Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.064734

ARTICLE

Low-Rank Adapter Layers and Bidirectional Gated Feature Fusion for Multimodal Hateful Memes Classification

Youwei Huang, Han Zhong*, Cheng Cheng and Yijie Peng

College of Information and Cyber Security, People's Public Security University of China, Beijing, 100038, China

*Corresponding Author: Han Zhong. Email: zhonghan@ppsuc.edu.cn

Received: 22 February 2025; Accepted: 25 April 2025; Published: 09 June 2025

ABSTRACT: Hateful meme is a multimodal medium that combines images and texts. The potential hate content of hateful memes has caused serious problems for social media security. The current hateful memes classification task faces significant data scarcity challenges, and direct fine-tuning of large-scale pre-trained models often leads to severe overfitting issues. In addition, it is a challenge to understand the underlying relationship between text and images in the hateful memes. To address these issues, we propose a multimodal hateful memes classification model named LABF, which is based on low-rank adapter layers and bidirectional gated feature fusion. Firstly, low-rank adapter layers are adopted to learn the feature representation of the new dataset. This is achieved by introducing a small number of additional parameters while retaining prior knowledge of the CLIP model, which effectively alleviates the overfitting phenomenon. Secondly, a bidirectional gated feature fusion mechanism is designed to dynamically adjust the interaction weights of text and image features to achieve finer cross-modal fusion. Experimental results show that the method significantly outperforms existing methods on two public datasets, verifying its effectiveness and robustness.

KEYWORDS: Hateful meme; multimodal fusion; multimodal data; deep learning

1 Introduction

With the widespread use of social media, memes have become one of the most important ways of spreading hate speech. A meme can be narrowly defined as a combination of images and words [1], usually conveying information, ideas or culture through humor, satire or images. Hateful memes take advantage of this form of communication to attack specific racial, religious or gender groups in a veiled or satirical manner. Hateful memes have posed a serious threat to social harmony [2]. Therefore, how to accurately and quickly identify hateful memes has become a popular research topic nowadays.

The content of hateful memes frequently intersects with sensitive domains such as cultural differences, political ideologies, and racial identities. The collection of hateful meme datasets is constrained by legal restrictions, resulting in the general small size of the currently available datasets. Under this background, mainstream models like CLIP [3], despite their superior performance across various tasks, usually have a large number of parameters. When trained on small scale datasets, these models tend to overfit, which greatly affects their generalization ability, thus limiting their effectiveness in classifying hateful memes. Effectively avoiding overfitting on small-scale datasets while ensuring that the model maintains high efficiency and stability has become a critical issue in hateful memes classification. To address this challenge, recent studies have proposed the use of the adapter technique [4–6] to fine-tune pre-trained models. By inserting a

small number of trainable adapter modules into the pre-trained model without fully updating the entire model, Adapter can effectively reduce the computational resource requirements while maintaining the strong performance of the original model. Although adapters reduce the number of trainable parameters, the additional parameters introduced by them may still exacerbate overfitting on small datasets. This creates higher demands for the parameter optimization and lightweight design of the adapters.

Cross-modal feature fusion of text and images is important in hateful memes classification task. Images or text alone may appear innocuous in the meme, but when images are combined with text, their underlying true intent becomes apparent [7]. Accurately modeling the cross-modal semantic interactions is key to enhancing the model's ability to recognize hateful memes. Current multimodal fusion methods, such as the use of concatenation in ConcatBERT [8] and elemental multiplication in Coinclip [9], tend to ignore the complex interactions between modalities and the importance of the information. These methods simply stack features at the feature level, lacking modeling of the semantic dependencies between modalities, making it difficult to effectively capture the deep and dynamic interaction information between text and image. On the other hand, these approaches often fail to flexibly adjust the weight of image and text features fusion based on the specific meme content. The limitations of this fixed fusion strategy reduce the model's ability to recognize complex memes and constrain its generalization performance.

To solve the above problems, we propose the following methods. (1) We adopted the low-rank adapter fine-tuning technique to alleviate the overfitting problem that arises in models trained on small datasets. Influenced by LoRA [10] and COMPACTER [11], we use a low-rank matrix to optimize the design of the feature adapter. By decomposing the feature mapping process into the product of two low-rank matrices, we reduce the number of parameters in the model. This design makes the model more effective in avoiding overfitting and improving training efficiency when fine-tuning on small-scale datasets. (2) To address the issues of insufficient modeling of semantic dependencies between modalities and inflexible feature weight adjustment in existing multimodal fusion methods, we designed a bidirectional gated feature fusion mechanism to model the dynamic semantic dependencies between image and text. This mechanism introduces two gating networks to control the information flow between image and text features. Two learnable scaling factors are used to dynamically adjust the fusion weights. The bidirectional gated feature fusion mechanism enables adaptive modeling of modality importance in different contexts, thereby enhancing the expressive power of cross-modal semantic interactions.

We summarize the following contributions:

To apply the CLIP model more efficiently for hateful memes classification, we employ a low-rank feature adapter to lightly fine-tune CLIP. By updating only a small number of parameters, we significantly improve its performance in the hateful memes classification task.

We design a Bidirectional Gated Feature Fusion (BiGFF) that dynamically adjusts the interaction weights between text and image, enhancing the fusion capability of cross-modal information and more accurately capturing potential offensive expressions in memes.

The proposed method achieves better performance than traditional methods on two small-scale datasets, demonstrating its robustness and efficiency in hateful memes classification.

2 Related Work

In this section, we introduce related work from three aspects: feature extraction of hateful memes, fine-tuning methods for multimodal models, and multimodal feature fusion methods.

2.1 Feature Extraction of Hateful Memes

In early hateful memes classification, image and text encoders are often pre-trained independently for image and text feature extraction. Wang et al. [12] used Twitter-RoBERTa [13] model to extract text features and Swin Transformer V2 [14] model to extract image features and classify them through multilayer perceptron (MLP) [15] fusion mechanism. Gomez et al. [16] used Inception v3 [17] model to extract image features and used LSTM network combined with GloVe word embedding to extract textual features, and multimodal fusion methods such as feature splicing and text convolution kernel to detect hate speech in multimodal publications. Riza Velioglu et al. [18] achieved an effective detection of hate speech through multimodal deep learning (VisualBERT) [19] and integrated learning.

With the proposal of the CLIP model, more and more research has started to apply it to multimodal hateful memes detection task. CLIP makes cross-modal feature fusion more efficient and accurate by mapping images and text to a shared embedding space through contrast learning. Burbi et al. [7] proposed a multimodal hate speech detection method (ISSUES) that combines text inversion techniques [20] with a frozen CLIP model. This method enhances the expressive power of textual features by mapping images from emoticons to pseudo-word tokens in the CLIP text embedding space. Mei et al. [21] proposed a multimodal hate speech detection method that combines retrieval-guided contrast learning (RGCL). This method refines the embedding space by dynamically retrieving pseudo-positive and hard-negative examples during training. This approach ensures that modals with the same label are tightly coupled while modals with different labels are effectively separated. However, ISSUES freezes both the image and text encoders while RGCL freezes the image encoder during training. This means that these models can only use the feature representations learned during the pre-training phase of CLIP. These models cannot further adapt and optimize these features based on the specific data of the hateful memes classification task. Siddhant Bikram Shah et al. [22] proposed a multimodal hate speech detection framework named MemeCLIP. Although this method freezes the parameters of both the CLIP visual encoder and the text encoder, it introduces feature adapters [4], allowing the model to perform task-specific optimization while preserving the pre-trained knowledge of CLIP. MemeCLIP improves multimodal meme classification by using feature adapters. These adapters optimize features through two trainable linear layers. While these linear layers enhance the model's performance, they also add extra parameters resulting in increased computational overhead. In contrast, our method applies a lightweight optimization to the adapters based on LoRA technology. By decomposing the feature mapping process into the product of two low-rank matrices, we significantly reduce the adapter's parameter count while improving training efficiency. Unlike the traditional adapter approach in MemeCLIP, the low-rank adapter is more efficient on small datasets, avoiding overfitting and enhancing the robustness and accuracy of the hateful memes classification task. Additionally, whereas MemeCLIP uses element-wise multiplication for image and text feature fusion, we employ the BiGFF mechanism. This mechanism dynamically adjusts feature fusion through a gating network, allowing for more effective modeling of the dependencies between image and text. By introducing Low-Rank Adapter and BIGFF feature fusion mechanism, we further improve the hate modality detection ability of the model.

