Contrastive Instruction Fine-Tuning Large Multimodal Model for Hateful Meme Classification

Ming Shan Hee[1*], Zihan Gao[2*], Yinglong Wang[3], Xiangxiang Chu[3], Roy Ka-Wei Lee[1] and Zengchang Qin[2]
[1]Singapore University of Technology and Design
[2]Beihang University
[3]Meituan
mingshan_hee@mymail.sutd.edu.sg

Abstract

Detecting hateful memes requires a model that possesses extensive background knowledge and robust reasoning abilities, especially when the memes contain ambiguous descriptions. Previous research has used large language models (LLMs) and large multimodal models (LMMs) to interpret and categorize these memes. However, distinguishing subtly different hateful and non-hateful memes is still challenging. In recognition of this, our study introduces a unique contrastive instruction fine-tuning approach, InstructMemeCL. This method improves an LMM's ability to discern between memes that have similar visual or textual elements by intensifying its focus on semantic subtleties that separate hateful from non-hateful content. We evaluated our model using AUROC and accuracy metrics on three publicly available hateful meme datasets. The results indicate that our improved LMM more accurately identifies hateful and non-hateful memes, demonstrating superior performance compared to conventional LLMs and LMMs used in similar tasks.

Introduction

Internet memes, a staple in daily social media interactions, often create engaging and emotionally rich conversations. These memes combine images with text to convey messages, typically with humorous or satirical intent. However, they can also be weaponized by malicious actors to spread hate, cleverly disguised as humor, targeting specific groups based on race, ethnicity, or religion. The broad appeal and viral nature of memes exacerbate this issue, as they can be rapidly shared across various platforms, amplifying the spread of such harmful content. Identifying and classifying hateful memes presents an ongoing challenge that requires strong multimodal comprehension capabilities that can interpret and analyze the complex interplay of visual and textual cues.

To combat the issue of hateful memes[1], a range of datasets (Kiela et al. 2020; Pramanick et al. 2021a; Fersini et al. 2022; Gasparini et al. 2021; Hee, Chong, and Lee 2023) have been constructed to foster the development of robust hateful meme detection solutions (Lee et al. 2021; Cao et al. 2022; Pramanick et al. 2021b). These datasets predominantly focus on multimodal classification tasks, where memes are categorized as either "*hateful*" or "*non-hateful*". Effective solutions must address the inherent visual-textual interactions in memes (Kiela et al. 2021). Current models, including fine-tuned visual-language models, struggle in two primary aspects. First, memes often recycle content, such as reusing images with different texts or the same text with different images, leading to varied classifications as Figure 2 (a) and (b) where both memes share the same text but contain different images, resulting in different class labels. Similarly, Figure 2 (c) and (d) share the same image but contain different texts. Second, similar visual elements in different

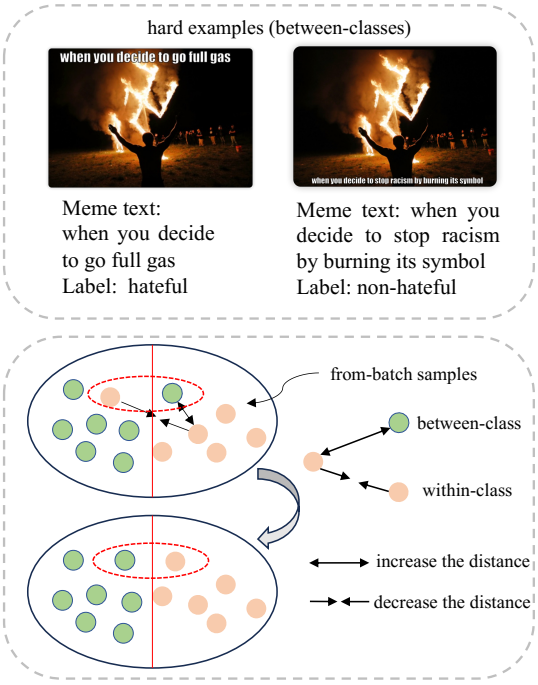

Figure 1: Overview of our proposed approach in differentiating hard examples apart (*between-classes*) and grouping similar cases together (*within-classes*)

*These authors contributed equally to this work

[1]**Disclaimer**: *This paper contains violent and discriminatory content that may be disturbing to some readers.*

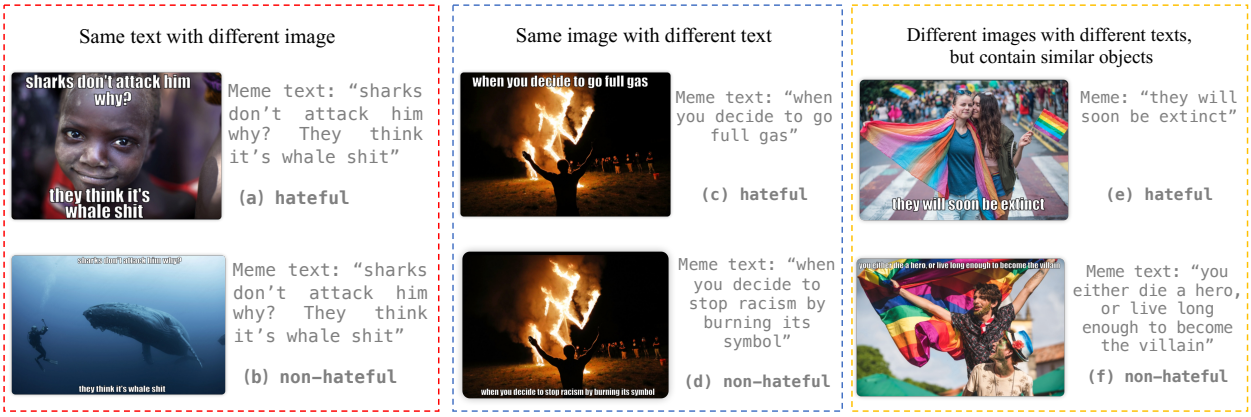

Figure 2: Examples of three hard cases that are difficult to distinguish.

memes can lead to misclassification. For example, the meme in Figure 2 (e) is included in the training set, which results in the misclassification of the meme in Figure 2 (f) from the test set. While these differences can be relatively apparent, the discernible distinctions between modalities pose challenges that stem from limited dataset resources and potential model bias acquired during fine-tuning. These challenges are exacerbated by the limited scale of available datasets, which raises risks of overfitting and impairs the models' generalizability. Furthermore, there's a tendency for these models to over-rely on a single modality (visual or textual), leading to inaccuracies when the other modality varies (Hee, Lee, and Chong 2022; Cuo et al. 2022).

To address these challenges, we introduce InstructMemeCL, a novel contrastive instruction fine-tuning approach using Large Multimodal Models (LMMs). Our method involves creating specific instruction templates, drawing from existing hateful memes datasets, to fine-tune LMMs for a range of downstream tasks. We then train the LMM on these instruction datasets, employing negative log-likelihood loss on targeted logits. A key step is the application of the contrastive loss technique, which helps to distinguish between difficult cases and groups similar instances together. This technique significantly enhances the LMM's capability to generate distinct multimodal semantic representations, crucial for the accurate classification of hateful memes. By effectively increasing the separation between memes with similar content but different labels at a deep representation level, our approach markedly improves the classification accuracy of challenging cases. To the best of our knowledge, this is the first work that fine-tunes LMMs for the hateful meme classification task, setting a benchmark for future research in this area.

We summarize our contributions as follows: (i) We perform the fine-tuning of LMMs for the multimodal classification of hateful memes, achieving state-of-the-art performance across diverse meme datasets. This exploration into applying LMMs in hateful meme classification is, to the best of our knowledge, the first in the field. (ii) We introduce a novel contrastive instruction fine-tuning method[2] specifically designed to address samples that are challenging to differentiate. Our experimental results demonstrate a significant improvement with this method over traditional vanilla instruction fine-tuning. It accomplishes this by more effectively distinguishing between-class representations while simultaneously clustering within-class representations. (iii) Through rigorous evaluation of our model on three widely recognized public datasets, we validate the efficacy of our approach. In addition, our comprehensive fine-grained analyses and case studies in various scenarios shed light on the enhanced capability of contrastive tuning to accurately classify hateful memes. Collectively, these advances not only set a new benchmark in the domain but also opened new avenues for research into the application of LMMs in complex multimodal tasks.

