 2023), it still struggles to capture multi-modal sarcasm cues due to the inherent inconsistency of sarcasm, which conflicts with CLIP's direct alignment of text and image. Furthermore, relying on a static semantic space for inference is ill-suited to handle the dynamic nature of evolving sample distributions. In fact, Qin et al. (2023) have shown that many models rely on spurious cues in the MMSD benchmark (Cai et al., 2019), leading to biased results.

Building on the limitations of prior multi-modal sarcasm detection approaches, we propose Interactive CLIP (InterCLIP) as the backbone, embedding cross-modal representations directly into text and vision encoders to enhance the understanding of multi-modal sarcasm cues (Figure 2, left). To complement this, we design a Memory-Enhanced Predictor (MEP) that dynamically utilizes historical test sample features to create a more adaptive and reliable non-parametric classifier for final predictions (Figure 2, right). Together, these components form the proposed framework, InterCLIP-MEP. Furthermore, InterCLIP-MEP employs an efficient training strategy that fine-tunes cross-modal interactions through a lightweight adaptation mechanism, ensuring computational efficiency while delivering state-of-the-art performance (Figure 2, left). Overall, our contributions are as follows:

- We introduce `InterCLIP-MEP`[1], a novel framework for multi-modal sarcasm detection, which combines Interactive CLIP (InterCLIP) for enhanced text-image interaction encoding and Memory-Enhanced Predictor (MEP) for more robust and reliable sarcasm predictions.
- We propose an efficient training strategy that significantly reduces computational overhead compared to state-of-the-art methods. By introducing approximately 20.6x fewer trainable parameters, our approach reduces GPU memory usage by about 2.5x and accelerates computation by roughly 8.7x with a batch size of 128, all while maintaining superior performance on a single NVIDIA RTX 4090 GPU.
- Through extensive experiments on the MMSD and MMSD2.0 benchmarks, we show that InterCLIP-MEP improves accuracy by 1.08% and F1 score by 1.51% over state-of-the-art methods, especially on MMSD2.0.

[1]Our code is available in the supplementary material.

2 Related Work

Early research in sarcasm detection focused primarily on text data (Bouazizi and Ohtsuki, 2015; Amir et al., 2016; Baziotis et al., 2018). With the rise of social media, detecting sarcasm in text-image pairs has become more challenging, driving the development of multi-modal approaches. Schifanella et al. (2016) were among the first to explore multi-modal social media posts for identifying sarcasm cues. Building on this, Cai et al. (2019) introduced the MMSD benchmark, demonstrating the effectiveness of a hierarchical fusion model that integrates image features. This benchmark has since become a foundation for multi-modal sarcasm detection, inspiring subsequent studies (Xu et al., 2020; Pan et al., 2020; Liang et al., 2021, 2022; Liu et al., 2022; Qin et al., 2023; Wen et al., 2023; Tian et al., 2023; Wei et al., 2024).

However, the MMSD benchmark was later found to contain spurious cues that could lead to model bias (Qin et al., 2023). To mitigate this, Qin et al. (2023) introduced the MMSD2.0 benchmark, which removes these cues and corrects mislabeled samples. Re-evaluations on MMSD2.0 revealed significant performance drops in existing models, emphasizing the need for more robust approaches. In parallel, Tang et al. (2024) explored the use of large language models (LLMs) in multi-modal sarcasm detection, incorporating instruction templates and retrieval modules. While promising, the performance improvements were modest compared to the substantial computational cost.

In this work, we present InterCLIP-MEP, a lightweight and efficient framework that achieves competitive performance without the high resource demands of LLM-based approaches. By overcoming the limitations of current methods, our approach offers a practical and scalable solution for multi-modal sarcasm detection.

3 Methodology

An overview of InterCLIP-MEP is illustrated in Figure 2. Initially, we elaborate on the Interactive CLIP (InterCLIP) and its training strategy, followed by an in-depth explanation of the Memory-Enhanced Predictor (MEP).

3.1 Interactive CLIP

The input to Interactive CLIP (InterCLIP) is a text-image pair $\mathcal{P} = (T, I)$, where T represents a piece

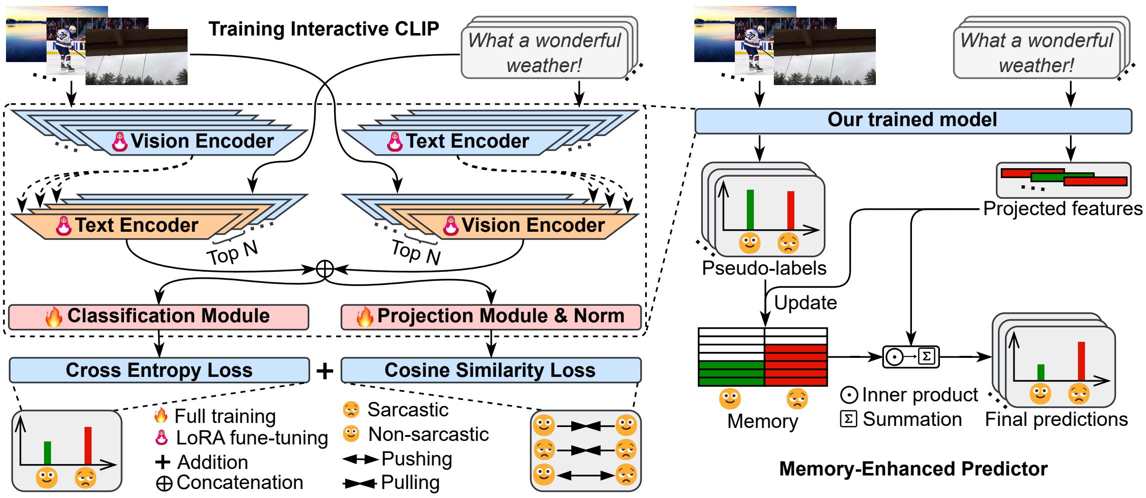

Figure 2: Overview of our framework. **(I) Training Interactive CLIP (InterCLIP):** Vision and text representations are extracted using separate encoders and embedded into the top-n layers of the opposite modality's encoder for interaction. The top-n layers are fine-tuned with LoRA, while the rest of the encoder remains frozen. Final vision and text representations are concatenated and used to train a classification module for identifying multi-modal sarcasm. A projection module is also trained to project representations into a latent space. **(II) Memory-Enhanced Predictor (MEP):** During inference, InterCLIP generates interactive representations. The classification module assigns pseudo-labels, and the projection module provides projection features. MEP updates dynamic memory with these features and pseudo-labels. The final prediction of the current sample is made by comparing its projected feature with those in memory.

of text and I represents an image. Here, for simplicity, we do not consider the case of batch inputs.

The text encoder $\mathcal{T}$ extracts the vanilla text representations $\mathbf{F}_t$:

$$\begin{aligned}\mathbf{F}_t &= \mathcal{T}(T) \\ &= \{h^t_{\text{bos}}(t_{\text{bos}}), h^t_1(t_1), \ldots, h^t_n(t_n), h^t_{\text{eos}}(t_{\text{eos}})\},\end{aligned} \tag{1}$$

where t_i denotes a text token, n is the length of T after tokenization, t_{bos} and t_{eos} are special tokens required by the text encoder. Here, $h^t_i(\cdot) \in \mathbb{R}^{d_t}$ represents the d_t-dimensional encoded representation of the corresponding token t_i, with i ranging from 1 to n, including the beginning-of-sequence (bos) and end-of-sequence (eos) tokens.