2.2 Fine-Tuning Methods for Multimodal Models

With the widespread application of multimodal pre-trained models, achieving parameter-efficient fine-tuning (PEFT) while maintaining performance has emerged as a research focus. Traditional full-scale fine-tuning methods involve a large number of parameters and high training costs. To address these challenges, researchers have proposed various PEFT methods to meet the needs of different tasks and modalities. In existing PEFT methods, prompt tuning is a widely adopted strategy. It primarily guides the model to complete downstream tasks by incorporating learnable prompt vectors. Zhou et al. [23] proposed

the Context Optimization method, which substitutes the fixed text prompt with a learnable context vector. This approach enables optimization solely on the prompt while keeping the pre-trained model parameters frozen. Chen et al. [24] proposed prompt learning with optimal Transport (PLOT) method, which learns multiple prompt vectors for each class to enhance the diversity of representations. PLOT introduces local visual features and leverages the optimal transport mechanism to achieve fine-grained alignment between prompts and visual features. In addition to prompt learning, another common parameter-efficient fine-tuning method is adapter, which is based on the core idea of inserting lightweight modules into the backbone model to learn task-specific knowledge. Gao et al. [4] proposed the CLIP-Adapter method, which inserts lightweight bottleneck adapter modules at the end of the visual or text branches. The newly learned features are fused with the original representations through residual connections. Yu et al. [5] proposed task residual tuning method, which effectively decouples pre-trained knowledge from task-specific knowledge by introducing learnable task residuals. Zhang et al. [6] proposed the Tip-Adapter method, which achieves efficient adaptation without fine-tuning the CLIP model parameters. By constructing a cache model based on few-shot image features and labels, it enables training-free adaptation.

2.3 Multimodal Model Feature Fusion

In the task of multimodal hateful memes classification, effectively integrating image and text information to achieve deeper semantic understanding is an important challenge. Lippe et al. [25] proposed a multimodal hateful memes classification model based on an early fusion strategy. This method extracts image features using Faster R-CNN [26] and text features using BERT [27]. The extracted features are concatenated and fed into the Transformer encoder, where deep semantic fusion between the image and text is achieved through a shared self-attention mechanism. Zichao Li et al. [28] employed ResNet [29] and XLM-RoBERTa [30] to encode images and text in the multimodal hateful memes classification. They used a bidirectional cross-attention feature fusion mechanism, treating the image and text as queries and applying attention weighting to the other modality. Pramanick et al. [31] adopted a hierarchical fusion strategy, achieving deep image-text fusion through intra-modal attention and cross-modal attention mechanisms. Kumar et al. [32] proposed an intermediate fusion strategy based on CLIP features from a feature-structure perspective. The core idea is to construct a feature interaction matrix, explicitly modeling the relationships between all dimensional levels of the image and text features through their outer product. Hossain et al. [33] proposed a multimodal fusion method based on alignment and attention mechanisms. This method performs semantic alignment of image and text features using an additive attention mechanism, generates context vectors, and finally concatenates them with the original features to form a multimodal representation.

3 Methodology

In this section, we will provide a detailed introduction to our proposed model, LABF, which is a CLIP-based hateful memes classification model. First, we use the CLIP framework for initial image and text feature extraction. Then, we introduce low-rank adapters to fine-tune the CLIP model to enhance the feature representation. Subsequently, we employ a bidirectional gated fusion mechanism to fuse image and text features, fully exploiting the complementarity between multimodal information. Finally, a cosine classifier is used for the classification task. Fig. 1 illustrates the overall architecture of the LABF model. In the following, we describe each component of the model in detail.

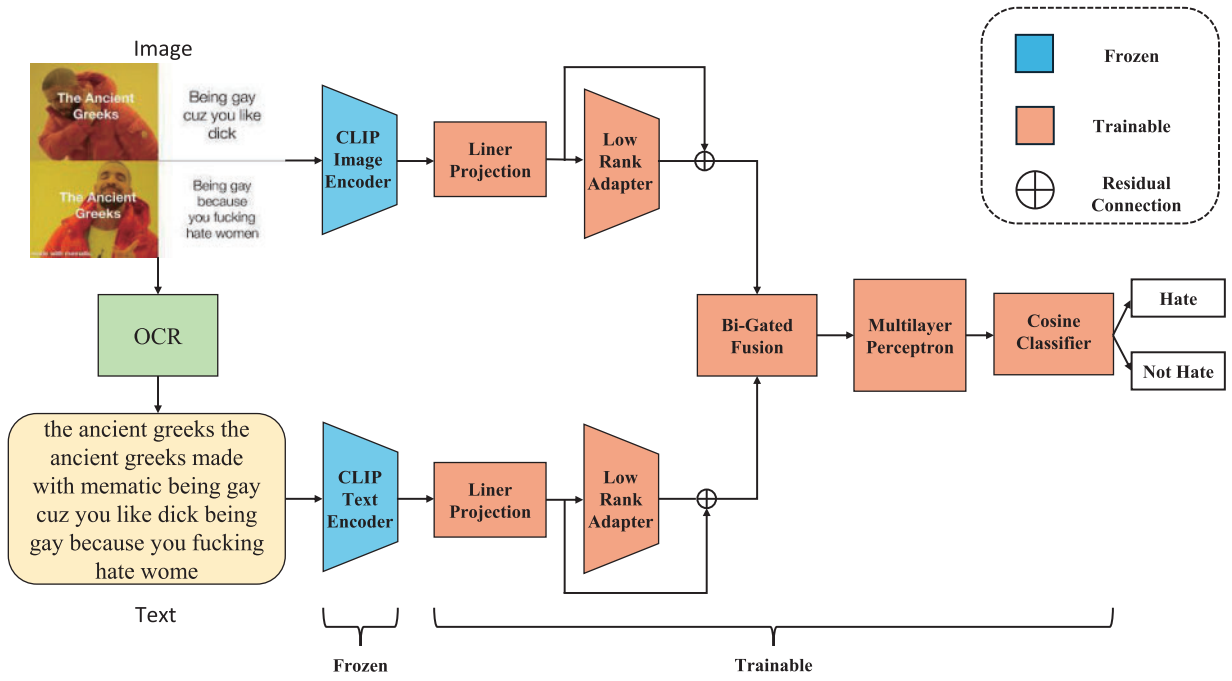

Figure 1: General framework of the hateful memes classification model (LABF) based on low-rank adapter layers and bidirectional gated feature fusion

3.1 Data Preprocessing

Before training the model, we perform the necessary data preprocessing on the text and images. For text data, we replace invalid texts with 'null'. We use the CLIP tokenize method to convert the text into tokens of a fixed length, with a maximum length of 77. Texts exceeding this length were truncated to ensure input consistency. For image data, we load the images and convert them to RGB format. We resized the images to a uniform size of (224 × 224) to ensure consistency and normalization of image inputs.