Related Work

Hateful Meme Detection

The proliferation of hateful memes on social media platforms has emerged as a significant challenge, compromising the trust and safety of online users (Hee et al. 2024a,b). To combat this, a variety of datasets have been curated, such as the comprehensive *Facebook Hateful Meme Dataset* released during the *Hateful Memes Challenge* (Kiela et al. 2020) and the more specific collection by Pramanick et al. (2021a) focusing on COVID-19 related memes. More recently, Fersini et al. (2022) launched the *Multimedia Automatic Misogyny Identification* challenge, releasing memes related to misogynistic attacks. The range and diversity of these datasets, which vary in size and theme, underscore the complexity of the hateful meme detection task.

Prevailing studies in the field of hateful meme classification have predominantly focused on two principal methodologies. The first involves *classic two-stream models* (Kiela et al. 2020; Suryawanshi et al. 2020), which independently process the textual and visual components of memes and

[2]https://github.com/Social-AI-Studio/InstructMemeCL

then combine these features using multimodal fusion techniques for classification. The second approach includes the fine-tuning of pre-trained visual language models (Muennighoff 2020; Velioglu and Rose 2020; Pramanick et al. 2021b), which seek to leverage the inherent capabilities of these sophisticated systems.

Recent advancements transform all meme-related content into textual format for processing by language models, as seen in studies like PromptHate (Cao et al. 2022) and Pro-Cap (Cao et al. 2023). While these methods capitalize on the contextual understanding capabilities of language models, they often encounter the challenge of information loss during the conversion of images to text. Pro-Cap, for instance, seeks to mitigate this by employing pre-trained visual language models for extracting more contextually relevant descriptions in a zero-shot manner. Nevertheless, these techniques can still overlook critical nuances inherent in visual data and often require additional preprocessing steps.

Our paper addresses these gaps by fine-tuning LMMs for hateful meme classification. The existing literature has yet to explore large multimodal models (LMMs) extensively, given their recent emergence and high computational requirements. Most prior work has relied on BERT-based models because of their lower resource demands. However, it is important to examine the capabilities and limitations of these newly released state-of-the-art multimodal models for this challenging task. LMMs (Zhu et al. 2023; **?**) integrate pre-trained models across multiple modalities into an end-to-end trainable framework and are pre-trained on large corpora of unlabeled data. These models are then further trained on instruction-tuning datasets, enabling them to understand and perform a wide range of natural language tasks. However, a recent study (Lin et al. 2024) found that LMMs tend to perform poorly on hateful meme detection tasks in zero-shot settings. To better adapt LMMs to the hateful meme classification task, we constructed a specialized instruction dataset that guides the models to follow task-relevant prompts. Given the substantial computational cost of fine-tuning LMMs, we adopt Low-Rank Adaptation (LoRA) (Hu et al. 2021) to improve training efficiency and model performance. Considering the computational cost involved in fine-tuning LMMs, we run our proposed approach on two LMMs for validation and comparisons.

Contrastive Learning for Text Generation

Contrastive learning aims to extract meaningful representations by contrasting positive and negative pairs of instances (Chen et al. 2020). This approach strengthens the feature distinction by drawing matched instances closer and pushing unmatched ones further in a learned representation space. Its efficacy is rooted in the principle that distinguishing between similar and dissimilar instances enhances representation quality. In the realm of computer vision, contrastive learning has notably advanced visual representation learning. Existing studies (Chen et al. 2020; He et al. 2020; Schroff, Kalenichenko, and Philbin 2015) have demonstrated improvements in diverse vision tasks. Beyond vision, this methodology plays a critical role in the pre-training of visual-language models. Pioneering contributions in this area (Radford et al. 2021; Jia et al. 2021; Li et al. 2022) have significantly enriched multimodal learning. Contrastive learning has also increasingly gained attention in natural language processing (Gao, Yao, and Chen 2021; Hjelm et al. 2018; Kong et al. 2020; Krishna et al. 2022), notably by integrating contrastive loss into language model fine-tuning. The COSINE framework (Yu et al. 2020) demonstrates the efficacy of this approach across various classification tasks at the sequence, token, and sentence-pair levels. Additionally, Lee, Lee, and Hwang (2020) pioneered the incorporation of contrastive learning into text generation, addressing the exposure bias issue by proposing an adversarial method to construct more challenging positive-negative samples, including those derived from batch samples. Similarly, SimCTG (Su et al. 2022) and CoNT (An et al. 2022) have extended this framework to token-level and sequence-level contrastive learning, respectively, in text generation.

The use of contrastive learning in LMMs is not entirely new. Several existing studies have applied contrastive learning to address factually inaccurate hallucinations in text generation (Jiang et al. 2024; Sarkar et al. 2024). These methods generate intentionally incorrect textual image captions as hard negatives for contrastive learning, enabling the model to distinguish between hallucinated and accurate information. This approach enhances the separation of embedding representations and establishes clearer decision boundaries. Our work differs from these existing studies that focus on generating synthetic hard negatives to address factually inaccurate hallucinations. Our work aim to understand and distinguish the subtle distinctions between two similar but valid pieces of content that may be assigned opposing labels. In real-world scenarios, two memes may feature the same image but contain different text, leading to different labels—one might be flagged as hateful, while the other is not. This nuanced understanding of valid yet contextually distinct content sets the focus of our work apart from previous studies using contrastive learning. Additionally, our approach reframes the text classification task as a text generation problem and employs LoRA for parameter-efficient fine-tuning of LMMs, achieving better classification performance with minimal parameter updates and storage.

Methodology

In this section, we formally define our problem, transforming the meme classification task as a text generation task for LMMs. Then, we explain our InstructMemeCL approach, which consists of three main steps. First, we design an instruction template for each meme to build a fine-tuning instruction data set. Next, we adopt low-rank adapters (LoRA) (Hu et al. 2021) to efficiently fine-tune the language component of LMMs. Finally, we explain our contrastive instruction fine-tuning strategy, which encourages the model to cluster similar (within-class) samples while separating dissimilar (between-class) ones in the embedding space. Figure 3 provides an overview of the framework.

Problem Formulation

Given a meme with image $\mathcal{I}$ and text $\mathcal{O}$, the task of hateful meme classification is to determine whether the meme

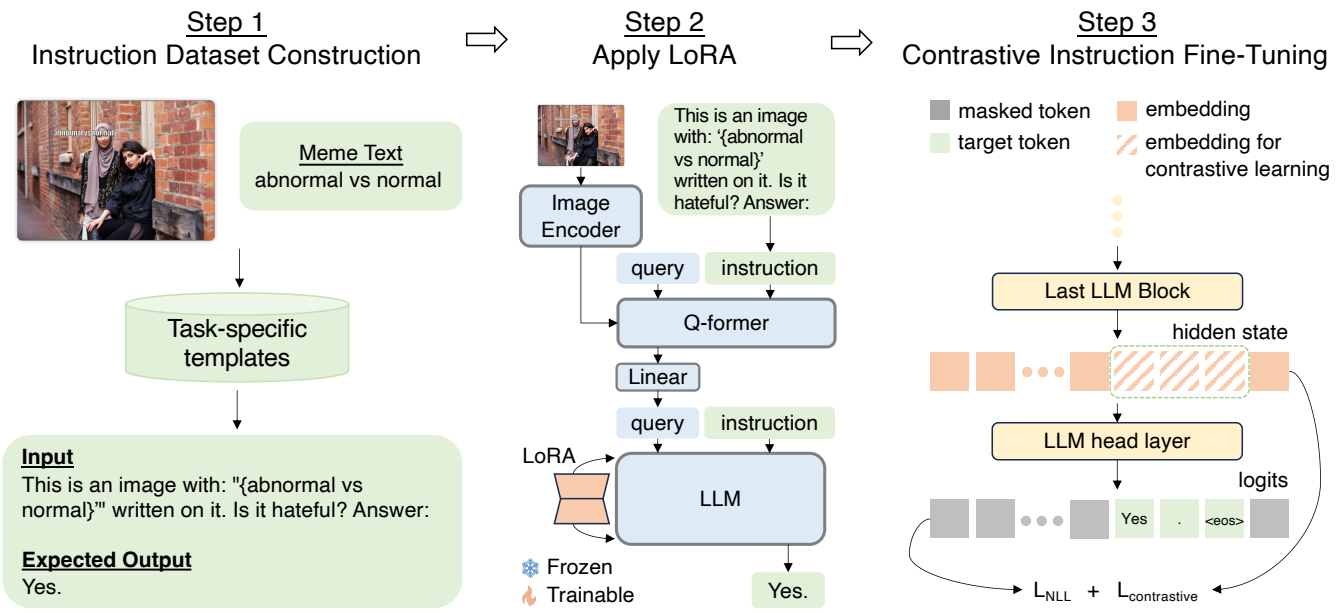

Figure 3: Our proposed IntMemeCL framework on InstructBLIP model. Step 1: Construct the instruction dataset using the task-specific template. Step 2: Apply LoRA to perform parameter-efficient fine-tuning on the LLM component in the InstructBLIP. Step 3: Add contrastive instruction fine-tuning to cluster the representations of within-class samples while distinctly separating those of between-class samples within the embedding space.