The vision encoder $\mathcal{V}$ extracts the vanilla image representations $\mathbf{F}_v$:

$$\mathbf{F}_v = \mathcal{V}(I) = \{h^v_{\text{cls}}(p_{\text{cls}}), h^v_1(p_1), \ldots, h^v_m(p_m)\}, \tag{2}$$

where I is processed into multiple patches p_i, m is the number of patches, and p_{cls} is a special token required by the visual encoder. Here, $h^v_i(\cdot) \in \mathbb{R}^{d_v}$ represents the d_v-dimensional encoded representation of the corresponding p_i, with i ranging from 1 to m, including the classification (cls) token. Specifically, both $\mathbf{F}_t$ and $\mathbf{F}_v$ are representations from the final layer outputs of their respective encoders. Conditioning on $\mathbf{F}_t$ or $\mathbf{F}_v$, we can obtain the interactive text representations $\tilde{\mathbf{F}}_t$ or the interactive image representations $\tilde{\mathbf{F}}_v$:

$$\tilde{\mathbf{F}}_t = \mathcal{T}(T|\mathbf{F}_v), \tilde{\mathbf{F}}_v = \mathcal{V}(V|\mathbf{F}_t). \tag{3}$$

We use $\tilde{h}^t_i(\cdot) \in \mathbb{R}^{d_t}$ and $\tilde{h}^v_i(\cdot) \in \mathbb{R}^{d_v}$ to denote the re-encoded interactive representations of each text token and image patch, respectively.

To be specific, we condition only the top-n self-attention layers of the text or vision encoder, where n is a hyperparameter that will be analyzed in the experiment section. Figure 3 illustrates the structure of the conditioned self-attention layers. Given that the text and vision encoder in CLIP share a similar architecture, for brevity, we denote the input representations to the self-attention layers of the text or vision encoder as $\mathbf{H}_{t/v}$, which are derived from the outputs of the previous layer. The previous layer can either be conditioned or non-conditioned. Due to the dimensional mismatch between the embedded representations $\mathbf{F}_{v/t}$ and the corresponding encoder representation space, we introduce an adapting projection layer $\mathcal{F}_{t/v}$ to project $\mathbf{F}_{v/t}$ into the appropriate representation space.

To fuse the input representations $\mathbf{H}_{t/v}$ with the projected embedded representations $\mathbf{F}'_{v/t} = \mathcal{F}_{t/v}(\mathbf{F}_{v/t})$, we concatenate them and feed them

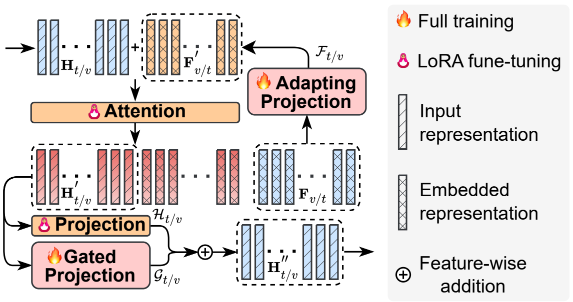

Figure 3: Structure of the conditional self-attention.

into the attention layer to obtain the transformed representations. We then extract the transformed input representations $\mathbf{H}'_{t/v}$ from the output. Following Ganz et al. (2024), we apply a gated projection layer $\mathcal{G}_{t/v}$ along with the self-attention's projection head $\mathcal{H}_{t/v}$ using a learnable gating mechanism to compute the self-attention output representations $\mathbf{H}''_{t/v}$. Given the similarity between the self-attention layers of the vision encoder and the text encoder, we use the text encoder $\mathcal{T}$ to illustrate the process as follows:

$$\begin{aligned}
&\mathbf{F}'_v = \mathcal{F}_t(\mathbf{F}_v), \quad \mathbf{F}_v \in \mathbb{R}^{m\times d_v}, \mathbf{F}'_v \in \mathbb{R}^{m\times d_t},\\
&\mathbf{H}'_t = \mathrm{Attn}_t(\mathbf{H}_t \oplus \mathbf{F}'_v)_{[:n]},\\
&\qquad \mathbf{H}_t, \mathbf{H}'_t \in \mathbb{R}^{n\times d_t}, \mathbf{H}_t \oplus \mathbf{F}'_v \in \mathbb{R}^{(n+m)\times d_t},\\
&\mathbf{H}''_t = \mathcal{H}_t(\mathbf{H}'_t) + \mathcal{G}_t(\mathbf{H}'_t)\cdot \tanh(\beta_t), \mathbf{H}''_t \in \mathbb{R}^{n\times d_t}.
\end{aligned} \tag{4}$$

Here, $\oplus$ denotes the concatenation operation, and β_t is a learnable gating parameter initialized to 0 to ensure training stability. The subsequent computation follows the original CLIP (Radford et al., 2021), ultimately yielding the interactive representations $\tilde{\mathbf{F}}_t$.

InterCLIP supports three interaction modes for fusing text and image features into the final representation $\tilde{h}^f \in \mathbb{R}^{d_t+d_v}$:

- **T2V:** Text representations $\mathbf{F}_t$ are embedded into the vision encoder to produce interactive image representations $\tilde{\mathbf{F}}_v$. $\tilde{h}^f$ is formed by concatenating h^t_{eos} and $\tilde{h}^v_{\text{cls}}$.
- **V2T:** Image representations $\mathbf{F}_v$ are embedded into the text encoder to produce interactive text representations $\tilde{\mathbf{F}}_t$. $\tilde{h}^f$ is formed by concatenating $\tilde{h}^t_{\text{eos}}$ and h^v_{cls}.
- **Two-way (TW):** Both text and image representations $\mathbf{F}_t$ and $\mathbf{F}_v$ are embedded into each other's encoders, resulting in $\tilde{\mathbf{F}}_t$ and $\tilde{\mathbf{F}}_v$. $\tilde{h}^f$ is formed by concatenating $\tilde{h}^t_{\text{eos}}$ and $\tilde{h}^v_{\text{cls}}$.

We will analyze the effectiveness of these three interaction modes in the experimental analysis.

Training Strategy. As shown in Figure 2 (left), to adapt InterCLIP for MEP, we introduce an efficient training strategy. Using InterCLIP as the backbone to obtain fused features of the samples, we introduce a classification module and a projection module.

Given the fused features of a batch of samples $\tilde{H}^f \in \mathbb{R}^{N\times(d_t+d_v)}$, the classification module $\mathcal{F}_c$ calculates the probabilities $\hat{y}$ of these samples being sarcastic or non-sarcastic:

$$\hat{y} = \mathrm{softmax}(\mathcal{F}_c(\tilde{H}^f)), \quad \hat{y} \in \mathbb{R}^{N\times 2}, \tag{5}$$

where N denotes the batch size. We optimize $\mathcal{F}_c$ using binary cross-entropy loss:

$$\mathcal{L}^c = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_{i,1}) + (1-y_i)\log(1-\hat{y}_{i,0})\right], \tag{6}$$

where y_i denotes the label of the i-th sample, with sarcastic labeled as 1 and non-sarcastic as 0, and $\hat{y}_i$ denotes the prediction for the i-th sample.

The projection module $\mathcal{F}_p$ maps $\tilde{H}^f$ into a latent feature space:

$$\hat{H}^f = \mathrm{norm}(\mathcal{F}_p(\tilde{H}^f)), \quad \hat{H}^f \in \mathbb{R}^{N\times d_f}, \tag{7}$$

where $\mathrm{norm}(\cdot)$ denotes L2 normalization, and d_f represents the dimension of the projected features. In this space, the cosine distance between features of the same class is minimized, while the distance between features of different classes is maximized. We use a label-aware cosine similarity loss to optimize $\mathcal{F}_p$:

$$\begin{aligned}
\mathcal{L}^p = \mathrm{mean}(\hat{H}^f_P \cdot \hat{H}^{f^T}_N) + \mathrm{mean}(1 - \hat{H}^f_P \cdot \hat{H}^{f^T}_P)\\
+ \mathrm{mean}(1 - \hat{H}^f_N \cdot \hat{H}^{f^T}_N),
\end{aligned} \tag{8}$$

where $\hat{H}^f_P$ and $\hat{H}^f_N$ represent the projected features of positive and negative samples, respectively.