3.2 Feature Extraction

CLIP is a multimodal deep learning model proposed by OpenAI that aims to achieve cross-modal understanding by jointly learning image and text representations. The CLIP model is trained on a large-scale image-text pairing dataset. This large-scale training allows CLIP to learn a wide range of visual and linguistic features, which enhances its cross-modal generalization capabilities. CLIP contains an image encoder and a text encoder. In order to avoid large-scale parameter tuning in the fine-tuning process, We freeze the parameters of clip throughout the training. As shown in Eqs. (1) and (2), We use the image encoder (f_{image}) and the text encoder (f_{text}) to extract image and text features.

$$v_{\text{image}} = f_{\text{image}}(I) \tag{1}$$

$$v_{\text{text}} = f_{\text{text}}(T) \tag{2}$$

where I is an image and T denotes the textual information extracted from the image.

We introduce a linear projection layer to effectively decouple image and text features and further optimize their alignment in a shared latent space. In hateful memes classification tasks, images and text often convey different semantic information, so we use a trainable linear projection layer that will map

image features and text features to a space more suitable for the task requirements, thus improving their expressiveness in a specific task. The linear projection operation can be described by Eqs. (3) and (4).

$$F_i = L_i(v_{\text{image}}) \tag{3}$$

$$F_t = L_t(v_{\text{text}}) \tag{4}$$

where L_i and L_t denote the projection layers applied to image and text features.

3.3 Low-Rank Adapters

Although CLIP is pre-trained on large-scale datasets and efficiently captures the relationship between images and text, the model may show symptoms of overfitting when applied to smaller datasets. On small datasets, models tend to memorize features of the training data and fail to generalize effectively. We add a low-rank adapters after the CLIP encoder to further adjust and fine-tune the feature space representation. Low-rank adapters allow the model to fine-tune on new data while retaining the pre-trained knowledge of CLIP. We use residual connection to fuse the features of the linear projection layer and the features of the low-rank adapters. We balance the fusion degree between the two features by the residual ratio α.

To enhance the feature representation capability of the model without introducing a large number of parameters, the low-rank adapter we use primarily involves three steps: dimensionality reduction, non-linear activation, and dimensionality expansion. First, the input features are reduced in dimensionality, followed by a non-linear activation to enhance feature interaction capability, and then the dimensionality is expanded to restore the representation. This design not only significantly reduces the number of parameters but also preserves the model's non-linear modeling ability. The low-rank adapter can be represented by Eq. (5).

$$y = W_{\text{up}} \cdot \text{gelu}(W_{\text{down}} \cdot x) \tag{5}$$

where $x \in \mathbb{R}^{1024}$ is the input feature, $W_{\text{down}} \in \mathbb{R}^{32 \times 1024}$ is the dimensionality reduction matrix, $W_{\text{up}} \in \mathbb{R}^{1024 \times 32}$ is the dimensionality expansion matrix, and gelu is the activation function.

To further reduce the number of parameters, we apply the concept of low-rank matrix factorization. The dimensionality reduction matrix $W_{\text{down}} \in \mathbb{R}^{32 \times 1024}$ and the dimensionality expansion matrix $W_{\text{up}} \in \mathbb{R}^{1024 \times 32}$ are approximated as the product of two rank-1 small matrices A_i and B_i. The dimensionality reduction matrix W_{down} and the dimensionality expansion matrix W_{up} can be specifically represented as Eqs. (6) and (7).

$$W_{\text{up}} = A_1 B_1 \tag{6}$$

$$W_{\text{down}} = A_2 B_2 \tag{7}$$

where $A_1 \in \mathbb{R}^{1024 \times 1}$, $B_1 \in \mathbb{R}^{1 \times 32}$, $A_2 \in \mathbb{R}^{32 \times 1}$, and $B_2 \in \mathbb{R}^{1 \times 1024}$. By using this approach, we significantly reduce the parameter scale while still preserving the model's fundamental ability to represent features.

To ensure good numerical stability and gradient propagation during the early stages of training, we apply the Xavier Uniform initialization strategy to the weight matrices A_i and B_i involved in the low-rank decomposition. This method automatically adjusts the initialization interval based on the input and output dimensions, with each weight element sampled from the following uniform distribution. The entire initialization method can be represented by Eq. (8).

$$W \sim U\left[-\sqrt{\frac{6}{n_{\text{in}} + n_{\text{out}}}}, \sqrt{\frac{6}{n_{\text{in}} + n_{\text{out}}}}\right] \tag{8}$$

where U represents the uniform distribution, and n_{in} and n_{out} represent the input and output dimensions of the layer. This initialization method helps maintain consistent output variance across layers during forward propagation, while avoiding issues such as vanishing or exploding gradients, thereby accelerating the model's convergence process.

We use a low-rank adapter on both the image and text sides. Each adapter consists of two linear transformations, which are used for down-sampling and up-sampling operations, to efficiently adjust the feature space. The formula for the low-rank adapter can be represented by Eqs. (9) and (10).

$$A_I(F_i) = W_{\text{up}} \cdot \text{gelu}(W_{\text{down}} \cdot F_i) = (A_1B_1) \cdot \text{gelu}((A_2B_2) \cdot F_i) \tag{9}$$

$$A_T(F_t) = W_{\text{up}} \cdot \text{gelu}(W_{\text{down}} \cdot F_t) = (A_1B_1) \cdot \text{gelu}((A_2B_2) \cdot F_t) \tag{10}$$

where F_i is the image feature, F_t is the text feature, A_I is the image low-rank adapter, and A_T is the text low-rank adapter.

As shown in Eqs. (11) and (12), we obtain the text representation F_T and the image representation F_I after integrating the output of the low-rank adapter with the linear projection.

$$F_I = \alpha A_I(F_i) + (1-\alpha)F_i \tag{11}$$

$$F_T = \alpha A_T(F_t) + (1-\alpha)F_t \tag{12}$$

Under the condition of freezing the CLIP parameters, we learn new features using the low-rank adapter and fuse the original features with the new features from the low-rank adapter through residual connections. This approach not only retains useful information of the original features but also allows the new features to better complement the original ones, thereby enhancing the model's representational ability and performance. In this way, we enhance the model's ability to capture complex patterns in the data without significantly increasing computational complexity, further improving performance in hateful memes classification tasks.

3.4 Bidirectional Gated Feature Fusion

In this study, we propose a bidirectional gated feature fusion called BiGFF, which aims to effectively fuse image and text features. In hateful memes classification task, images and text often contain complementary information. It is crucial to capture the complex relationship between images and text. BiGFF further improves the effectiveness of feature fusion and the task performance by introducing a bidirectional gating mechanism that dynamically adjusts the interaction between image and text features. BiGFF allows each modality to selectively absorb information from the other modality based on its own context, thus mitigating conflicts caused by semantic inconsistencies. Additionally, multimodal features often contain redundant information, and the gating signals generated by BiGFF can dynamically adjust the fusion intensity, suppressing irrelevant interfering features and retaining discriminative information. This bidirectional, adjustable interaction mechanism enables more precise modeling of the deep semantic dependencies between modalities.