is *hateful* or *non-hateful*. Traditionally, this binary classification task is solved by introducing a classifier to predict a probability vector $\mathbf{y} \in \mathbb{R}^2$ over the two classes. As we aim to leverage LMM to perform hateful meme classification, we convert hateful meme classification into a text generation task. In this setup, the LMM is typically trained using the language model objective with the maximum likelihood estimation. Specifically, given a text instruction sequence $\boldsymbol{x_T} = \{x_i\}_{i=0}^{M}$ with the input image $\boldsymbol{x_I}$, we feed them into the LMM, which in turn outputs a text response sequence $\boldsymbol{y} = \{y_i\}_{i=0}^{N}$, where we minimize the following negative log likelihood loss:

$$\mathcal{L}_{\text{NLL}} = -\sum_{t=1}^{N} \log p_\theta(y_t | \boldsymbol{x_I}, \boldsymbol{x_T}, \boldsymbol{y}_{<t}) \quad (1)$$

At the inference stage, we evaluate model performance using the logit vectors of the `[No]` and `[Yes]` tokens. In this context, `No` and `Yes` are responses to the question in the instruction template, representing `Non-Hateful` and `Hateful` classification decisions, respectively.

Instruction Dataset Construction

Recent studies (Zhang et al. 2023) have found that instruction tuning enables the pre-trained LMMs to accurately interpret user instructions by constraining the model's outputs to align with the desired response characteristics or domain knowledge. Notably, this facilitates efficient adaptation to a particular domain without requiring extensive retraining or architectural modifications. As we are instruction-tuning the LMM for hateful meme classification, we transform the original hateful meme datasets into instruction datasets.

We construct instructions in the format of input-output pairs. Given a meme, we first extract the text in the meme using open-source Python packages EasyOCR[3], followed by in-painting with MMEditing[4] to remove the text. Subsequently, we place the extracted text and clean image (denoted as `[text]` and `[image]`, respectively) into a customized instruction template:

Input:

`[image]` This is an image with '{`[text]`}' written on it. Is it hateful? Answer:

Expected Output:

Yes. / No.

Where `[image]` and `[text]` are placeholders for the meme image and text, which will be substituted before being input into the model. The "*Expected Output*" is then determined according to the label of the provided record. An example of an instruction sample is shown in Figure 3.

Parameter-Efficient Fine-Tuning

LoRA adapters enable efficient fine-tuning of large multimodal models (LMMs) by updating only a small set of low-rank matrices, as shown in Equation 2.

$$h = W_0 x + \Delta W x = W_0 x + BAx \quad (2)$$

Where $W_0 \in \mathbb{R}^{d \times k}$ signifies a pre-trained weight matrix, which is frozen during training. ΔW representing the trainable adapter, is decomposed into a low-rank decomposition

[3]https://github.com/JaidedAI/EasyOCR

[4]https://github.com/open-mmlab/mmediting

Datasets	Train		Test	
	#Hate.	#Non-hate.	#Hate.	#Non-hate.
FHM	3,050	5,450	250	250
MAMI	5,000	5,000	500	500
HarM	1,064	1,949	124	230

Table 1: Statistical summary of datasets: FHM, MAMI and HarM.

denoted as $\Delta W = BA$, where $B \in \mathbb{R}^{d\times r}, A \in \mathbb{R}^{r\times k}$, and the rank $r \ll \min(d, k)$. For our implementation, we apply LoRA specifically to the query and value projection layers of each attention block within the LMM, following the recommendations of Hu et al. (Hu et al. 2021).

Contrastive Instruction Fine-tuning

Finally, we propose using contrastive learning to handle the difficult-to-distinguish samples shown in Figure 2. The key component of our contrastive learning method is to learn representations that promote similarity among within-class samples while separating between-class samples. Specifically, we can create a series of example pairs $(\boldsymbol{h}^i, \boldsymbol{h}^j) \in \mathcal{B}$, where i and j are sampled from one batch and $\boldsymbol{h}_{(\cdot)}$ is the output hidden state of the LMM's final decoder layer. For each (h_i, h_j), we define their distance as follows:

$$d_{ij} = 1 - \frac{h_i^T \cdot h_j}{\|h_i\|_2 \|h_j\|_2} \tag{3}$$

We further denote the relations between i and j as R_{ij}:

$$R_{ij} = \begin{cases} 1, & \text{if i and j are samples from the same class} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

Next, we define the contrastive loss between i and j as

$$\ell_{ij} = R_{ij} d_{ij}^2 + (1 - R_{ij})[\max(0, \gamma - d_{ij})]^2 \tag{5}$$

Where γ is a pre-defined margin. During training, we minimize the final contrastive loss:

$$\mathcal{L}_{\text{C}} = \sum_{(h_i, h_j)\in\mathcal{B}\times\mathcal{B}} \ell(h_i, h_j, R_{ij}) \tag{6}$$

Intuitively, training with $\mathcal{L}_{\text{C}}$ encourages the model to minimize the distance between representations of samples from the same class, while maximizing the distance between those from different classes. As a result, our contrastive learning method enables the model to distinguish between samples with similar content but belonging to different classes.

Finally, the overall training objective $\mathcal{L}$ is then defined as

$$\mathcal{L} = \alpha\mathcal{L}_{\text{C}} + \mathcal{L}_{\text{NLL}} \tag{7}$$

where α controls the contribution of the contrastive loss. Note that when the margin γ is set to 0, the overall objective simplifies to the vanilla NLL loss. Both α and γ are treated as hyperparameters.

Experiments

In this section, we introduce the datasets, models, evaluation metrics, and implementation details used in our experiments. We then compare the performance of our InstructMemeCL approach with state-of-the-art models. In addition, we provide visualization analyses and case studies to demonstrate the effectiveness of InstructMemeCL. Finally, we discuss the limitations of the model through error cases analysis.

Experiment Settings

Datasets. We used three publicly available datasets in our experiments: the *Facebook Hateful Meme* dataset (**FHM**) (Kiela et al. 2020), the *Multimedia Automatic Misogyny Identification* dataset (**MAMI**) (Fersini et al. 2022) and the *HarMeme* dataset (**HarM**) (Pramanick et al. 2021a). The FHM dataset was constructed and released by Facebook as part of a challenge to crowd-source multimodal hateful meme classification solutions. The FHM dataset contains hateful memes targeting various vulnerable groups in categories including Religion, Race, Gender, Nationality, and Disability. The memes are labeled with either *hateful* and *non-hateful* class. As we do not have labels of the memes in the test split, we utilize the *dev-seen* split as the *test* split. The MAMI dataset focuses on a particular category of hateful memes, namely, misogyny memes. The memes are labeled with either *misogynous* and *non-misogynous* class. The HarM dataset contains memes related to COVID-19, which are classified into three categories: *harmless*, *partially harmful*, and *very harmful*. We merge *partially harmful* and *very harmful* into one category. Table 1 outlines the statistical distributions of the three datasets.