We fully train the modules $\mathcal{F}_c$, $\mathcal{F}_p$, the adapting projection layers ($\mathcal{F}_t$ and $\mathcal{F}_v$), the gated projection layers ($\mathcal{G}_t$ and $\mathcal{G}_v$), and the learnable gating parameters (β_t and β_v). We use LoRA (Hu et al., 2022) to fine-tune parts of the weight matrices $\mathbf{W}$ in the self-attention modules of all encoders, specifically various combinations of W_q, W_k, W_v, and W_o. We consider $\mathbf{W}$ and the rank r of LoRA as hyperparameters for our study. All learnable parts are optimized by minimizing the joint loss:

$$\mathcal{L} = \mathcal{L}^c + \mathcal{L}^p. \tag{9}$$

Algorithm 1 Memory-Enhanced Predictor

Input: Memory size L, Learned InterCLIP model, classification module $\mathcal{F}_c$ and projection module $\mathcal{F}_p$
Output: Final prediction $\hat{y}^p$

Initialize memory $\mathcal{M} \in \mathbf{0}^{2\times L\times d_f}$
Initialize index $\mathcal{I} \in \mathbf{0}^{2}$
Initialize entropy records $\mathcal{C} \in \mathbf{0}^{2\times L}$
for $i \leftarrow 1$ **to** N_{test} **do**
  $\tilde{h}_i^f \leftarrow \text{InterCLIP}(\mathcal{P}_i)$
  $\hat{y}_i \leftarrow \text{softmax}(\mathcal{F}_c(\tilde{h}_i^f))$
  $\ell_{\text{pse}_i} \leftarrow \arg\max_j(\hat{y}_{i,j}),\ j \in \{0,1\}$
  $c_i \leftarrow -\hat{y}_{i,0}\log\hat{y}_{i,0} - \hat{y}_{i,1}\log\hat{y}_{i,1}$
  $\hat{h}_i^f \leftarrow \text{norm}(\mathcal{F}_p(\tilde{h}_i^f))$
  if $\mathcal{I}[\ell_{\text{pse}_i}] < L$ **then**
    $\mathcal{M}[\ell_{\text{pse}_i}][\mathcal{I}[\ell_{\text{pse}_i}]] \leftarrow \hat{h}_i^f$
    $\mathcal{C}[\ell_{\text{pse}_i}][\mathcal{I}[\ell_{\text{pse}_i}]] \leftarrow c_i$
    $\mathcal{I}[\ell_{\text{pse}_i}] \leftarrow \mathcal{I}[\ell_{\text{pse}_i}] + 1$
  else
    $j \leftarrow \text{GetMaxIdx}(\mathcal{C}[\ell_{\text{pse}_i}])$
    if $c_i < \mathcal{C}[\ell_{\text{pse}_i}][j]$ **then**
      $\mathcal{M}[\ell_{\text{pse}_i}][j] \leftarrow \hat{h}_i^f$
      $\mathcal{C}[\ell_{\text{pse}_i}][j] \leftarrow c_i$
    end if
  end if
  logits $\leftarrow \left[\sum_{k=0}^{\mathcal{I}[0]}(\hat{h}_i^f\mathcal{M}[0]^T)_k, \sum_{k=0}^{\mathcal{I}[1]}(\hat{h}_i^f\mathcal{M}[1]^T)_k\right]$
  $\hat{y}_i^p \leftarrow \text{softmax}(\text{logits})$
  yield $\hat{y}_i^p$
end for

3.2 Memory-Enhanced Predictor

As depicted in Figure 2 (right), we present the Memory-Enhanced Predictor (MEP) that builds upon the learned InterCLIP, along with the classification module and the projection module, leveraging the valuable historical knowledge of test samples to enhance multi-modal sarcasm detection.

The detailed computational process of MEP is provided in Algorithm 1, where N_{test} denotes the number of test samples. MEP uses the trained InterCLIP to extract fused features of the samples. It utilizes the classification module $\mathcal{F}_c$ to assign a pseudo-label ℓ_{pse_i} to each sample $\mathcal{P}_i$ and the projection module $\mathcal{F}_p$ to obtain the sample's projected feature $\hat{h}_i^f$. To store valuable projected features of test samples as historical knowledge, MEP maintains a dynamic fixed-length dual-channel memory $\mathcal{M} \in \mathcal{R}^{2\times L\times d_f}$, where L is the memory length per channel. The first channel stores projected features of non-sarcastic samples, while the second channel stores those of sarcastic samples. Based on the pseudo-label ℓ_{pse_i}, the appropriate memory channel $\mathcal{M}[\ell_{\text{pse}_i}]$ is selected for updating. If the selected channel has available space, the sample's projected features are added directly, and the prediction entropy is recorded. If the memory is full, the prediction entropy of all samples in the memory is compared with that of the current sample. Samples with the highest entropy are replaced, ensuring the retained samples have lower entropy. Finally, the current sample's projected feature is combined with the historical features stored in both memory channels $\mathcal{M}$ using cosine similarity to yield the final prediction.

MMSD/MMSD2.0	Sarcastic	Non-sarcastic	All
Train	8,642/9,576	11,174/10,240	19,816/19,816
Validation	959/1,042	1,451/1,368	2,410/2,410
Test	959/1,037	1,450/1,372	2,409/2,409

Table 1: Statistics of MMSD and MMSD2.0.

4 Experiment

4.1 Experimental Settings

Datasets and metrics. Following Qin et al. (2023), we evaluate performance on MMSD (Cai et al., 2019) and MMSD2.0 (Qin et al., 2023) using accuracy (Acc.), precision (P), recall (R), and F1-score (F1) as metrics. We present the statistics of the two datasets in Table 1.

Baselines. We compare the effectiveness of the InterCLIP-MEP framework against several uni-modal and multi-modal methods. For text modality methods, we compare with TextCNN (Kim, 2014), Bi-LSTM (Zhou et al., 2016), SMSD (Xiong et al., 2019), and RoBERTa (Liu et al., 2019). For image modality methods, we compare with ResNet (He et al., 2016) and ViT (Dosovitskiy et al., 2021). We compare with state-of-the-art multi-modal methods, including HFM (Cai et al., 2019), Att-BERT (Pan et al., 2020), CMGCN (Liang et al., 2022), HKE (Liu et al., 2022), DIP (Wen et al., 2023), DynRT (Tian et al., 2023), Multi-view CLIP (Qin et al., 2023), and G^{2}SAM (Wei et al., 2024), which employ various techniques such as hierarchical fusion, graph neural networks, and dynamic routing for multi-modal sarcasm detection.

4.2 Main Results

To validate the effectiveness of our InterCLIP-MEP framework, we conduct experiments using the original CLIP as the backbone instead of InterCLIP, referred to as `w/o Inter`. We compare this configuration with three interaction modes of InterCLIP: `w/ V2T`, `w/ T2V`, and `w/ TW`.