The BiGFF module controls the interactions between image and text features through two gating networks. The specific structure is shown in Fig. 2. We generate a composite feature vector by concatenating text and image features. Then, we map the concatenated features to the range $[0,1]$ using a linear transformation followed by a Sigmoid activation function to generate the gated signals G_I and G_T. The two gating networks can be represented by Eqs. (13) and (14).

$$G_I = \sigma(l(\text{Concat}(F_I, F_T))) \tag{13}$$

$$G_T = \sigma(l(\text{Concat}(F_T, F_I))) \tag{14}$$

where Concat denotes the stitching of text features F_T and image features F_I, l is a linear mapping function, σ is a Sigmoid activation function, and the outputs G_I and G_T are gated signals from image to text and from text to image, which are used to control the strength of the interaction between the features.

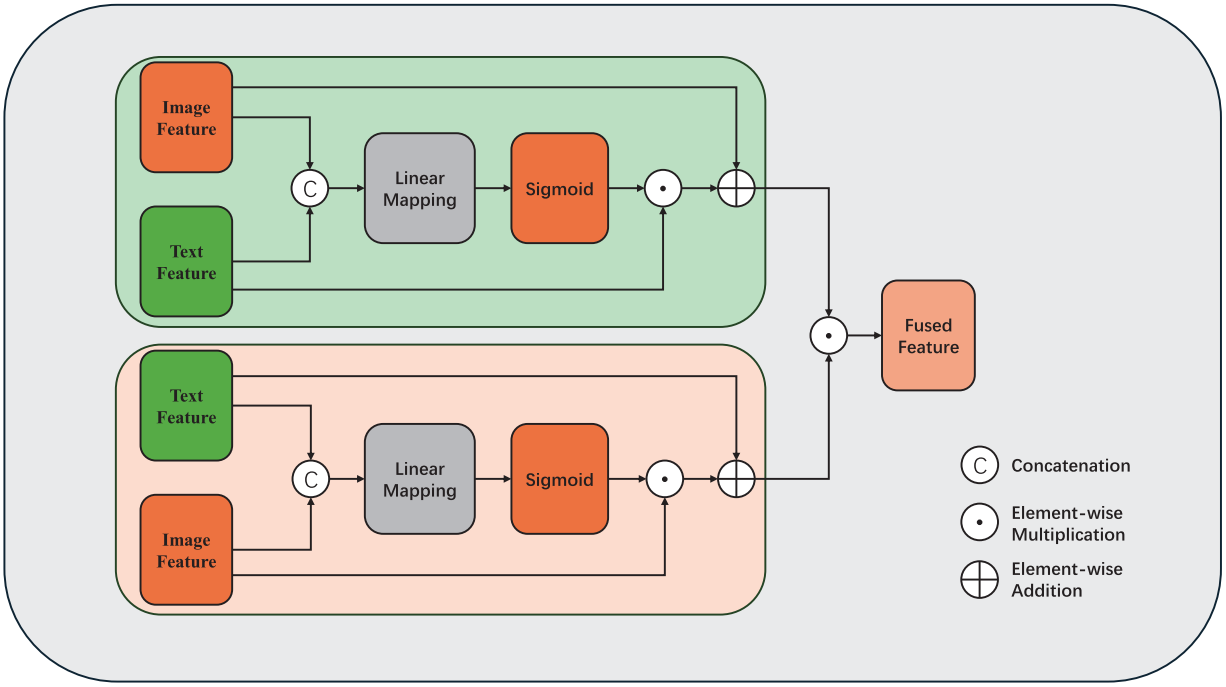

Figure 2: Schematic diagram of Bidirectional Gated Feature Fusion module

As shown in Eqs. (15) and (16), after generating the gating signals, BiGFF uses these signals to regulate the interaction between image and text features, thus enhancing the effective information flow between them. At the same time, we introduce learnable scaling factors λ and μ to control the interaction intensity, resulting in the updated image feature F_I' and text feature F_T'.

$$F_I' = \lambda(G_T \odot F_T) \oplus F_I \tag{15}$$

$$F_T' = \mu(G_I \odot F_I) \oplus F_T \tag{16}$$

where $\odot$ denotes an element-by-element multiplication operation, $\oplus$ denotes an element-by-element addition operation, and λ and μ are learnable scaling factors that control the strength of the modulation. The scaling factors λ and μ are initialized to zero, ensuring that the model initially uses only the raw features during the early stages of training. Subsequently, the model gradually learns the fusion strategy throughout the training process. Through this modulation, BiGFF can flexibly adjust the interaction between image and text features to ensure their effective fusion.

As shown in Eq. (17), F_I' and F_T' are fused by element-by-element multiplication to obtain the final fused feature representation.

$$F_f = F_I' \odot F_T' \tag{17}$$

BiGFF can dynamically adjust the fusion degree of image and text features to ensure the effective combination of the two modalities. This gated mechanism enables the model to automatically adjust the information flow between images and texts according to different inputs, which enhances the ability of feature expression in the classification of hateful memes.

3.5 Classification

We further process the fused multimodal features through multiple fully connected layers with ReLU activation functions. As shown in Eq. (18), we use a cosine classifier for classification. The cosine classifier improves the generalization ability of the model by normalizing and dynamically adjusting the size of the category weights so that the model can better adapt to the feature distributions of different categories during the training process. Specifically, the cosine classifier classifies the input feature vectors based on the cosine similarity between them and the center vector of each category.

$$C = \frac{F_l \cdot F_f}{\|F_l\|_2 \|F_f\|_2} \tag{18}$$

where F_l denotes the center vector of each category, F_f denotes the final fused features, and C denotes the similarity score. Throughout the training process, we use the cross-entropy loss function for training, so that the model can more accurately classify hate modalities.

4 Experimentation

In this section, we present the details of dataset, evaluation metrics. Then, we compare our model with several strong baselines. Finally, a detailed analysis is presented.

4.1 Datasets

To evaluate the effectiveness of the LABF model, we conducted experiments on two representative datasets: PrideMM and HarMeme. The PrideMM dataset focuses on hate speech directed at the LGBTQ+ community, encompassing discrimination and hateful behaviors related to gender identity and sexual orientation. The HarMeme dataset focuses on hate speech associated with COVID-19, primarily including racial discrimination and biased remarks that emerged during the pandemic. Through experiments on these two datasets, we can explore the model's performance across different social contexts. Experiments on these datasets, which have distinct social backgrounds and themes, enabled us to validate the effectiveness and adaptability of the LABF model. The statistical information of the datasets is shown in Table 1.