Models - Baselines. We compare our proposed models with 12 widely used models that have been fine-tuned for the detection of hateful memes. These baseline models span a diverse range of techniques, including both unimodal and multimodal approaches.

For the unimodal models, we consider both a text-only approach and an image-only approach. For the text-only model, we fine-tune a pre-trained BERT model (Devlin et al. 2019) using only the meme text for classification, referring to this model as **Text-BERT**. For the image-only model, we first extract object-level image features using the Faster-RCNN (Ren et al. 2016) feature extractor, which is trained for object detection. We then apply average pooling to the object features and input the resulting vector into a classification layer. This model is referred to as **Image-Region**.

For the multimodal models, we categorize them into two groups: 1) models that fine-tune generic multimodal architectures designed for various multimodal tasks; and 2) models specifically developed for detecting hateful memes. In the first group, we include the **MMBT-Region** model (Kiela et al. 2019), a widely used multimodal baseline in hateful meme detection (Kiela et al. 2020; Pramanick et al. 2021a), which has not been pre-trained with multimodal data. We also consider several pre-trained multimodal models, such as VisualBERT (Li et al. 2019), pre-trained on MS-COCO (Lin et al. 2014) (**VisualBERT COCO**), and ViLBERT (Lu et al.

Model	Components		FHM		MAMI		HarM	
	LoRA.	C.FT	AUC.	Acc.	AUC.	Acc.	AUC.	Acc.
Text BERT	-	-	$66.10_{\pm 0.55}$	$57.12_{\pm 0.49}$	$74.48_{\pm 0.60}$	$67.37_{\pm 0.57}$	$81.39_{\pm 0.91}$	$75.68_{\pm 1.59}$
Image-Region	-	-	$56.69_{\pm 1.05}$	$52.34_{\pm 1.39}$	$70.20_{\pm 0.63}$	$64.18_{\pm 0.81}$	$76.46_{\pm 0.47}$	$73.05_{\pm 1.80}$
VisualBERT COCO	-	-	$68.71_{\pm 1.02}$	$61.48_{\pm 1.19}$	$78.71_{\pm 0.59}$	$71.06_{\pm 0.94}$	$80.46_{\pm 1.04}$	$75.31_{\pm 1.44}$
ViLBERT CC	-	-	$73.05_{\pm 0.62}$	$64.70_{\pm 1.12}$	$77.71_{\pm 1.20}$	$69.48_{\pm 1.00}$	$84.11_{\pm 0.88}$	$78.70_{\pm 1.17}$
MMBT-Region	-	-	$72.86_{\pm 0.64}$	$65.06_{\pm 1.76}$	$79.17_{\pm 0.91}$	$70.46_{\pm 0.76}$	$85.48_{\pm 0.75}$	$79.83_{\pm 2.00}$
CLIP-BERT	-	-	$66.97_{\pm 0.34}$	$58.28_{\pm 0.63}$	$77.66_{\pm 0.64}$	$68.44_{\pm 1.07}$	$82.63_{\pm 3.83}$	$80.48_{\pm 1.95}$
DisMultiHate	-	-	$69.11_{\pm 0.84}$	$62.42_{\pm 0.72}$	$78.21_{\pm 0.61}$	$70.58_{\pm 1.13}$	$83.69_{\pm 1.33}$	$78.05_{\pm 0.73}$
PromptHate	-	-	$76.76_{\pm 0.95}$	$67.82_{\pm 1.23}$	$76.21_{\pm 1.05}$	$68.08_{\pm 0.58}$	$87.51_{\pm 0.74}$	$79.38_{\pm 1.72}$
BLIP	-	-	$76.80_{\pm 2.37}$	$69.20_{\pm 1.84}$	$80.59_{\pm 0.87}$	$71.84_{\pm 1.11}$	$87.09_{\pm 1.46}$	$81.81_{\pm 1.74}$
ALBEF	-	-	$79.40_{\pm 0.53}$	$70.58_{\pm 0.50}$	$83.24_{\pm 0.93}$	$72.77_{\pm 1.00}$	$85.49_{\pm 1.23}$	$80.99_{\pm 0.80}$
Pro-Cap$_{\text{BERT}}$	-	-	$77.50_{\pm 0.58}$	$68.14_{\pm 0.64}$	$79.62_{\pm 0.91}$	$71.06_{\pm 0.88}$	$89.04_{\pm 1.00}$	$82.06_{\pm 1.92}$
Pro-Cap$_{\text{PromptHate}}$	-	-	$80.87_{\pm 0.66}$	$72.28_{\pm 0.90}$	$82.53_{\pm 0.49}$	$73.06_{\pm 0.82}$	$90.25_{\pm 0.54}$	$83.25_{\pm 1.00}$
Inst.Meme$_{\text{InstructBLIP}}$	✓	✗	$83.01_{\pm 0.24}$	$75.58_{\pm 0.44}$	$85.72_{\pm 0.41}$	$77.55_{\pm 0.51}$	$92.03_{\pm 0.10}$	$87.23_{\pm 0.25}$
Inst.MemeCL$_{\text{InstructBLIP}}$	✓	✓	$\mathbf{84.21}_{\pm 0.80}$	$\mathbf{77.42}_{\pm 0.37}$	$\mathbf{85.99}_{\pm 0.27}$	$\underline{\mathbf{78.53}}_{\pm 0.93}$	$\mathbf{92.19}_{\pm 0.20}$	$\underline{\mathbf{87.77}}_{\pm 0.36}$
Inst.Meme$_{\text{Qwen2.5-VL}}$	✓	✗	$87.43_{\pm 0.82}$	$77.56_{\pm 1.07}$	$86.68_{\pm 0.52}$	$77.37_{\pm 1.04}$	$92.14_{\pm 0.18}$	$86.44_{\pm 0.4}$
Inst.MemeCL$_{\text{Qwen2.5-VL}}$	✓	✓	$\underline{\mathbf{87.62}}_{\pm 0.35}$	$\underline{\mathbf{78.92}}_{\pm 1.21}$	$\underline{\mathbf{87.10}}_{\pm 1.11}$	$\mathbf{78.13}_{\pm 0.88}$	$\underline{\mathbf{92.39}}_{\pm 0.5}$	$\mathbf{86.95}_{\pm 0.64}$

Table 2: Comparison of our proposed approach with baseline model performance on three meme datasets. Inst.Meme and Inst.MemeCL refer to InstructMeme and InstructMemeCL, respectively. For each dataset, the better performance between Inst.Meme and Inst.MemeCL is shown in bold, while the best overall performance across all models is underlined.

2019), pre-trained on Conceptual Captions (Sharma et al. 2018) (**ViLBERT CC**). Furthermore, we include more recent powerful models like *Align before Fusion* model (Li et al. 2021) (**ALBEF**) and the *Bootstrapping Language-Image Pre-training* model (Li et al. 2022) (**BLIP**).

In the second group, which includes models specifically designed for meme detection, we consider the following: The **CLIP-BERT** model (Pramanick et al. 2021b), which combines the CLIP model for handling noisy meme images with the pre-trained BERT for text representation, using concatenation to fuse both modalities. The **DisMultiHate** model (Lee et al. 2021) disentangles target information from memes, recognizing that target identification is crucial for detecting hateful content. The **PromptHate** model (Cao et al. 2022) uses a prompt-based approach with few-shot demonstrations to classify memes. Lastly, the **Pro-Cap** model (Cao et al. 2023) prompts a frozen vision language model to generate informative image captions and additional information for the hateful meme detection task.

Models - InstructMemeCL. We evaluated the effectiveness of our InstructMemeCL approach on two open-source LMMs: InstructBLIP[5] (Dai et al. 2023), and Qwen2.5-VL[6] (Bai et al. 2025). Specifically, we fine-tune the LMMs using the instruction fine-tuning solely ("InstructMeme"), evaluating the performance of state-of-the-art open-source LMMs in classifying hateful memes. Subsequently, we use our proposed contrastive instruction fine-tuning approach to better distinguish difficult samples containing similarities but of different classes ("InstructMemeCL").