For each experiment, we condition only the top four layers of the self-attention modules, with the projection dimension d_f set to 1024. We set the

Method	MMSD2.0				MMSD			
	Acc. (%)	F1 (%)	P (%)	R (%)	Acc. (%)	F1 (%)	P (%)	R (%)
Text								
TextCNN (Kim, 2014)	71.61*	69.52*	64.62*	75.22*	80.03*	75.32*	74.29*	76.39*
Bi-LSTM (Zhou et al., 2016)	72.48*	68.05*	68.02*	68.08*	81.90*	77.53*	76.66*	78.42*
SMSD (Xiong et al., 2019)	73.56*	69.97*	68.45*	71.55*	80.90*	75.82*	76.46*	75.18*
RoBERTa (Liu et al., 2019)	79.66*	76.21*	76.74*	75.70*	**93.97***	**92.45***	**90.39***	**94.59***
Image								
ResNet (He et al., 2016)	65.50*	57.58*	61.17*	54.39*	64.76*	61.53*	54.41*	70.80*
ViT (Dosovitskiy et al., 2021)	72.02*	69.72*	65.26*	74.83*	67.83*	63.40*	57.93*	70.07*
Text-Image								
HFM (Cai et al., 2019)	70.57*	66.88*	64.84*	69.05*	83.44*	80.18*	76.57*	84.15*
Att-BERT (Pan et al., 2020)	80.03*	77.04*	76.28*	77.82*	86.05*	82.92*	80.87*	85.08*
CMGCN (Liang et al., 2022)	79.83*	76.90*	75.82*	78.01*	86.54*	84.09*	-*	-*
HKE (Liu et al., 2022)	76.50*	72.25*	73.48*	71.07*	87.36*	72.25*	81.84*	86.48*
DIP (Wen et al., 2023)	80.59†	78.23†	75.52†	81.14†	89.59†	87.17†	87.76†	86.58†
DynRT (Tian et al., 2023)	70.37†	68.55†	63.02†	75.15†	<u>93.59</u>†	<u>91.93</u>†	<u>90.30</u>†	<u>93.62</u>†
Multi-view CLIP (Qin et al., 2023)	<u>85.64</u>*	<u>84.10</u>*	<u>80.33</u>*	<u>88.24</u>*	88.33*	85.55*	82.66*	88.65*
G^{2}SAM (Wei et al., 2024)	79.43†	78.07†	72.04†	85.20†	90.48†	88.48†	87.95†	89.02†
InterCLIP-MEP (Ours)								
w/o Inter ($L_2 = 1024, L_1 = 1280$)	**86.05**	**84.81**	79.83	**90.45**	88.75	86.31	83.73	89.05
w/ TW ($L_2 = 128, L_1 = 1152$)	85.51	**84.26**	79.15	**90.07**	88.54	86.32	82.25	90.82
w/ V2T ($L_2 = 640, L_1 = 1024$)	**86.26**	**85.00**	80.17	**90.45**	88.92	86.66	83.21	90.41
w/ T2V ($L_2 = 1024, L_1 = 1152$)	**86.72**	**85.61**	80.20	**91.80**	88.83	86.37	84.02	88.84
InterCLIP-MEP w/ RoBERTa (Ours)								
w/o Inter ($L_2 = 640, L_1 = 128$)	77.21	75.55	70.20	81.77	**93.94**	**92.54**	**90.77**	**94.37**
w/ TW ($L_2 = 896, L_1 = 256$)	81.98	80.78	74.69	87.95	**93.73**	**92.28**	**90.48**	**94.16**
w/ V2T ($L_2 = 640, L_1 = 512$)	76.96	75.26	69.98	81.39	**93.94**	**92.54**	**90.69**	**94.47**
w/ T2V ($L_2 = 1024, L_1 = 384$)	82.81	81.55	75.81	88.24	**93.73**	**92.28**	**90.56**	**94.06**

Table 2: Main results. We use $*$ to indicate that the results are taken from Qin et al. (2023). - indicates that results are not reported. † indicates our reproduced results. <u>Underlined</u> values represent the best multi-modal baseline for comparison. **Bold** values indicate those that surpass the underlined baseline. L_2 for MMSD2.0 and L_1 for MMSD denote the optimal MEP memory sizes.

LoRA rank r to 8, fine-tuning the self-attention weight matrices $\mathbf{W}$, specifically W_k, W_v, and W_o. For the memory size L, we select the optimal size from a fixed set of candidate values $\mathbf{L}$ [2]. The main results are shown in Table 2.

Performance on MMSD2.0. For MMSD2.0, our framework consistently outperforms or matches the performance of state-of-the-art methods, whether using InterCLIP or the original CLIP as the backbone, as shown in Table 2 (*InterCLIP-MEP*). This demonstrates the effectiveness of our training strategy and MEP. Our results show that `w/ V2T` and `w/ T2V` outperform `w/o Inter`, demonstrating that InterCLIP captures text-image interactions more effectively. Furthermore, `w/ T2V` achieves superior performance compared to `w/ V2T`, likely due to the inherent complexity of the visual space, which presents challenges for the projection layer when mapping vision representations into the text encoder space. In contrast, `w/ TW` performs worse than other configurations, possibly because embedding representations within both encoders increases the learning difficulty. In summary, InterCLIP with T2V interaction, combined with our training strategy and MEP, delivers the most promising results. These findings underscore the robustness and adaptability of our framework, establishing it as a highly effective solution for capturing nuanced text-image interactions and addressing the complexities of multi-modal sarcasm detection.

Performance on MMSD. For MMSD, the RoBERTa-based text modality baseline significantly outperforms other methods due to spurious cues in the text, enabling accurate pre-

[2] The fixed candidate values are {128, 256, 384, 512, 640, 768, 896, 1024, 1152, 1280}.

Method	Accuracy (%)	Trainable Parameters (M)	Fitting Time / Epoch (s)	Inference Time (s)	GPU Memory Peak (GB)
Multi-view CLIP	85.64	165	488	51	15.59
DIP	80.59	196	OOM	OOM	OOM
G2SAM	79.43	116	90	13	18.32
DynRT	70.37	25	370	26	8.03
InterCLIP-MEP	86.72	8	55	6	6.14

Table 3: Efficiency comparison of different methods. To demonstrate the efficiency of InterCLIP-MEP, we selected several recent baselines for comparison. The analysis was conducted using the MMSD2.0 dataset on a single NVIDIA RTX 4090 GPU with a batch size of 128. In the table, `Fitting Time / Epoch` indicates the time required for each epoch during training and validation and `OOM` indicates Out of Memory, referring to GPU memory overflow.

	w/o Inter		*w/ TW*		*w/ V2T*		*w/ T2V*	
Variant	Acc. (%)	F1 (%)	Acc. (%)	F1 (%)	Acc. (%)	F1 (%)	Acc. (%)	F1 (%)
BASELINE	**86.05**	**84.81**	**85.51**	**84.26**	**86.26**	**85.00**	**86.72**	**85.61**
w/o Proj	85.76	84.43	85.43	84.05	85.68	84.22	86.22	84.51
w/o MEP	85.39	83.99	85.22	83.79	86.26	84.78	86.26	84.82
w/o LoRA	82.44	77.73	76.42	74.37	73.31	72.22	75.13	71.79

Table 4: Ablation study of InterCLIP-MEP, with BASELINE denoting results without ablation.

dictions solely based on textual features (Qin et al., 2023). Consequently, models like DynRT, G^2SAM, and DIP, which utilize RoBERTa or BERT for text feature extraction, achieve high performance on MMSD but experience a significant drop on MMSD2.0. To further investigate, we conduct an additional experiment, *InterCLIP-MEP w/ RoBERTa*, replacing the original text encoder with RoBERTa. While this change led to state-of-the-art performance on MMSD, it disrupted InterCLIP's modality alignment, causing a reasonable performance drop on MMSD2.0. This suggests that MMSD's text data contains spurious cues that allow models to rely heavily on the text encoder, while MMSD2.0, having been cleaned, requires more robust multi-modal capabilities. We further find that the `w/ V2T` variant consistently outperforms others in both *InterCLIP-MEP* and *InterCLIP-MEP w/ RoBERTa* experiments, underscoring the model's tendency to overly depend on text modality.