Table 1: Statistical information on the datasets

Dataset	HarMeme			PrideMM		
	Memes	Harmful	Harmless	Memes	Hate	Not hate
Train	3013	1064	1949	4328	2120	2208
Validation	177	61	116	228	115	113

(Continued)

Table 1 (continued)

Dataset	HarMeme			PrideMM		
	Memes	Harmful	Harmless	Memes	Hate	Not hate
Test	354	124	230	507	247	260
Total	3544	1249	2295	5063	2482	2581

PrideMM [22] dataset was constructed by Siddhant Bikram Shah et al. The dataset was manually searched and extracted relevant images from three popular social media platforms, Facebook, Twitter and Reddit, which were filtered with hashtags related to LGBTQ+ discussions. The dataset contains a total of 5063 images, of which the training set contains 4328 images, the validation set contains 228 images and the test set contains 507 images. In the training set, 2120 images are hate memes and 2208 images are non-hate memes. In the validation set, 115 images are hate memes and 113 are non-hate memes; in the test set, 247 are hate memes and 260 are non-hate memes. The entire dataset consists of 2482 hate memes and 2581 non-hate memes. The PrideMM dataset is primarily centered around the LGBTQ+ movement. Accurate identification of hate speech requires combining the specific LGBTQ+ socio-political context with the unique emotional expressions and cultural context of the field. This presents a significant challenge to existing methods.

The HarMeme dataset [34] was constructed by Pramanick et al. The COVID-19-related memes were collected using various search services and social media platforms, covering keywords such as 'Wuhan virus meme', 'U.S. election meme', and 'U.S. election meme'. The keywords covered include 'Wuhan virus meme', 'US election meme' and 'COVID vaccine meme'. Initially, 5027 memes were collected, but 3544 valid memes were retained after de-emphasis and filtering. The dataset contains 3544 images, of which the training set contains 3013 images, the validation set contains 177 images, and the test set contains 354 images. The training set contains 1064 images labeled as harmful and 1949 images labeled as harmless; the validation set includes 61 images labeled as harmful and 116 images labeled as harmless; the test set consists of 124 images labeled as harmful and 230 images labeled as harmless. The memes in the HarMeme dataset convey their potential harmfulness through satire, political satire, or ambiguity. The harmfulness of many contents needs to be assessed in the context of the underlying social, political, and cultural backgrounds, which presents a significant challenge to the model's detection capabilities.

4.2 Experimental Setup

4.2.1 Evaluation Metrics

For the evaluation metrics of the model, we use accuracy and AUC as the evaluation criteria for the experiments. The formulas for accuracy and AUC are shown in Eqs. (19) and (20).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{19}$$

$$\text{AUC} = \int_0^1 \frac{\text{TP}}{\text{TP} + \text{FN}} \cdot \frac{\text{FP}}{\text{FP} + \text{TN}} \, d\left(\frac{\text{FP}}{\text{FP} + \text{TN}}\right) \tag{20}$$

- *TP* represents the number of samples correctly predicted as positive by the model.
- *TN* represents the number of samples correctly predicted as negative by the model.
- *FP* represents the number of negative samples incorrectly predicted as positive by the model.
- *FN* represents the number of positive samples incorrectly predicted as negative by the model.

AUC measures the model's performance across all possible classification thresholds, providing a comprehensive reflection of the model's ability to distinguish between hate speech and non-hate speech. Compared to evaluation metrics based on a single threshold, AUC is not affected by a specific threshold and offers a more holistic assessment of the model's performance.

4.2.2 Experimental Parameters

The performance of deep learning models is highly dependent on experimental parameters and environment settings. The model parameters we used are shown in the following Table 2. In this study, all experiments were conducted on an NVIDIA GeForce RTX 3070 GPU, using PyTorch version 2.5.0. The learning rate was set to 1×10^{-4}, the batch size to 16, and the number of training epochs to 10. The residual ratio of the low-rank feature adapter was set to 0.2, and the rank of the low-rank adapter was set to 1. We used AdamW as the optimizer with a weight decay of 1×10^{-4}. The ViT-L/14 model is used as the image encoder.

Table 2: Model parameter settings

Parameter	Value
CLIP model	ViT-L/14
Batch size	16
Learning rate	1×10^{-4}
Epochs	10
Optimizer	AdamW
Weight decay	1×10^{-4}
Residual ratio	0.2
Rank of low-rank adapter	1

4.3 Comparison with Baselines

We compare our model (LABF) with some strong baselines. The following is a brief description of the model: On the unimodal approach we use CLIP image encoder and CLIP text encoder as image-based and text-based approaches. On multimodal methods we have selected MOMENTA [31], Hate-CLIPper [32], ISSUES [7], MemCLIP [22] and our method LABF for comparison. The relevant results are shown in Table 3.

Table 3: The comparison results between LABF and existing methods, with results marked with an asterisk (∗) imported from the literature

Models	PrideMM Acc	PrideMM AUC	HarMeme Acc	HarMeme AUC
CLIP Text-Only	69.6	76.9	78.8	86.6
CLIP Image-Only	73.0	79.0	79.1	91.0
CLIP	72.0	79.1	81.1	89.4
MOMENTA [31]*	72.2	78.6	82.4	87.9
Hate-CLIPper [32]*	75.5	83.1	83.9	91.9
ISSUES [7]*	74.7	84.2	81.6	92.8
MemeCLIP [22]	75.8	84.5	83.6	93.2
LABF	76.7	85.0	84.5	93.9

The overall performance comparison of LABF is shown in Table 2. We can see that the performance of CLIP Text-Only is lower than that of CLIP Image-Only, which indicates that CLIP Image-Only can extract representations that capture both the image and overlaid text semantics. The performance of unimodal models on both datasets is worse than most multimodal models, demonstrating the necessity of using multimodal processing for hateful memes classification.

Although MOMENTA, Hate-CLIPper, and ISSUES all use the CLIP model as the image and text encoder, they freeze the CLIP parameters during training, which limits the model's ability to learn features from new data and makes it difficult to adapt to subtle changes in data when faced with complex multimodal tasks. Although MemeCLIP introduces Feature Adapters to learn features from new data while retaining CLIP's prior knowledge, its use of element-wise multiplication as a feature fusion strategy fails to fully promote interaction between image and text features, which somewhat affects the model's ability to capture cross-modal semantic relationships and limits its performance in multimodal classification tasks.

Our model achieves the best performance on both datasets. Experimental results show that Low-Rank Adapters play a key role in enhancing the CLIP model's adaptability to small-scale datasets. By employing low-rank matrix factorization techniques, Low-rank Adapters not only effectively reduce the model's parameter count but also significantly decreases the risk of overfitting, enabling the model to achieve more robust learning and generalization on limited data. The BiGFF mechanism dynamically adjusts the interaction weights between text and image features, significantly improving the accuracy of cross-modal information fusion.

4.4 Ablation Experiments

4.4.1 Ablation Study of Each Module

To evaluate the effectiveness of the proposed model and analyze the contribution of each component to the overall performance, we conducted comprehensive ablation experiments. The experimental results are shown in Table 4.

Table 4: Results of ablation experiments

Module			PrideMM		HarMeme	
LR	**BiGFF**	**CC**	**Acc**	**AUC**	**Acc**	**AUC**
			72.0	79.1	81.1	89.4
✓			75.5	84.6	84.5	92.0
	✓		75.3	84.0	84.7	93.0
		✓	75.7	84.4	84.5	92.5
✓		✓	74.4	83.5	84.7	92.4
	✓	✓	75.7	84.3	83.1	92.5
✓	✓		75.3	84.0	81.9	91.6
✓	✓	✓	76.7	85.0	84.5	93.9

In this study, we evaluated the impact of the Low-Rank Adapter (LR), Bidirectional Gated Feature Fusion module (BiGFF), and Cosine Classifier (CC) on the performance of the multimodal hate speech detection model through ablation experiments.