The two variants enable a fair comparison between instruction-only and contrastive instruction fine-tuning approaches, highlighting the effectiveness of learning better representations and establishing the benefits of using contrastive learning in hateful meme classification. Notably, InstructMemeCL is not applicable to BERT-based models due to their fundamental differences in pre-training and fine-tuning paradigms. Unlike generative LMMs, BERT-based models are not instruction-tuned and lack the autoregressive architecture needed for instruction generation-based setup.

Model Exclusions. We excluded the ensemble methods (Zhu 2020; Muennighoff 2020) to facilitate a fair comparison of the models. These methods combine predictions from multiple individual models, which not only makes the comparison unfair to single-model approaches but also reduces the interpretability of collective decisions.

Evaluation Metrics. We adopt the evaluation metrics commonly used in existing hateful meme classification studies (Kiela et al. 2020; Velioglu and Rose 2020): Area Under the Receiver Operating Characteristic (AUROC) and Accuracy (Acc). In order to report more reliable results, we measure the average performance of models across multiple random seeds. Specifically, we run the baseline model with five random seeds and our proposed model with ten random seeds, allowing for more robust statistical analysis.

Implementation Details. We used a batch size of 16 for all experiments and performed them on a single A100 GPU. Each experiment was run for up to 10 epochs, with early stopping applied using a patience value of 3. The models were trained using the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and a weight decay of 0.05.

Experiment Results

Table 2 shows the performance of all models on the three publicly available datasets. The average performance and standard deviations ($\pm$) of the multiple runs are reported, and the best results are in **bold**.

[5]https://github.com/salesforce/LAVIS/blob/main/projects/instructblip/README.md

[6]https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct

Vanilla Instruction Fine-tuning (Inst.Meme) . Our results reveal that the vanilla instruction fine-tuning approach on both LMMs outperform state-of-the-art baselines in performance across all test datasets. In a specific comparison with the baseline that previously performed the best, ProCap$_{\text{PromptHate}}$, we observed significant enhancements in both the AUROC and the accuracy metrics. The improvements are approximately 2 – 7% in terms of AUROC and 3 – 6% in terms of accuracy, across the various datasets. This data underscores the efficacy of instruction fine-tuning applied to LMM in the context of hateful meme classification. Moreover, our experimental results underscore the stability and generalizability of the LMM instruction fine-tuning method. This is particularly evident in the consistently low standard deviation of our method's performance metrics, as compared to previous methods.

Contrastive Instruction Fine-tuning (Inst.MemeCL) . The contrastive instruction fine-tuning approach demonstrates superior performance over the vanilla instruction fine-tuning method in most scenarios, although the degree of improvement varies between the three datasets. We observe an increase in accuracy of approximately 1.5 - 2% for FHM, 1% for MAMI, and a modest 0.5% for HarM. In contrast, the improvements in AUROC are less pronounced, with gains of approximately 0.2 – 1% for FHM, 0.25 – 0.5% for MAMI, and 0.15 – 0.25% for HarM. These results highlight the effectiveness of our contrastive learning approach in enhancing model performance, not only by improving the correctness of predictions (as reflected in higher accuracy), but also by learning more discriminative representations that lead to better ranking quality (as indicated by AUROC).

The general trend indicates a more significant performance gain in the FHM dataset. This disparity in performance improvements can be attributed to the distinct construction of each dataset. Specifically, the FHM dataset, synthesized with benign confounders, is particularly conducive to the contrastive method. These confounders, by design, manipulate the memes, altering the image or text to change the label from hateful to non-hateful and vice versa. Such deliberate manipulation presents challenging cases for contrastive learning, enabling the model to better distinguish difficult examples in the FHM dataset.

Statistical Analysis of Performance Improvements. The Wilcoxon signed-rank test was used to assess whether the performance improvements observed with the contrastive instruction fine-tuning method are statistically significant compared to the vanilla instruction fine-tuning method. This non-parametric test is particularly well-suited for comparing paired samples, as it evaluates whether the median differences between the two methods are consistently in favor of one approach. In our case, the p-values obtained for both accuracy and AUROC across all three datasets (FHM, MAMI, and HarMeme) and both LMMs are below the commonly accepted threshold of 0.05. Specifically, the p-values for AUROC and accuracy for all datasets fall between 0.002 and 0.0098, confirming that the performance gains achieved by our contrastive method are statistically significant. These results underscore the robustness and effectiveness of the proposed approach.

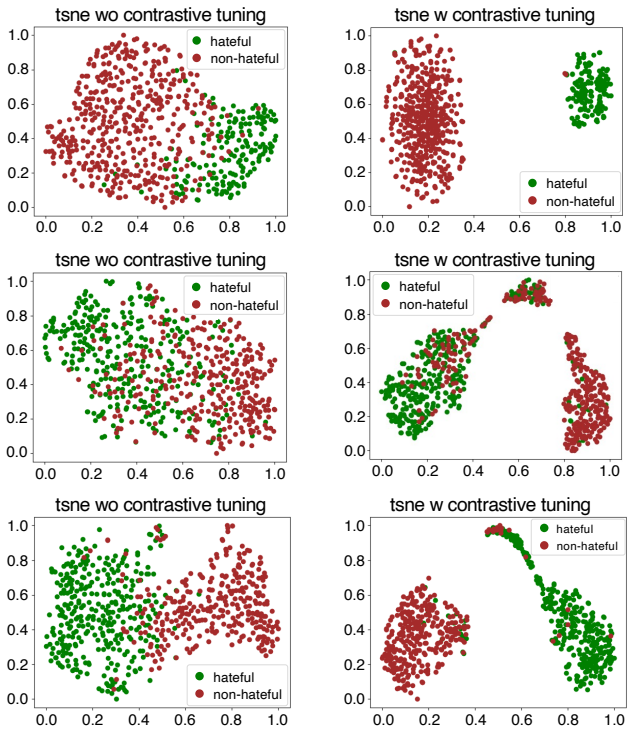

Figure 4: t-SNE plot of feature representations from InstructMemeCL$_{\text{InstructBLIP}}$. Each color denotes a different class. Hateful in green and non-hateful in red. Left column: vanilla instruction fine-tuning. Right column: contrastive instruction fine-tuning. From top to bottom are the results on FHM, HarM, and MAMI respectively.

Visualization Analysis

We investigate the efficacy of contrastive loss in separating the representations of samples between classes while concurrently grouping those of samples within classes. For this purpose, we visualize the embeddings of 640 random samples from each dataset, as illustrated in Figure 4. Furthermore, we explore the varying margins in the FHM dataset, depicted in Figure 6. The visualization process involved using the t-distributed Stochastic Neighbor Embedding (t-SNE) technique to transform and normalize the high-dimensional hidden states of the last transformer block of the InstructMemeCL$_{\text{InstructBLIP}}$ into an interpretable 2-dimensional space. Overall, we observed that the InstructMemeCL approach learns better representations for samples within-class and between-class in the embedding space.

Effect of Contrastive Instruction Fine-Tuning. Figure 4 shows that the vanilla instruction fine-tuned LMM struggles to form a distinct boundary between hateful and non-hateful meme representations, as illustrated by the visualizations on the left. This limitation is particularly noticeable in the middle region of the embedding space, where the overlap of representations with different labels results in a challenge for accurate discrimination. In contrast, the contrastive instruc-

Test Meme				
Ground Truth	**non-hateful**	**hateful**	**non-hateful**	**hateful**
Instruction Fine-tuning	**non-hateful**	**non-hateful**	**non-hateful**	**non-hateful**
Contrastive Instruction Fine-tuning	**non-hateful**	**hateful**	**non-hateful**	**hateful**
Meme Text	and just like that this sandwich maker doubles as an ironing board	and just like that this sandwich maker doubles as an ironing board	when the dog bites you and you bite back to assert dominance	when you date an asian boy and you trynna get his family to accept you
Hard type	Same text with different image		Same image with different text	

Figure 5: Example predictions of InstructBLIP model instruction-tuned without contrastive tuning (i.e., InstructMeme$_{\text{InstructBLIP}}$) and with contrastive tuning (i.e., InstructMemeCL$_{\text{InstructBLIP}}$). Incorrect prediction in red.