Efficiency Comparison. Our training strategy demonstrates both remarkable effectiveness and outstanding efficiency. To validate this, we conducted a comprehensive comparative analysis against leading state-of-the-art methods, as detailed in Table 3. For instance, the Multi-view CLIP method (Qin et al., 2023) employs a multi-layer Transformer encoder for feature fusion, which, while effective, introduces a significant number of trainable parameters. This results in slower training and inference speeds and greater memory consumption. Similarly, the DIP method (Wen et al., 2023) caches historical samples during training, which hinders its ability to support large-batch training under limited resource conditions. In contrast, our method operates with a batch size of 128 while utilizing a significantly smaller number of trainable parameters, which translates to notably faster training and validation cycles. Furthermore, by incorporating minimal parameter modifications to adapt CLIP and utilizing simple yet effective linear layers for representation fusion, our approach achieves superior inference speeds and drastically reduced memory consumption. These results highlight the practicality of our framework, establishing it as a benchmark for both computational efficiency and performance in multi-modal sarcasm detection.

4.3 Analysis of InterCLIP-MEP

To robustly validate the effectiveness of InterCLIP-MEP, we conduct comprehensive ablation studies and case studies on the more reliable MMSD2.0 benchmark, offering deeper insights into its design and performance. In addition, we include visualization analyses to provide an intuitive understanding of how the framework processes multi-modal sarcasm cues.

Ablation study. We remove the projection module $\mathcal{F}_p$ and train only the classification module $\mathcal{F}_c$ for prediction, denoted as `w/o Proj`. To test the necessity of using LoRA (Hu et al., 2022) for fine-tuning, we keep the rest of InterCLIP-MEP un-

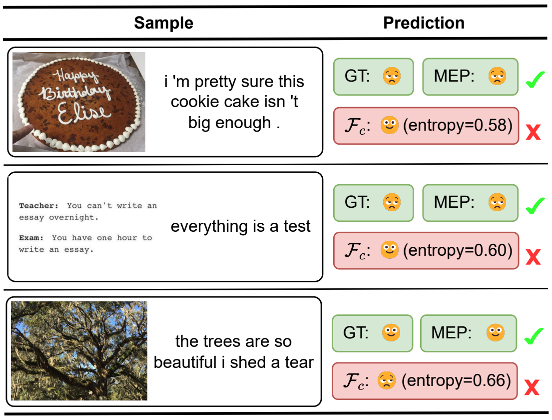

Figure 4: Case study of InterCLIP-MEP. In the figure, *GT* represents the labels annotated by human experts, *MEP* represents the labels predicted by the memory-enhanced predictor, and $\mathcal{F}_c$ represents the labels provided by the classification module.

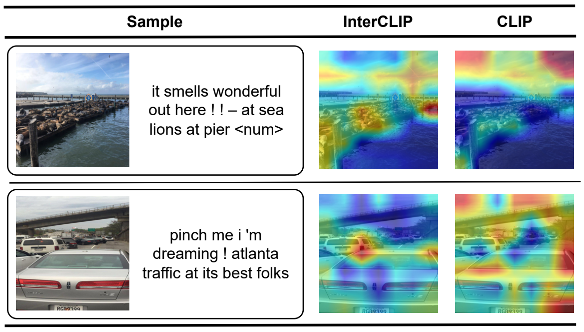

Figure 5: Visual comparison between InterCLIP and original CLIP. Both InterCLIP and the original CLIP were fine-tuned using the same training set and identical parameters. The key distinction is that the original CLIP does not incorporate interaction.

changed and freeze all self-attention weights of InterCLIP, denoted as `w/o LoRA`, always selecting the optimal memory size L for the MEP during inference. To evaluate the effectiveness of the MEP, we train both $\mathcal{F}_p$ and $\mathcal{F}_c$ but use only $\mathcal{F}_c$ during inference, denoted as `w/o MEP`.

Table 4 reports all results. All variants show performance declines compared to the baseline, demonstrating the importance of each module in InterCLIP-MEP. For `w/ TW` and `w/o Inter`, the `w/o MEP` variant performed worse than the `w/o Proj` variant. However, for `w/ T2V` and `w/ V2T`, the `w/o MEP` variant performs better than the `w/o Proj` variant. This suggests that backbones with strong image-text interaction capabilities benefit from training the classification module $\mathcal{F}_c$ along with the projection module $\mathcal{F}_p$, even without using MEP during inference. We also find that not using LoRA to fine-tune the self-attention modules results in significant performance loss, indicating that the original CLIP's vanilla space is not directly suitable for the sarcasm detection task.

Case study. As shown in Figure 4, we select three examples to further demonstrate the robustness of InterCLIP-MEP. We observe that direct predictions through the classification module $\mathcal{F}_c$ result in high prediction entropy and incorrect outcomes. However, MEP effectively mitigates the issue for cases where $\mathcal{F}_c$ fails to correctly identify the results by using the historical knowledge of the test samples. This integration ensures that even in complex situations, the model maintains a high level of accuracy.

Visulization. To further validate that InterCLIP is capable of more effectively capturing the interactive information between text and images compared to original CLIP, thereby aiding in the detection of sarcasm cues, we use GradCAM (Selvaraju et al., 2017) to visualize the areas of focus during the inference process of the visual model in Figure 5. We observe that in the first example, which complains about the *unpleasant odor caused by the sea lions*, InterCLIP focuses more accurately on the location of the sea lions compared to the original CLIP. In the second example, which complains about *traffic congestion*, InterCLIP correctly focuses on the distribution of cars on the road, whereas the original CLIP's focus is scattered.

5 Conclusion

In this paper, we propose InterCLIP-MEP, a novel framework for multi-modal sarcasm detection that directly addresses the challenges of modeling nuanced text-image interactions and managing prediction uncertainty. We design Interactive CLIP (InterCLIP) to embed cross-modal information within text and image encoders, enabling a deeper understanding of sarcasm cues. Additionally, we develop a Memory-Enhanced Predictor (MEP) to dynamically leverage historical sample knowledge, making our inference process more robust and adaptive. Through extensive experiments on MMSD and MMSD2.0 benchmarks, we demonstrate that InterCLIP-MEP achieves state-of-the-art performance while significantly reducing computational costs. By requiring fewer trainable parameters and less GPU memory, our method offers a lightweight, efficient, and scalable solution, setting a new benchmark for multi-modal sarcasm detection.

Limitations

While InterCLIP-MEP delivers strong performance, there remain areas for further refinement. For instance, the framework could benefit from additional techniques to better capture sarcasm that heavily relies on subtle cultural or highly specific contextual cues. Moreover, extending its application to more diverse and less structured datasets could be an interesting direction for future work, further broadening its practical applicability.

Ethical Considerations

This work focuses on advancing multi-modal sarcasm detection to improve understanding of complex communication in online content. While the proposed framework enhances detection accuracy and efficiency, potential misuse must be considered. Automated sarcasm detection could inadvertently amplify biases present in training data or be deployed for unethical purposes, such as targeted content moderation or surveillance. To mitigate these risks, we encourage the responsible use of this technology and emphasize the importance of using diverse and unbiased datasets during training to minimize unintended consequences. Furthermore, this research strictly adheres to ethical guidelines for data collection and usage.