The results show that the LR module significantly improved the model's accuracy and AUC, particularly in enhancing feature representation and model adaptability. Since we froze the parameters of the CLIP model

and fine-tuned it through low-rank adapters, the low-rank adapters effectively enhanced the model's learning capability in the hateful memes classification task. They demonstrated significant advantages in feature learning and task adaptability. When the low-rank adapters were removed, the model lost this fine-grained feature processing and task adaptability, resulting in a substantial performance drop.

BIGFF further improves the model's performance by effectively fusing multimodal features, especially in the HarMeme model, where its performance is particularly outstanding. BiGFF utilizes a gating mechanism to dynamically adjust image and text features, making the interaction between the two modalities more refined. In BiGFF, the dynamic weights generated by the gating network can modulate the features based on their similarity and importance, effectively enhancing the relevance and fusion effect of features from different modalities. Ablation experiment results show that after removing BiGFF, the model's ability to fuse multimodal features significantly decreases, leading to weakened feature interaction, which in turn impacts performance improvement.

The CC module compares the joint image-text features with class representations using cosine similarity, improving classification performance. It plays a key role in multimodal hate speech detection tasks, particularly when combined with the low-rank adapter (LR) and BiGFF. This combination optimizes feature fusion and further improves classification accuracy. Overall, the synergy of LR, BIGFF, and CC enables the model to achieve optimal performance in both accuracy and AUC, demonstrating the critical role of these modules in multimodal hate speech detection tasks.

4.4.2 The Effect of Image Size Variation on Model Robustness

To validate the model's effectiveness and robustness across different scenarios, experiments were conducted on two datasets, PrideMM and HarMeme, with different input image sizes (224×224, 128×128, 256×256, 512×512). The experimental results of Table 5 show the model's performance stability and adaptability across different datasets and image sizes. While variations in image size had a minor impact on the model's performance, the overall performance maintained a high level. These results show the model's ability to maintain strong performance when faced with different datasets and input sizes, thereby validating its effectiveness and robustness in diverse scenarios.

Table 5: Performance of the model at different image sizes

Image size	PrideMM Acc	PrideMM Auc	HarMeme Acc	HarMeme Auc
(224×224)	76.7	85.0	84.5	93.9
(128×128)	75.3	83.2	83.1	93.0
(256×256)	75.1	83.4	83.9	93.5
(512×512)	74.2	83.8	83.6	93.4

4.4.3 Performance of Low-Rank Adapter across Different Modalities

In this experiment, we evaluated the performance of Low-Rank Adapter across different modalities. As shown in Table 6, on the PrideMM dataset, the model performance is comparable when the Low-Rank Adapter is applied to either the image modality or the text modality individually. When the Low-Rank Adapter is applied to both modalities simultaneously, a substantial enhancement in model performance is observed. These results indicate that simultaneous adaptation of both modalities facilitates more effective feature optimization, thereby enhancing the overall performance of the model.

On the HarMeme dataset, while the accuracy across all configurations remains comparable, the AUC values are lower when the Low-Rank Adapter is applied to either the image modality or the text modality individually. When the Low-Rank Adapter is applied to both modalities, the AUC value increases. This demonstrates that simultaneously fine-tuning both image and text modalities can better enhance the model's learning effectiveness for individual modalities, further boosting the model's robustness and performance.

Table 6: Performance of low-rank adapter across different modalities

Adapter type	PrideMM Acc	PrideMM Auc	HarMeme Acc	HarMeme Auc
Image adapter	74.4	84.1	84.2	93.1
Text adapter	74.4	83.9	84.5	92.7
Low rank adapter	76.7	85.0	84.5	93.9

4.4.4 Comparison of Different Multimodal Feature Fusion Methods

To further evaluate the effectiveness of the BIGFF fusion mechanism, we compared the performance of different feature fusion methods on the PrideMM and HarMeme datasets. The fusion methods mainly include feature concat, element-wise multiplication, DualCo-Attention [2], and Combiner [35].

The experimental results of Table 7 show that BiGFF exhibits significant advantages in multimodal tasks. Traditional methods such as Concat and element-wise multiplication have similar performance but fail to effectively capture the complex interactions between image and text features, resulting in mediocre performance on both datasets. Dual Co-Attention enhances feature interaction between text and image by introducing an attention mechanism. However, the inclusion of the attention mechanism increases the model's computational complexity and optimization difficulty, leading to poorer performance on the HarMeme dataset. Combiner provides effective feature fusion but lacks the dynamic adjustment capability of BiGFF, which leads to insufficient exploitation of modality potentials when handling complex interactions between image and text.

Table 7: Comparison of different multimodal feature fusion methods

Fusion method	PrideMM Acc	PrideMM Auc	HarMeme Acc	HarMeme Auc
Concat	75.7	83.7	82.2	92.1
Element-wise multiplication	75.0	83.7	83.1	92.8
DualCo-Attention [2]	76.9	84.3	81.1	91.7
Combiner [35]	75.1	84.5	85.3	91.6
BIGFF	76.7	85.0	84.5	93.9

Our proposed BiGFF method achieves the best performance on both the PrideMM and HarMeme datasets. BiGFF dynamically adjusts the fusion strength of image and text features through a bidirectional gating mechanism, enhancing the interaction and information flow between modalities. This characteristic enables BiGFF to perform excellently in hateful memes classification task.

4.5 Parameter Sensitivity Analysis

During the training process of a deep learning model, the choice of hyperparameters has a crucial impact on the performance of the model. Different hyperparameter settings may significantly change the convergence speed, final performance, and computational efficiency of the model. Therefore, to further

improve the performance of the model in the multimodal hate speech detection task, we designed a series of key parameter experiments focusing on the analysis of three core hyperparameters: the learning rate, the feature fusion ratio, and the rank of the Low-Rank Adapters. By systematically tuning these hyperparameters and evaluating their effects on model accuracy, AUC metrics.

4.5.1 Learning Rate

Appropriately setting the learning rate is critical to improving model performance. Keeping other parameters constant, the results of experiment in Fig. 3 show that the model performs best on all datasets when the learning rate is 1.0×10^{-4}. The AUC values of 85.0 and 93.9 for PrideMM and HarMeme are the best performances. When the learning rate is 1.0×10^{-5}, the model performance decreases significantly, with AUC of 83.2 and 93.1 on the two datasets. The model performance improves with the gradual increase of the learning rate, but when the learning rate is more than 1.0×10^{-4}, the model's performance starts to decrease, which may lead to instability in the training process. Therefore, the optimal learning rate should be set to 1.0×10^{-4}.