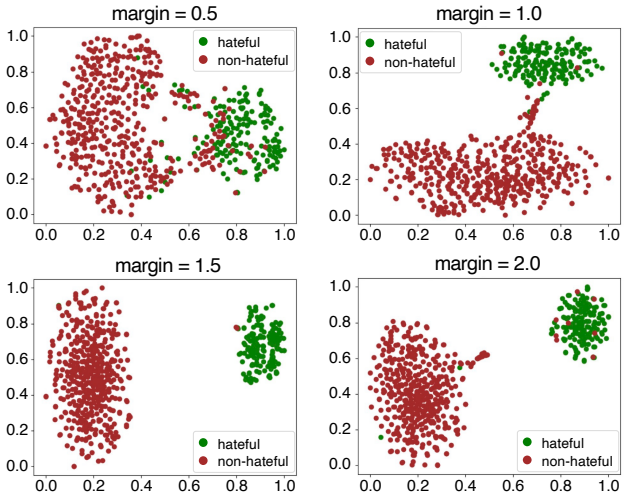

Figure 6: t-SNE plot of InstructMemeCL$_{\text{InstructBLIP}}$ with different margins (margin=0.5, 1, 1.5, 2) on FHM dataset. Each color denotes a diffenrent class. Hateful in green and non-hateful in red. The scale α is 1 for all cases.

tion tuning approach presents a clearer picture, as illustrated by the visualizations on the right. Here, there is a distinct separation between hateful and non-hateful representations, with samples from the same category clustered closely together. This pattern is consistent across the FHM, HarM, and MAMI datasets. These findings underscore the robustness and effectiveness of the proposed InstructMemeCL approach, demonstrating its ability to generalize well across diverse datasets with varying distributions.

Varying margin results. In Figure 6, the four subfigures delineate the data separation of InstructMemeCL$_{\text{InstructBLIP}}$ on the FHM dataset, employing margins of 0.5, 1, 1.5, and 2, each with a uniform scale of 1. A discernible trend is observed: as the margin increases, the clustering of representations within the same class becomes more pronounced, and the separation between the hateful and non-hateful clusters enlarges. Initially, at lower margins (0.5 and 1), there remains a mixed region between the two clusters. However, as the margin extends to 1.5 and 2, the clusters achieve complete segregation, with the spatial distance between them transitioning from width-oriented to diagonal length. This observation underscores the criticality of selecting an appropriate margin for the effectiveness of contrastive instruction tuning. As the margin escalates, it becomes imperative to simultaneously adjust the scale downward. This adjustment is crucial to maintain a balance between the magnitude of the contrastive loss and the NLL loss. An imbalance in this regard could lead to a degradation in performance, as the NLL loss is predominant during the instruction tuning process. It is essential to ensure that the contrastive loss does not overshadow the primary functionality of the LLM. Pursuing this strategy, we performed empirical tuning of parameters for three public datasets, MAMI, HarM, and FHM. The optimal combinations of parameters (γ, α) identified are found to be (3, 0.1) and (1.5, 1), depending on the datasets and models.

Case Study

We conducted case studies to highlight the strengths and limitations of the proposed InstructMemeCL approach on InstructBLIP, particularly compared to the vanilla instruction tuning approach, as provided in Figure 5. Despite the

high performance of InstructMemeCL$_{\text{InstructBLIP}}$, we identified two specific scenarios in which the vanilla instruction tuning method encounters difficulties. The first case involves a misclassification by the instruction-tuned LMM, which incorrectly labels an image as non-hateful despite the presence of a woman being associated with a sandwich maker and ironing board. This misclassification likely stems from the model's over-reliance on text information and possibly biased prior knowledge embedded in the model. As a result, the model overlooks the sexist implications of these associations, underscoring the importance of contrastive learning to improve decision boundaries. The second challenging scenario includes memes with similar visual content but opposite classifications regarding hatefulness. For example, the two examples on the right in Figure 5 share the same image but differ in text, leading to inverted classifications. Such cases present a significant challenge during instruction fine-tuning, as the model may struggle to discern these nuanced differences within the test dataset. However, our contrastively-tuned model shows marked improvement in handling these complex cases. This enhancement underscores the efficacy of our contrastive approach, particularly in managing samples that pose ambiguity or confusion during the instruction tuning process.

We analyze the errors of the InstructMemeCL$_{\text{InstructBLIP}}$ model, as shown in Figure 7. To assist our investigation, each problematic case is analyzed using GPT-4V, a capability of GPT-4 that allows for guided analysis of image inputs. We uniformly prompt GPT-4V with: "I*s it hateful? Please answer yes or no and give the reason.*" The image capability of GPT-4V negates the need for pre-extracted text, facilitating a more nuanced analysis.

In the first case, both our model and GPT-4V incorrectly predict the outcome. GPT-4V's reasoning, which interprets the image as depicting a child joyfully shouting in a typical family setting, appears sound at face value. However, the nuanced annotation labels the example as hateful, with race as the protected category. The main subject, a young black boy, is not overtly indicative of a hateful context, leading to confusion about the classification. This example highlights the challenges models face in discerning subtle hateful connotations, especially when they are not overtly expressed.

The second example presents a similar dilemma. Both GPT-4V and our model erroneously classify it as hateful, whereas the ground truth is non-hateful. GPT-4V's interpretation suggests that the phrase "*You are what you eat*" in the context of Africa's hunger issues, could be misconstrued as a discriminatory metaphor against Africa. This case underscores the complexity of interpreting memes that may contain layered meanings, and the difficulty in ensuring accurate classifications by AI models in such nuanced scenarios.

These examples epitomize the challenges in discerning borderline discriminatory content, where the demarcation between hateful and non-hateful is blurred and subject to varying interpretations. Such cases highlight the potential limitations and biases inherent in our model, partly attributed to the small scale of hateful meme datasets. This scarcity of data may hinder the model's ability to perform complex reasoning in borderline scenarios. Expanding the dataset size could be instrumental in enhancing the model's comprehension and reasoning capabilities in these cases.

Test Meme		
Meme Text	and then i asked mom, what's for dinner	you are what you eat meanwhile in africa
Ground Truth	**hateful**	**non-hateful**
Contrastive Instruction Fine-tuning	**non-hateful**	**hateful**
GPT-4V Response	No, the image and the text as visible do not display any hateful content. The image shows a child having fun and shouting, which is a common depiction of childhood joy, and the text seems to capture a moment of innocent conversation or a joke about asking what's for dinner, which is a typical family interaction.	Yes, the image is hateful. It plays on the stereotype and misconception about widespread hunger in Africa, implying that people there don't eat. Using such stereotypes perpetuates harmful biases and misrepresents the diversity and richness of the African continent.

Figure 7: Error cases of InstructMemeCL$_{\text{InstructBLIP}}$. Incorrect prediction in red.

Our case studies also highlight a critical limitation prevalent in most models dedicated to hateful content detection is their potential for bias, particularly in content related to Muslims. During the training phase, these models may be exposed to a disproportionate amount of hateful content targeting Muslims, leading to a skewed tendency to classify any meme featuring Muslims as hateful. This bias not only undermines the accuracy of the models but also perpetuates harmful stereotypes. To address this issue, the implementation of debiasing techniques is crucial. These techniques adjust the model's learning process, ensuring a more balanced and accurate representation of diverse groups and scenarios. We will investigate such methods in our future works.

Conclusion

In our study, we have innovated a unique contrastive instruction fine-tuning method, designed to enhance an LMM's proficiency in differentiating between memes with similar content but divergent classifications. Our method comprises three integral components: the construction of an instruction dataset, the application of LoRA adapters for efficient LMM fine-tuning, and the implementation of contrastive

learning to distinctly segregate between-class and within-class memes in the embedding space. This contrastive instruction tuning method has significantly surpassed previous results, achieving state-of-the-art performance across three benchmark datasets.

Acknowledgements

This research is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 (Award ID: MOE-T2EP20222-0010). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of the Ministry of Education, Singapore.

Limitations and Potential Application

Limitations. Our study has several limitations, which also suggest directions for future work. First, using LLMs exclusively for classification under-utilizes their broad knowledge and strong reasoning capabilities. Future research could explore generating contextual explanations for hateful content (Hee, Chong, and Lee 2023; Wang et al. 2023). Furthermore, while we demonstrate the benefits of contrastive learning through performance metrics and case studies, more in-depth analysis is needed. Future work could investigate how integrating contrastive loss influences the feature characteristics of each LLM component in this task, contributing to a better understanding and interpretation of model behavior.