References

Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 167–177, Berlin, Germany. Association for Computational Linguistics.

Alan D. Baddeley. 2000. The episodic buffer: a new component of working memory? *Trends in Cognitive Sciences*, 4:417–423.

Christos Baziotis, Athanasiou Nikolaos, Pinelopi Papalampidi, Athanasia Kolovou, Georgios Paraskevopoulos, Nikolaos Ellinas, and Alexandros Potamianos. 2018. NTUA-SLP at SemEval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive RNNs. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 613–621, New Orleans, Louisiana. Association for Computational Linguistics.

Mondher Bouazizi and Tomoaki Ohtsuki. 2015. Sarcasm detection in twitter:" all your products are incredibly amazing!!!"-are they really? In *2015 IEEE global communications conference (GLOBECOM)*, pages 1–6. IEEE.

Yitao Cai, Huiyu Cai, and Xiaojun Wan. 2019. Multi-modal sarcasm detection in twitter with hierarchical fusion model. In *Annual Meeting of the Association for Computational Linguistics*.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *North American Chapter of the Association for Computational Linguistics*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Hang Du, Guoshun Nan, Sicheng Zhang, Binzhu Xie, Junrui Xu, Hehe Fan, Qimei Cui, Xiaofeng Tao, and Xudong Jiang. 2024. Docmsu: A comprehensive benchmark for document-level multimodal sarcasm understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17933–17941.

Roy Ganz, Yair Kittenplon, Aviad Aberdam, Elad Ben Avraham, Oren Nuriel, Shai Mazor, and Ron Litman. 2024. Question aware vision transformer for multimodal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13861–13871.

Raymond W. Gibbs and Jennifer O'Brien. 1991. Psychological aspects of irony understanding. *Journal of Pragmatics*, 16(6):523–530.

R.W. Gibbs and H.L. Colston. 2007. *Irony in Language and Thought: A Cognitive Science Reader*. Lawrence Erlbaum Associates.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

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

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *International conference on machine learning*, pages 5583–5594. PMLR.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Jiahao Li, Greg Shakhnarovich, and Raymond A. Yeh. 2022. Adapting clip for phrase localization without further training. *arXiv preprint arXiv: 2204.03647*.

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 *Annual Meeting of the Association for Computational Linguistics*.

Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. 2023. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7061–7070.

Hui Liu, Wenya Wang, and Hao Li. 2022. Towards multi-modal sarcasm detection via hierarchical congruity modeling with knowledge enhancement. In *Conference on Empirical Methods in Natural Language Processing*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv: 1907.11692*.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *International Conference on Learning Representations*.

D. C. Muecke. 1982. *Irony and the Ironic*. Methuen, New York.

Hongliang Pan, Zheng Lin, Peng Fu, Yatao Qi, and Weiping Wang. 2020. 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.

Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in information retrieval*, 2(1–2):1–135.

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.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626.

Mark G Stokes. 2015. 'activity-silent'working memory in prefrontal cortex: a dynamic coding framework. *Trends in cognitive sciences*, 19(7):394–405.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. *Advances in neural information processing systems*, 28.

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, Mexico City, Mexico. Association for Computational Linguistics.

Yuan Tian, Nan Xu, Ruike Zhang, and Wenji Mao. 2023. Dynamic routing transformer network for multimodal sarcasm detection. In *Annual Meeting of the Association for Computational Linguistics*.

Oren Tsur, Dmitry Davidov, and Ari Rappoport. 2010. Icwsm — a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. *Proceedings of the International AAAI Conference on Web and Social Media*, 4(1):162–169.

Qiang Wang, Junlong Du, Ke Yan, and Shouhong Ding. 2023. Seeing in flowing: Adapting clip for action recognition with motion prompts learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 5339–5347.

Yiwei Wei, Shaozu Yuan, Hengyang Zhou, Longbiao Wang, Zhiling Yan, Ruosong Yang, and Meng Chen.

2024. G2sam: Graph-based global semantic awareness method for multimodal sarcasm detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(8):9151–9159.

Changsong Wen, Guoli Jia, and Jufeng Yang. 2023. Dip: Dual incongruity perceiving network for sarcasm detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2540–2550.

J. Weston, S. Chopra, and Antoine Bordes. 2014. Memory networks. *International Conference on Learning Representations*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742.

Tao Xiong, Peiran Zhang, Hongbo Zhu, and Yihui Yang. 2019. Sarcasm detection with self-matching networks and low-rank bilinear pooling. In *The World Wide Web Conference*, WWW '19, page 2115–2124, New York, NY, USA. Association for Computing Machinery.

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.

Yabin Zhang, Wenjie Zhu, Hui Tang, Zhiyuan Ma, Kaiyang Zhou, and Lei Zhang. 2024. Dual memory networks: A versatile adaptation approach for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, Berlin, Germany. Association for Computational Linguistics.

A Implementation Details

The model training and testing were conducted using PyTorch Lightning[3]. InterCLIP was constructed by leveraging the Transformers library (Wolf et al., 2020). For the MMSD2.0 experiments, the initial weights for InterCLIP are based on clip-vit-base-patch32[4]. For the MMSD experiments, we utilized the roberta-ViT-B-32 model architecture provided by OpenCLIP[5], with the pre-trained checkpoint laion2b_s12b_b32k[6]. Custom scripts were developed to adapt its format to the Transformers library, ensuring compatibility with our framework. The model parameters were optimized using AdamW (Loshchilov and Hutter, 2019), with a learning rate set to 1e-4 for the LoRA fine-tuning modules and 5e-4 for other trainable modules. A cosine annealing scheduler with warmup was employed to dynamically adjust the learning rate, where the warmup steps constituted the first 20% of the total optimization steps, and the minimum learning rate was set to 1% of the initial rate. For the modules $\mathcal{G}_{t/v}$, $\mathcal{F}_{t/v}$, $\mathcal{F}_c$, and $\mathcal{F}_p$, simple multi-layer perceptrons (MLPs) were utilized. The training processing was performed with a batch size of 64 for 3 epochs. All experiments were run on a machine equipped with an NVIDIA RTX 4090 GPU.

B Experimental Details

B.1 Hyperparameter details

We summarize the hyperparameters involved in InterCLIP-MEP and their descriptions in Table 5. The hyperparameter settings for obtaining the main results in the paper are summarized in Table 6. For other baseline methods, we follow the optimal hyperparameter settings they reported.

B.2 Hyperparameter study

We further investigate the method using Interactive-CLIP with T2V interaction as the backbone. Keeping the other hyperparameters constant, we condition different top-n layers of the self-attention modules. We also study the impact of different projection dimensions d_f, different LoRA ranks r, and different memory sizes L on the `w/ T2V`

[3] https://lightning.ai/
[4] https://huggingface.co/openai/clip-vit-base-patch32
[5] https://github.com/mlfoundations/open_clip
[6] https://huggingface.co/laion/CLIP-ViT-B-32-roberta-base-laion2B-s12B-b32k

Parameter	Description
r	The rank of LoRA, determining the dimension of the low-rank update matrices.
$\mathbf{W}$	The weight matrices in the self-attention module fine-tuned using LoRA, specifically targeting combinations of $W_{\{q,k,v,o\}}$.
top-n	The number of top self-attention layers conditioned during fine-tuning.
d_f	The dimensionality of the latent space for the projected features.
$\mathbf{L}$	The configurable range of memory sizes maintained by the Memory-Enhanced Predictor (MEP).

Table 5: Summary of hyperparameters.