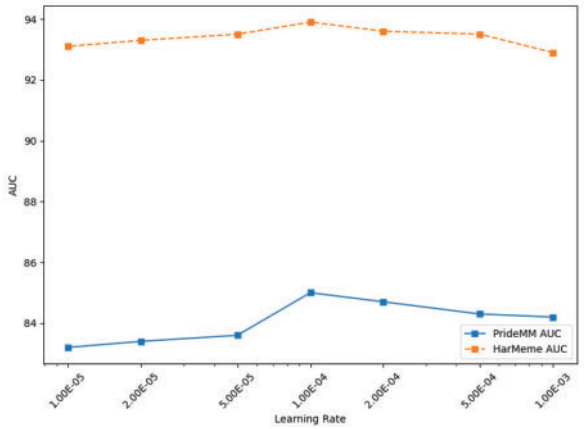

Figure 3: Trend of AUC at different learning rates

4.5.2 Residual Ratio

We analyzed the impact of the low-rank feature adapter residual ratio α on model performance. The experimental results in Fig. 4 show that the residual ratio α of the low-rank feature adapter plays a key role in balancing the CLIP prior features and the new features learned by the adapter, significantly affecting the model's performance. When $\alpha = 0.1$, the model mainly relies on CLIP pretrained features and the tuning effect of the adapter is weak, resulting in relatively low AUC values (PrideMM: 84.3, HarMeme: 93.7). As α increases to 0.2, the role of the adapter increases and the feature representation capability is optimized, with AUC values improving to 85.0 and 93.9 for optimal performance. However, as α continues to increase to 0.3 and above, the model performance begins to degrade due to over-tuning of the adapter, which introduces noise and weakens the effectiveness of the features. Therefore, $\alpha = 0.2$ was determined to be the optimal configuration, allowing the model to achieve efficient adaptation to downstream data while retaining CLIP priori knowledge, thus improving classification performance.

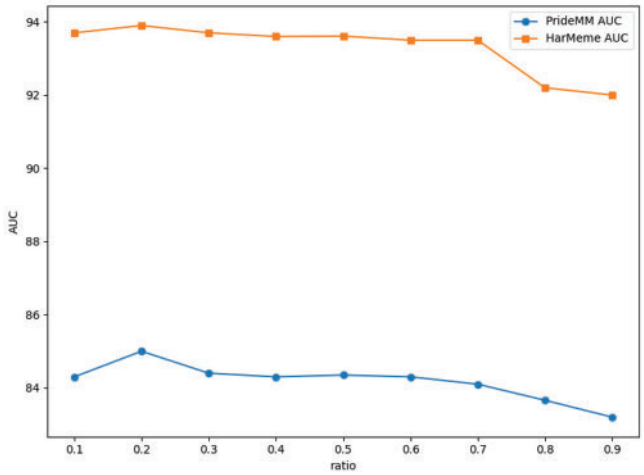

Figure 4: Trends in AUC with different residual ratios

4.5.3 Rank of Adapter

In the Low-Rank Adapter, the update of the matrix weights ΔW is achieved through the product of two low-rank matrices, $A \in \mathbb{R}^{d \times r}$ and $B \in \mathbb{R}^{r \times k}$, where the rank r controls the amount of information the Low-Rank Adapter can learn and represent. A higher rank implies larger dimensions for the low-rank matrices A and B, enabling the Low-Rank Adapter to capture more complex patterns and weight adjustments, thus enhancing the model's adaptability [10]. However, as the rank increases, the number of parameters and computational complexity also grow, which may lead to performance degradation and an increased risk of overfitting, especially when the data size is limited. The number of trainable parameters introduced by $r \times (d + k)$, meaning that higher ranks require more memory and computational resources, potentially impacting the model's performance. Therefore, we explore the impact of varying the rank of the low-rank matrices in the Low-Rank Adapter on the model's performance.

The experimental results in Table 8 show that the Low-Rank Adapter rank significantly affects the model performance. As the rank increases, the accuracy and AUC values on the PrideMM dataset tend to decrease, and the model performance regresses especially when the rank is 4 and above. Specifically, at rank 1, the PrideMM and HarMeme datasets have the best performance with AUCs of 85.0 and 93.9, whereas at rank 16, although the accuracy of the HarMeme dataset is maintained at 84.5, its AUC value decreases to 91.8, and the performance of PrideMM is further degraded. This suggests that higher-ranked adapters may lead to an increase in computational complexity and introduce the risk of overfitting. In summary, a lower rank is the optimal choice for this task, as it can ensure higher performance while avoiding excessive adapter size and computational overhead.

Table 8: Effect of different low-rank adapter ranks on model performance

Rank	PrideMM		HarMeme		Adapter size
	Acc	AUC	Acc	AUC	
1	76.7	85.0	84.5	93.9	3.2 k
2	76.5	83.8	85.0	93.6	5.3 k
4	74.8	84.8	84.5	92.5	9.5 k

(Continued)

Table 8 (continued)

Rank	PrideMM		HarMeme		Adapter size
	Acc	AUC	Acc	AUC	
8	76.3	84.3	83.6	93.8	18.0 k
16	74.6	83.7	84.5	91.8	34.8 k

To further validate the impact of different adapter designs on model performance, we conducted a comparative experiment to examine the differences in model performance between low-rank adapter and clip-adapter [4].

The experimental results in Table 9 show that low-rank adapter outperform clip-adapter on the PrideMM and HarMeme datasets. The low-rank adapter achieves 0.850 on PrideMM and 0.939 on HarMeme, outperforming the clip-adapter with scores of 0.846 and 0.928. Additionally, low-rank adapter has only 3.2 k parameters, significantly fewer than Clip-adapter's 524 k, demonstrating a clear parameter advantage. This indicates that low-rank adapter can significantly enhance model performance while maintaining a lower computational cost, validating the effectiveness of LoRA technology in adapter design.

Table 9: The impact of two different adapters on model performance

Adapter	PrideMM		HarMeme		Adapter size
	Acc	AUC	Acc	AUC	
Low-Rank Adapter	0.767	0.850	0.845	0.939	3.2 k
Clip-Adapter [4]	0.759	0.846	0.839	0.928	524 k

4.5.4 Sensitivity Analysis of the BiGFF Module's Scaling Factors

In this experiment, we conducted a sensitivity analysis of the learnable scaling factors λ and μ in the BiGFF module to explore their impact on the model's performance on the PrideMM and HarMeme datasets. As shown in Table 10, when the initialized values of λ and μ are 0, the model achieves the best performance with AUC values of 0.85 and 0.939 on the PrideMM and HarMeme datasets. As the initialization values of λ and μ increase, the model performance gradually decreases. When the initialization values of λ and μ are 2, the AUC values of the model on the two datasets drop to 0.841 and 0.923. This trend suggests that strong feature interactions at the beginning of model training weaken the effect of feature fusion and lead to a decrease in model performance.

Table 10: Sensitivity analysis of BiGFF module's scaling factors on model performance

λ, μ	PrideMM		HarMeme	
	Acc	AUC	Acc	AUC
(0, 0)	0.767	0.850	0.845	0.939
(0.5, 0.5)	0.761	0.845	0.853	0.932
(1, 1)	0.757	0.842	0.853	0.930
(2, 2)	0.755	0.841	0.831	0.923

5 Conclusions and Future Work

In this study, we propose a multimodal classification framework that fuses Low-Rank Adapter with Bidirectional Gated Feature Fusion. By introducing Low-Rank Adapter layers into the CLIP model, we achieve efficient migration of pre-trained knowledge; The designed bidirectional gated network can dynamically adjust the strength of the interaction between image and text features, which significantly improves the ability to capture cross-modal semantics. Experimental results show that the proposed model outperforms the existing baseline model on both datasets.

In future work, we will further explore and optimize the model from two directions. On one hand, we plan to introduce dynamic adaptation mechanisms and explore more efficient low-rank parameter decomposition strategies to further enhance the flexibility and computational performance of the LABF framework. On the other hand, we will introduce adversarial learning mechanisms to enhance the model's robustness and generalization ability in the presence of adversarial examples or high-noise scenarios. We consider leveraging adversarial training methods and adversarial example generation techniques. These approaches offer new possibilities for building robust representations in multimodal models. We believe that by incorporating adversarial noise perturbation mechanisms tailored for both image and text modalities, the LABF model is expected to maintain stable performance in more complex and dynamic real-world scenarios.