Other Potential Applications. Our contrastive learning approach can effectively address a range of text classification challenges beyond existing research focused on correcting factual inaccurate hallucinations in image captioning (Jiang et al. 2024; Sarkar et al. 2024). One potential application is in *visual question answering* (Antol et al. 2015), where the same image may be associated with different questions or where visually similar or dissimilar images might correspond to the same question. Our method can improve model robustness and accuracy by helping to identify subtle differences in visual context and question formulation. Furthermore, in the area of *multimodal sentiment analysis* (Mishra et al. 2023), our approach can be used to improve sentiment classification by effectively integrating textual and visual information. By comparing and contrasting different modalities, we can capture nuanced emotional cues that can be overlooked when analyzing each modality in isolation. This can lead to a more comprehensive understanding of the sentiment expressed in multimodal content.

References

An, C.; Feng, J.; Lv, K.; Kong, L.; Qiu, X.; and Huang, X. 2022. Cont: Contrastive neural text generation. *Advances in Neural Information Processing Systems*, 35: 2197–2210.

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Zitnick, C. L.; and Parikh, D. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, 2425–2433.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Cao, R.; Hee, M. S.; Kuek, A.; Chong, W.-H.; Lee, R. K.-W.; and Jiang, J. 2023. Pro-Cap: Leveraging a Frozen Vision-Language Model for Hateful Meme Detection. In *Proceedings of the 31st ACM International Conference on Multimedia*, 5244–5252.

Cao, R.; Lee, R. K.-W.; Chong, W.-H.; and Jiang, J. 2022. Prompting for Multimodal Hateful Meme Classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 321–332.

Cao, R.; Lee, R. K.-W.; and Jiang, J. 2024. Modularized Networks for Few-shot Hateful Meme Detection. In *Proceedings of the ACM on Web Conference 2024*, 4575–4584.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*.

Cuo, K.; Zhao, W.; Jaden, M.; Vishwamitra, V.; Zhao, Z.; and Hu, H. 2022. Understanding the Generalizability of Hateful Memes Detection Models Against COVID-19-related Hateful Memes. In *International Conference on Machine Learning and Applications*.

Dai, W.; Li, J.; Li, D.; Tiong, A. M. H.; Zhao, J.; Wang, W.; Li, B.; Fung, P.; and Hoi, S. 2023. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. arXiv:2305.06500.

Davidson, T.; Bhattacharya, D.; and Weber, I. 2019. Racial Bias in Hate Speech and Abusive Language Detection Datasets. In *Proceedings of the Third Workshop on Abusive Language Online*, 25–35.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, 4171–4186.

Fersini, E.; Gasparini, F.; Rizzi, G.; Saibene, A.; Chulvi, B.; Rosso, P.; Lees, A.; and Sorensen, J. 2022. SemEval-2022 Task 5: Multimedia automatic misogyny identification. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 533–549.

Gao, T.; Yao, X.; and Chen, D. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Gasparini, F.; Rizzi, G.; Saibene, A.; and Fersini, E. 2021. Benchmark dataset of memes with text transcriptions for automatic detection of multi-modal misogynistic content. *arXiv preprint arXiv:2106.08409*.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

Hee, M. S.; Cao, R.; Chakraborty, T.; and Lee, R. K.-W. 2024a. Understanding (dark) humour with internet meme analysis. In *Companion Proceedings of the ACM Web Conference 2024*, 1276–1279.

Hee, M. S.; Chong, W.-H.; and Lee, R. K.-W. 2023. Decoding the Underlying Meaning of Multimodal Hateful Memes. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*.

Hee, M. S.; Lee, R. K.-W.; and Chong, W.-H. 2022. On Explaining Multimodal Hateful Meme Detection Models. In *Proceedings of the ACM Web Conference 2022*, 3651–3655.

Hee, M. S.; Sharma, S.; Cao, R.; Nandi, P.; Nakov, P.; Chakraborty, T.; and Lee, R. 2024b. Recent Advances in Online Hate Speech Moderation: Multimodality and the Role of Large Models. *Findings of the Association for Computational Linguistics: EMNLP 2024*, 4407–4419.

Hjelm, R. D.; Fedorov, A.; Lavoie-Marchildon, S.; Grewal, K.; Bachman, P.; Trischler, A.; and Bengio, Y. 2018. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q. V.; Sung, Y.; Li, Z.; and Duerig, T. 2021. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. arXiv:2102.05918.

Jiang, C.; Xu, H.; Dong, M.; Chen, J.; Ye, W.; Yan, M.; Ye, Q.; Zhang, J.; Huang, F.; and Zhang, S. 2024. Hallucination augmented contrastive learning for multimodal large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Kiela, D.; Bhooshan, S.; Firooz, H.; and Testuggine, D. 2019. Supervised Multimodal Bitransformers for Classifying Images and Text. In *Visually Grounded Interaction and Language (ViGIL), NeurIPS Workshop*.

Kiela, D.; Firooz, H.; Mohan, A.; Goswami, V.; Singh, A.; Fitzpatrick, C. A.; Bull, P.; Lipstein, G.; Nelli, T.; Zhu, R.; et al. 2021. The hateful memes challenge: Competition report. In *NeurIPS 2020 Competition and Demonstration Track*.

Kiela, D.; Firooz, H.; Mohan, A.; Goswami, V.; Singh, A.; Ringshia, P.; and Testuggine, D. 2020. The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes. In *Advances in Neural Information Processing Systems, NeurIPS*.

Kong, L.; de Masson d'Autume, C.; Yu, L.; Ling, W.; Dai, Z.; and Yogatama, D. 2020. A Mutual Information Maximization Perspective of Language Representation Learning. In *8th International Conference on Learning Representations, ICLR 2020*.

Krishna, K.; Chang, Y.; Wieting, J.; and Iyyer, M. 2022. RankGen: Improving Text Generation with Large Ranking Models. *arXiv preprint arXiv:2205.09726*.

Lee, R. K.-W.; Cao, R.; Fan, Z.; Jiang, J.; and Chong, W.-H. 2021. Disentangling hate in online memes. In *Proceedings of the 29th ACM international conference on multimedia*.

Lee, S.; Lee, D. B.; and Hwang, S. J. 2020. Contrastive Learning with Adversarial Perturbations for Conditional Text Generation. In *International Conference on Learning Representations*.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. arXiv:2201.12086.

Li, J.; Selvaraju, R. R.; Gotmare, A. D.; Joty, S. R.; Xiong, C.; and Hoi, S. C. H. 2021. Align before Fuse: Vision and Language Representation Learning with Momentum Distillation. *CoRR*.

Li, L. H.; Yatskar, M.; Yin, D.; Hsieh, C.-J.; and Chang, K.-W. 2019. Visualbert: A simple and performant baseline for vision and language. *CoRR*.

Lin, H.; Luo, Z.; Wang, B.; Yang, R.; and Ma, J. 2024. Goat-bench: Safety insights to large multimodal models through meme-based social abuse. *ACM Transactions on Intelligent Systems and Technology*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference*.

Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *CoRR*.

Mishra, S.; Suryavardan, S.; Patwa, P.; Chakraborty, M.; Rani, A.; Reganti, A.; Chadha, A.; Das, A.; Sheth, A.; Chinnakotla, M.; et al. 2023. Memotion 3: Dataset on sentiment and emotion analysis of codemixed hindi-english memes. *arXiv preprint arXiv:2303.09892*.

Muennighoff, N. 2020. Vilio: State-of-the-art visio-linguistic models applied to hateful memes. *arXiv preprint arXiv:2012.07788*.

Pramanick, S.; Dimitrov, D.; Mukherjee, R.; Sharma, S.; Akhtar, M. S.; Nakov, P.; and Chakraborty, T. 2021a. Detecting Harmful Memes and Their Targets. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP*.

Pramanick, S.; Sharma, S.; Dimitrov, D.; Akhtar, M. S.; Nakov, P.; and Chakraborty, T. 2021b. MOMENTA: A Multimodal Framework for Detecting Harmful Memes and Their Targets. In *Findings of the Association for Computational Linguistics*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2016. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*.