Param.	Value
For trainer	
epoch	3
batch_size	64
lr	5e-4
lora_lr	1e-4
warmup_ratio	0.2
min_lr_rate	0.01
For our model	
r	8
top-n	4
d_f	1024
$\mathbf{W}$	W_k, W_v, W_o
$\mathbf{L}$	{128, 256, 384, 512, 640, 768, 896, 1024, 1152, 1280}

Table 6: Hyperparameter settings.

method. We present all results in Figure 6. Figure 6(a) shows that conditioning the top four self-attention layers yields the best results. From Figure 6(b), a rank of 8 is optimal. Figure 6(c) indicates the projection dimension is best at 256 or 1024. Figure 6(d) reveals that a memory size of 640 in MEP outperforms the others, confirming the value of historical sample knowledge.

B.3 Empirical study of self-attention fine-tuning and interaction modes

Keeping the other hyperparameters constant as shown in Table 6, we fine-tune all possible weight matrices $\mathbf{W}$ and employ different interaction modes of InterCLIP as the backbone. We consistently select the optimal memory size from $\mathbf{L}$ for MEP. We calculate the average metrics for different methods and different weight matrices. The results are presented in Table 7 and Table 8. We observe that fine-tuning the weight matrices W_k, W_v, W_o and using the T2V interaction mode of InterCLIP are the best choices for InterCLIP-MEP.

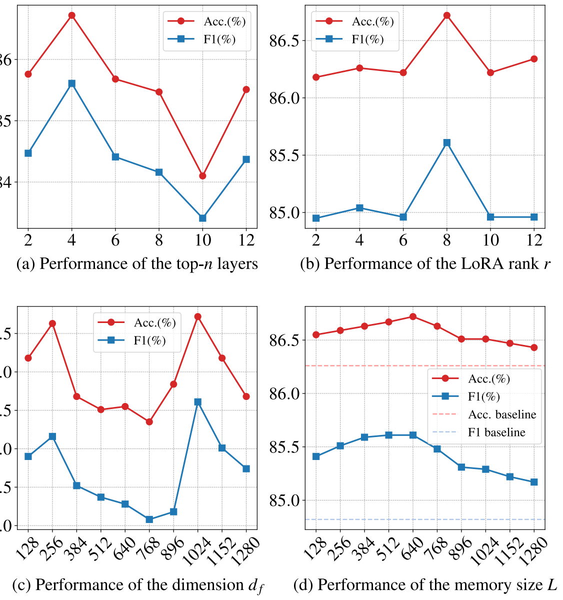

Figure 6: Hyperparameter study curves for `w/ T2V`. Panel (d) compares results with those from using only the classification module $\mathcal{F}_c$ for prediction.

C More Visual Examples

Figure 7 presents additional examples illustrating the focus differences between InterCLIP and CLIP. These visualizations highlight InterCLIP's improved ability to capture sarcasm-related cues by focusing on relevant areas in the images.

D Extended Experiments

To further verify the performance of our framework, we conducted an extended set of experiments.

D.1 Benchmark

DocMSU (Du et al., 2024) is a recently introduced multi-modal sarcasm benchmark designed specifically for long-text analysis. It facilitates the evaluation of multi-modal sarcasm comprehension as well as detection tasks. In this study, we concentrate on the multi-modal sarcasm detection task. The benchmark statistics can be found in Table 9.

D.2 Baselines

We follow the evaluation protocol of Du et al. (2024). We compare against unimodal baselines: BERT-base (text-only) (Devlin, 2018) and Swin Transformer (image-only) (Liu et al., 2021). For multi-modal approaches, we include CLIP (Radford et al., 2021), Vision-and-Language Transformer (ViLT) (Kim et al., 2021), and CMGCN (Liang et al., 2022). The method proposed by Du et al. (2024) is taken as the state-of-the-art baseline.

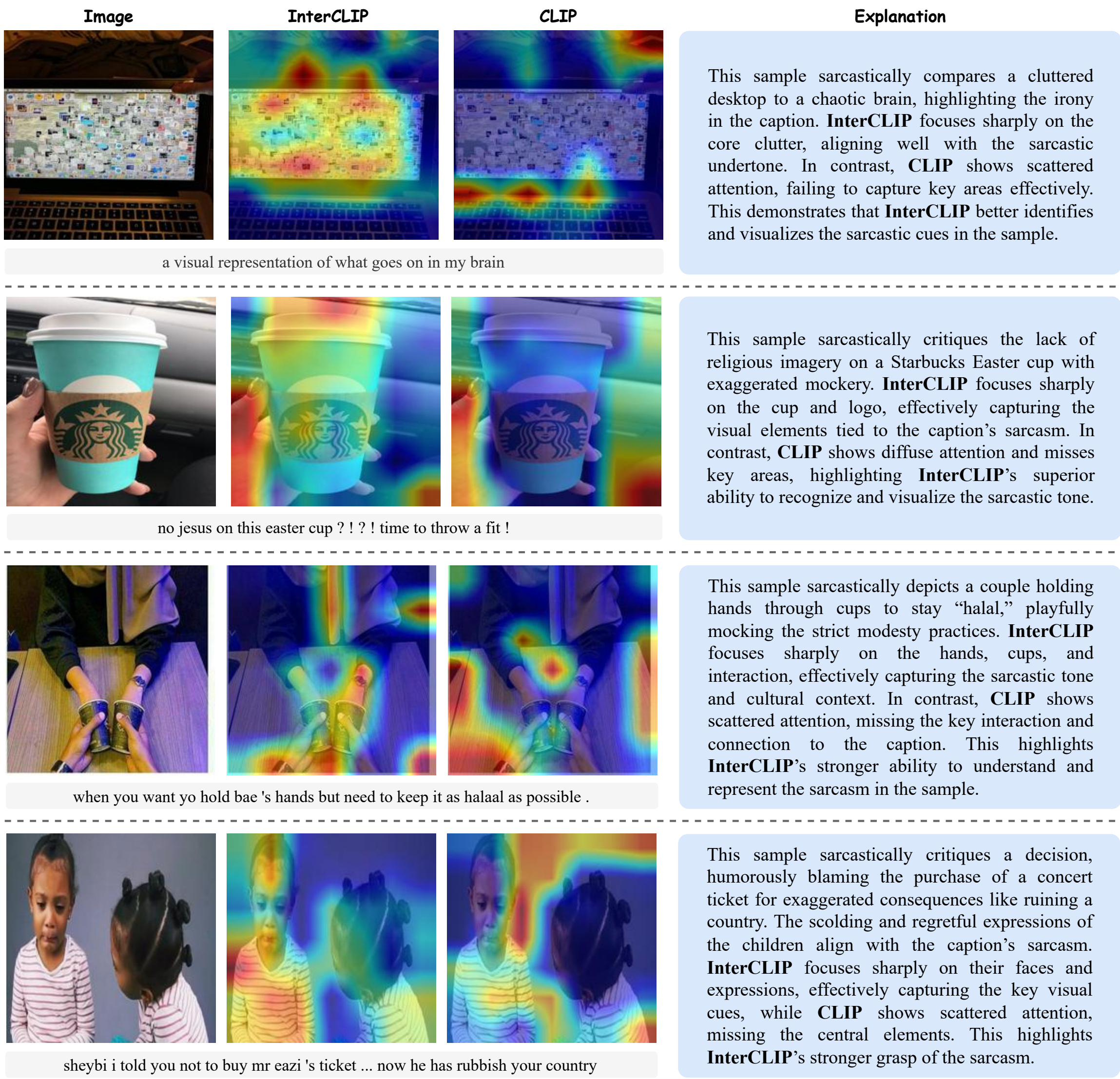

Figure 7: Additional visual examples showcasing InterCLIP's improved focus on sarcasm-related visual cues compared to CLIP.