Acknowledgement: We would like to thank the editors and reviewers for their valuable work.

Funding Statement: This research was supported by the Funding for Research on the Evolution of Cyberbullying Incidents and Intervention Strategies (24BSH033) and Discipline Innovation and Talent Introduction Bases in Higher Education Institutions (B20087).

Author Contributions: Conceptualization, Youwei Huang and Han Zhong; methodology, Youwei Huang; formal analysis, Youwei Huang and Yijie Peng; writing—original draft preparation, Youwei Huang and Cheng Cheng; writing—review and editing, Youwei Huang and Han Zhong; funding acquisition: Han Zhong. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the Han Zhong upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. de Hermida PCQ, dos Santos EM. Detecting hate speech in memes: a review. Artif Intell Rev. 2023;56(11):12833–51. doi:10.1007/s10462-023-10459-7.
2. Hossain E, Sharif O, Hoque MM, Preum SM. Deciphering hate: identifying hateful memes and their targets. arXiv:2403.10829. 2024.
3. Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, et al. Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning; 2021 Jul 18–24; Online. p. 8748–63.
4. Gao P, Geng S, Zhang R, Ma T, Fang R, Zhang Y, et al. Clip-adapter: better vision-language models with feature adapters. Int J Comput Vis. 2024;132(2):581–95. doi:10.1007/s11263-023-01891-x.
5. Yu T, Lu Z, Jin X, Chen Z, Wang X. Task residual for tuning vision-language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2023; 2023 Jun 17–24; Vancouver, BC, Canada. p. 10899–909.

6. Zhang R, Zhang W, Fang R, Gao P, Li K, Dai J, et al. Tip-adapter: training-free adaption of clip for few-shot classification. In: 2022 European Conference on Computer Vision; 2022 Oct 23–27; Tel Aviv, Israel. p. 493–510.
7. Burbi G, Baldrati A, Agnolucci L, Bertini M, Del Bimbo A. Mapping memes to words for multimodal hateful meme classification. In: Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision; 2023 Oct 2–6; Paris, France. p. 2832–6.
8. Kiela D, Firooz H, Mohan A, Goswami V, Singh A, Ringshia P, et al. The hateful memes challenge: detecting hate speech in multimodal memes. Adv Neural Inf Process Syst. 2020;33:2611–24.
9. Long HW, Li H, Cai W. CoinCLIP: a multimodal framework for evaluating the viability of memecoins in the Web3 ecosystem. arXiv:2412.07591. 2024.
10. Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, et al. Lora: low-rank adaptation of large language models. arXiv:2106.09685. 2021.
11. Karimi Mahabadi R, Henderson J, Ruder S. Compacter: efficient low-rank hypercomplex adapter layers. Adv Neural Inf Process Syst. 2021;34:1022–35.
12. Wang Y, Markov I. CLTL@ Multimodal hate speech event detection 2024: the winning approach to detecting multimodal hate speech and its targets. In: Proceedings of the 7th Workshop on Challenges and Applications of Automated Extraction of Socio-Political Events From Text (CASE 2024); 2024; St. Julians, Malta. p. 73–8.
13. Loureiro D, Rezaee K, Riahi T, Barbieri F, Neves L, Anke LE, et al. Tweet insights: a visualization platform to extract temporal insights from twitter. arXiv:2308.02142. 2023.
14. Liu Z, Hu H, Lin Y, Yao Z, Xie Z, Wei Y, et al. Swin transformer v2: scaling up capacity and resolution. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun 18–24; New Orleans, LA, USA. p. 12009–19.
15. Shi X, Mueller J, Erickson N, Li M, Smola A. Multimodal automl on structured tables with text fields. In: 8th ICML Workshop on Automated Machine Learning (AutoML); 2021; Online. p. 1–15.
16. Gomez R, Gibert J, Gomez L, Karatzas D. Exploring hate speech detection in multimodal publications. In: Proceedings of the 2020 IEEE/CVF Winter Conference on Applications of Computer Vision; 2020 Mar 1–5; Snowmass Village, CO, USA. p. 1470–8.
17. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA. p. 2818–26.
18. Velioglu R, Rose J. Detecting hate speech in memes using multimodal deep learning approaches: prize-winning solution to hateful memes challenge. arXiv:2012.12975. 2020.
19. Li LH, Yatskar M, Yin D, Hsieh CJ, Chang KW. Visualbert: a simple and performant baseline for vision and language. arXiv:1908.03557. 2019.
20. Baldrati A, Agnolucci L, Bertini M, Del Bimbo A. Zero-shot composed image retrieval with textual inversion. In: Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision; 2023 Oct 2–6 2023; Paris, France. p. 15338–47.
21. Mei J, Chen J, Lin W, Byrne B, Tomalin M. Improving hateful meme detection through retrieval-guided contrastive learning. arXiv:2311.08110. 2023.
22. Shah SB, Shiwakoti S, Chaudhary M, Wang H. Memeclip: leveraging clip representations for multimodal meme classification. arXiv:2409.14703. 2024.
23. Zhou K, Yang J, Loy CC, Liu Z. Learning to prompt for vision-language models. Int J Comput Vis. 2022;130(9):2337–48. doi:10.1007/s11263-022-01653-1.
24. Chen G, Yao W, Song X, Li X, Rao Y, Zhang K. Plot: prompt learning with optimal transport for vision-language models. arXiv:2210.01253. 2022.
25. Lippe P, Holla N, Chandra S, Rajamanickam S, Antoniou G, Shutova E, et al. A multimodal framework for the detection of hateful memes. arXiv:2012.12871. 2020.
26. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst. 2015;28:91–9.

27. Devlin J, Chang MW, Lee K, Toutanova K. 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; 2019 Jun 2–7; Minneapolis, MN, USA. Stroudsburg PA USA: Association for Computational Linguistics; 2019. p. 4171–86.
28. Li Z. Codewithzichao@ DravidianLangTech-EACL2021: exploring multimodal transformers for meme classification in Tamil language. In: Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages; 2021; Kyiv, Ukraine. p. 352–6.
29. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA. p. 770–8.
30. Conneau A, Khandelwal K, Goyal N, Chaudhary V, Wenzek G, Guzmán F, et al. Unsupervised cross-lingual representation learning at scale. arXiv:1911.02116. 2019.
31. Pramanick S, Sharma S, Dimitrov D, Akhtar MS, Nakov P, Chakraborty T. MOMENTA: a multimodal framework for detecting harmful memes and their targets. arXiv:2109.05184. 2021.
32. Kumar GK, Nandakumar K. Hate-CLIPper: multimodal hateful meme classification based on cross-modal interaction of CLIP features. arXiv:2210.05916. 2022.
33. Hossain E, Sharif O, Hoque MM, Preum SM. Align before attend: aligning visual and textual features for multimodal hateful content detection. arXiv:2402.09738. 2024.
34. Pramanick S, Dimitrov D, Mukherjee R, Sharma S, Akhtar MS, Nakov P, et al. Detecting harmful memes and their targets. arXiv:2110.00413. 2021.
35. Baldrati A, Bertini M, Uricchio T, Del Bimbo A. Effective conditioned and composed image retrieval combining clip-based features. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun 18–24; New Orleans, LA, USA. p. 21466–74.