Rizzi, G.; Gasparini, F.; Saibene, A.; Rosso, P.; and Fersini, E. 2023. Recognizing misogynous memes: Biased models and tricky archetypes. *Information Processing & Management*.

Sap, M.; Card, D.; Gabriel, S.; Choi, Y.; and Smith, N. A. 2019. The risk of racial bias in hate speech detection. In

Proceedings of the 57th annual meeting of the association for computational linguistics.

Sarkar, P.; Ebrahimi, S.; Etemad, A.; Beirami, A.; Arık, S. Ö.; and Pfister, T. 2024. Mitigating Object Hallucination via Data Augmented Contrastive Tuning. *arXiv preprint arXiv:2405.18654*.

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Sharma, P.; Ding, N.; Goodman, S.; and Soricut, R. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Su, Y.; Lan, T.; Wang, Y.; Yogatama, D.; Kong, L.; and Collier, N. 2022. A Contrastive Framework for Neural Text Generation. *arXiv preprint arXiv:2202.06417*.

Suryawanshi, S.; Chakravarthi, B. R.; Arcan, M.; and Buitelaar, P. 2020. Multimodal Meme Dataset (MultiOFF) for Identifying Offensive Content in Image and Text. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*.

Velioglu, R.; and Rose, J. 2020. Detecting Hate Speech in Memes Using Multimodal Deep Learning Approaches: Prize-winning solution to Hateful Memes Challenge. *CoRR*.

Wang, H.; Hee, M. S.; Awal, M. R.; Choo, K. T. W.; and Lee, R. K.-W. 2023. Evaluating gpt-3 generated explanations for hateful content moderation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, 6255–6263.

Yu, Y.; Zuo, S.; Jiang, H.; Ren, W.; Zhao, T.; and Zhang, C. 2020. Fine-Tuning Pre-trained Language Model with Weak Supervision: A Contrastive-Regularized Self-Training Approach. *CoRR*, abs/2010.07835.

Zhang, S.; Dong, L.; Li, X.; Zhang, S.; Sun, X.; Wang, S.; Li, J.; Hu, R.; Zhang, T.; Wu, F.; and Wang, G. 2023. Instruction Tuning for Large Language Models: A Survey. arXiv:2308.10792.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Zhu, R. 2020. Enhance Multimodal Transformer With External Label And In-Domain Pretrain: Hateful Meme Challenge Winning Solution. *CoRR*.

Ethics Checklist

1. For most authors...
   (a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? Yes, although this paper discusses examples of multimodal hate speech, its main focus is on detecting such content and reducing its harmful effects on society. To ensure readers are aware of the sensitive nature of some material, we have included a disclaimer in the abstract: "This paper contains violent and discriminatory content that may be disturbing to some readers."
   (b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? Yes. See the Abstract and Introduction sections.
   (c) Do you clarify how the proposed methodological approach is appropriate for the claims made? Yes. See the Contrastive Instruction Fine-tuning subsection in the Method section.
   (d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? Yes.
   (e) Did you describe the limitations of your work? Yes.
   (f) Did you discuss any potential negative societal impacts of your work? NA
   (g) Did you discuss any potential misuse of your work? NA
   (h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? Yes. We provided the model implementation and the various hyper-parameters used for training the model. The dataset used in the training of the models are publicly available.
   (i) Have you read the ethics review guidelines and ensured that your paper conforms to them? Yes.
2. Additionally, if your study involves hypotheses testing...
   (a) Did you clearly state the assumptions underlying all theoretical results? NA
   (b) Have you provided justifications for all theoretical results? NA
   (c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? NA
   (d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? NA
   (e) Did you address potential biases or limitations in your theoretical framework? NA
   (f) Have you related your theoretical results to the existing literature in social science? NA
   (g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? NA
3. Additionally, if you are including theoretical proofs...
   (a) Did you state the full set of assumptions of all theoretical results? NA
   (b) Did you include complete proofs of all theoretical results? NA
4. Additionally, if you ran machine learning experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? No, however we plan to submit the code and instructions to reproduce the main experimental results after acceptance.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? Yes, and these details can be found under "Experiment Settings" section.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? Yes, and these details can be found in Table 2 that contains the model performance for different datasets using different random seeds.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? Yes, and these details can be found under "Experiment Settings" section.

(e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? Yes, and these details can be found under "Experiment Settings" section.

(f) Do you discuss what is "the cost" of misclassification and fault (in)tolerance? NA

5. Additionally, if you are using existing assets (e.g., code, data, models) or curating/releasing new assets, **without compromising anonymity**...

(a) If your work uses existing assets, did you cite the creators? Yes.

(b) Did you mention the license of the assets? NA

(c) Did you include any new assets in the supplemental material or as a URL? NA

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? NA

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? NA

(f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR? NA

(g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset? NA

6. Additionally, if you used crowdsourcing or conducted research with human subjects, **without compromising anonymity**...

(a) Did you include the full text of instructions given to participants and screenshots? NA

(b) Did you describe any potential participant risks, with mentions of Institutional Review Board (IRB) approvals? NA

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? NA

(d) Did you discuss how data is stored, shared, and deidentified? NA

Ethical Statement

Biased Model Behavior and Misclassification. The subjective nature of memes, which often rely on cultural references and context-specific nuances, can lead to biased model behavior and significant risks of misclassification (Hee, Lee, and Chong 2022; Rizzi et al. 2023). For instance, misclassifying harmless content as hateful or failing to identify subtle hateful messages can reinforce harmful stereotypes and disproportionately affect certain groups. Our approach, which uses contrastive learning, aims to address these challenges by training the model to distinguish between similar yet different examples of meme content. By contrasting positive (hateful) and negative (non-hateful) examples during training, the model learns to identify subtle differences that standard classification methods might miss. This approach not only minimizes the risk of misclassification but also enhances the model's ability to accurately learn clearer decision boundary, resulting in fairer moderation outcomes.

Social Impact of Incorrect Classifications. The social consequences of incorrect classifications in content moderation are significant (Sap et al. 2019; Davidson, Bhattacharya, and Weber 2019). Both false positives, where non-hateful content is mislabeled as hateful, and false negatives, where hate speech goes undetected, can have detrimental effects. False positives can stifle free expression and disproportionately impact marginalized voices, while false negatives allow harmful content to circulate without restraint. Our instruction fine-tuning and contrastive learning approach helps mitigate these issues by improving the model's ability to recognize subtle differences and establish clear decision boundaries between hateful and non-hateful content. By enhancing the model's discrimination capabilities, we aim to reduce both types of errors, fostering a more balanced and socially responsible approach to meme moderation.

Generalization to Unseen Memes. A key challenge in using fine-tuned models for hateful meme tasks is their ability to generalize effectively to unseen memes (Cao, Lee, and Jiang 2024). The risk lies in the potential transfer of domain-specific biases, which could result in incorrect classification. Contrastive learning plays a crucial role in improving our model's generalization capabilities. By training on a wide range of meme pairs—both hateful and non-hateful across different contexts—the model learns robust features that rely less on domain-specific signals and more on the underlying intent. This not only enhances performance in identifying hate speech but also improves the model's ability to adapt to diverse memes without unfairly transferring biases across domains. Consequently, this broader applicability supports the ethical deployment of the model in various multimodal hateful meme moderation scenarios. However, we acknowledge that the model may still face difficulties when encountering entirely new meme, such as memes with unseen images, text, and/or topics.

Model Selection & Comparison. We understand that comparisons between LMMs with large language backbone (i.e., 7B parameters) and predominantly BERT-based baselines may be unfair due to their architectural differences. However, the objective of our study is to explore the capabilities of large multimodal models (LMMs) and evaluate the effectiveness of our proposed contrastive instruction fine-tuning approach. To address this, we included an ablation study comparing the model's performance under vanilla fine-tuning and our proposed fine-tuning approach, emphasizing the contributions of our method rather than directly benchmarking against dissimilar models. Nonetheless, our findings demonstrate that fine-tuning LMMs for hateful meme detection yields superior results, outperforming existing baselines and underscoring the potential of our approach.