D.3 Results

Our InterCLIP-MEP framework demonstrates strong performance across various configurations, as shown in Table 10. In particular, the `w/o Inter` and `w/ V2T` variants consistently achieve higher F1 scores compared to the baselines, thereby showcasing their robustness in handling multi-modal sarcasm detection tasks. Notably, all variants achieve nearly perfect recall, highlighting their outstanding capability in accurately identifying sarcasm across diverse datasets. The `w/o Inter` variant also achieves the highest accuracy, further demonstrating its effectiveness and precision.

Overall, the comprehensive results in Table 10 affirm the unparalleled effectiveness of InterCLIP's modality interaction mechanism and our proposed memory-enhanced predictor (MEP), particularly in surpassing existing state-of-the-art methods in both accuracy and F1 score, thus setting a new benchmark in the field.

E Other Related Works

E.1 CLIP adaptation

The Contrastive Language-Image Pretraining (CLIP) model (Radford et al., 2021) excels in vision-language tasks. Adapting CLIP for specific domains has shown substantial improvements, as demonstrated by Li et al. (2022) for phrase localization, Liang et al. (2023) for open-vocabulary semantic segmentation, and Wang et al. (2023) for action recognition. In this work, inspired by Ganz et al. (2024), we conditionally enhance both the text and vision encoders of CLIP, making it more

$\mathbf{W}$	Mean Acc. (%)	Mean F1 (%)
W_q	85.14	83.99
W_k	85.09	84.00
W_v	85.34	84.24
W_o	85.39	84.23
W_q, W_k	85.36	84.16
W_q, W_v	85.63	84.40
W_q, W_o	85.73	84.49
W_k, W_o	85.67	84.43
W_v, W_o	85.73	84.54
W_k, W_v	85.70	84.51
W_q, W_k, W_o	85.92	84.63
W_q, W_v, W_o	85.87	84.60
W_q, W_k, W_v	86.05	84.75
W_k, W_v, W_o	**86.14**	**84.92**
W_q, W_k, W_v, W_o	85.82	84.55

Table 7: Average results of fine-tuning different weight matrices $\mathbf{W}$ across four baseline methods.

Method	Mean Acc. (%)	Mean F1 (%)
w/o Inter	85.49	84.32
w/ TW	85.66	84.44
w/ V2T	85.62	84.41
w/ T2V	**85.78**	**84.55**

Table 8: Average results of four baseline methods for fine-tuning different weight matrices $\mathbf{W}$.

effective in capturing the interplay between text and images to identify multi-modal sarcasm cues. Unlike Ganz et al. (2024), who focused solely on embedding text representations into the vision encoder, we also explore embedding image representations into the text encoder. Furthermore, their approach is limited to general classification tasks and does not address the complexities of multi-modal sarcasm detection.

E.2 Memory-enhanced prediction

Inspired by cognitive science (Stokes, 2015; Baddeley, 2000), memory has been introduced to enhance neural networks (Weston et al., 2014; Sukhbaatar et al., 2015). Several studies (Wu et al., 2018; Wen et al., 2023) have used memory mechanisms to improve model training, and some (Zhang et al., 2024; Wei et al., 2024) leverage memory to store historical knowledge, enhancing prediction accuracy. In this work, we introduce a memory-enhanced predictor for multi-modal sarcasm detection. In contrast to other methods, our memory dynamically updates during testing, utilizing relevant historical information for improved accuracy and robustness.

	Sarcastic	Non-sarcastic	All
Train	4,014	46,265	50,279
Validation	1,125	13,097	14,222
Test	555	6,772	7,327

Table 9: Statistics of DocMSU.

Method	Acc.	F1	P	R
BERT-base*	87.12	86.51	77.61	70.37
Swin-Transformer*	74.83	61.51	67.57	56.45
CMGCN*	88.12	75.23	78.11	72.55
CLIP*	96.19	77.62	78.99	76.30
ViLT*	93.15	41.44	69.03	29.61
Du et al. (2024)*	<u>97.83</u>	<u>87.25</u>	<u>81.20</u>	<u>94.27</u>
InterCLIP-MEP (Ours)				
w/o Inter	**97.84**	**87.48**	78.08	**99.45**
w/ TW	97.79	87.24	77.48	**99.81**
w/ V2T	97.83	**87.45**	77.81	**99.82**
w/ T2V	97.67	86.65	76.45	**99.99**

Table 10: Results of the extended experiments. <u>Underline</u> results denote the compared SOTA baseline, **boldface** highlights results that surpass the baseline, and * indicates results sourced from Du et al. (2024).

F List of Symbols

In Table 11, we have listed the main symbols used in the paper and their descriptions.

Symbol	Description
T	T denotes a short text.
I	I represents an image.
$\mathcal{P}$	$\mathcal{P}$ denotes a text-image pair (T, I).
$\mathcal{T}$	$\mathcal{T}$ denotes CLIP's text encoder.
$\mathcal{V}$	$\mathcal{V}$ denotes CLIP's vision encoder.
$\mathbf{F}$	$\mathbf{F}$ represents the final layer representations encoded by either the text or vision encoder, with text representations as $\mathbf{F}_t$ and image representations as $\mathbf{F}_v$.
$\tilde{\mathbf{F}}$	$\tilde{\mathbf{F}}$ represents the final layer representations encoded by either the text or vision encoder after embedding representations from another modality, with text representations as $\tilde{\mathbf{F}}_t$ and image representations as $\tilde{\mathbf{F}}_v$.
$\mathbf{H}$	$\mathbf{H}$ represents the input representations for each sub-attention layer in the text or vision encoders. Each layer's input comes from the output of the previous layer, denoted $\mathbf{H}_t$ for the text encoder and $\mathbf{H}_v$ for the vision encoder.
$\mathcal{F}_{t/v}$	$\mathcal{F}_{t/v}$ denotes the adapting projection layer in the text or vision encoders used to project the embedded representations of the other modality into the current encoder space. It is denoted as $\mathcal{F}_t$ in the text encoder and $\mathcal{F}_v$ in the vision encoder.
$\mathbf{F}^{'}$	$\mathbf{F}'$ represents the representations projected into the corresponding encoder space. For example, embedding visual representations $\mathbf{F}_v$ in the text encoder and projecting it through $\mathcal{F}_t$ results in $\mathbf{F}'_v$.
$\mathbf{H}^{'}$	$\mathbf{H}'$ represents the representations after embedding another modality's representations and processing them through a self-attention layer.
$\mathcal{H}_{t/v}$	$\mathcal{H}_{t/v}$ denotes the projection module in the self-attention layer used to transform the output of the self-attention module, denoted $\mathcal{H}_t$ for the text encoder and $\mathcal{H}_v$ for the vision encoder.
$\mathcal{G}_{t/v}$	$\mathcal{G}_{t/v}$ denotes the projection module in the self-attention layer that has embedded representations from another modality, used to jointly transform the output representation in combination with $\mathcal{H}_{t/v}$.
$\mathbf{H}^{''}$	$\mathbf{H}''$ represents the final representations in the self-attention layer.
$\tilde{h}^f$	$\tilde{h}^f$ denotes the final fused feature obtained from a sample.
$\mathcal{F}_c$	$\mathcal{F}_c$ denotes the classification module used to assign pseudo-labels to samples.
$\mathcal{F}_p$	$\mathcal{F}_p$ denotes the projection module used to project samples into a latent space.
$\hat{h}^f$	$\hat{h}^f$ represents the feature of a sample's fused feature after transformation by $\mathcal{F}_p$ and L2 normalization.

Table 11: List of symbols