YassirEA at MAHED 2025: Fusion-Based Multimodal Models for Arabic Hate Meme Detection

Yassir El Attar
Institute of Natural Language Processing, University of Stuttgart
yassir.el.attar@gmail.com

Abstract

We present our system for the MAHED 2025 Shared Task on Arabic Hate Meme Detection (subtask 3), a binary classification task to determine whether a multimodal meme containing Arabic text and an image conveys a hateful message. Our approach uses multimodal fusion combining a visual encoder and an Arabic text encoder. We explored four fusion strategies—transformer fusion, early fusion, cross-attention, and bilinear fusion—and found transformer fusion offered the best single-model trade-off, while an ensemble of all four achieved the highest score. To address the severe class imbalance (90.05% not-hate vs. 9.95% hate), we applied class-weighted loss, focal loss, strong regularization, and light augmentation. Our best submission reached a macro-F1 score of **0.75** on the gold test set.

1 Introduction

Social media enables rapid information sharing but also accelerates the spread of harmful content, including hate speech. While text-only hate speech detection is well studied, much hateful content now appears in **multimodal formats**, such as memes, which combine text and images into a single communicative unit. These memes often use humor, irony, or cultural symbols to mask or amplify harmful messages, making automated detection challenging (Kiela et al., 2021; Boishakhi et al., 2021). Figure 1 shows examples of Arabic memes from the two classes (*hate* and *not-hate*), illustrating the diversity in visual style and text content.

The **MAHED 2025 Shared Task** (Zaghouani et al., 2025) targets hateful meme detection in Arabic, a language with rich morphology, diverse dialects, and high orthographic variation. Memes may contain Modern Standard Arabic, dialectal Arabic, or a mix, with images referencing culturally specific or political contexts (Mubarak et al., 2023). These factors, along with OCR errors, slang, and stylized fonts, complicate feature extraction. Modeling the interplay between Arabic text and images requires fine-grained cross-modal alignment, motivating our exploration of multiple multimodal fusion strategies.

Figure 1: Examples of *hate*/*not-hate* memes from the **Evaluation-phase test split**.

We address the task under two constraints: a small, imbalanced dataset and the need for effective multimodal fusion. Using state-of-the-art encoders for text and vision, we compare four fusion mechanisms and evaluate an ensemble.

This work makes three main contributions:

1. We provide a systematic comparison of four fusion strategies for Arabic multimodal hate detection.

2. We conduct an in-depth analysis of strategies to mitigate extreme class imbalance, including class-weighted loss, focal loss, and multimodal augmentation.

Proceedings of The Third Arabic Natural Language Processing Conference, pages 608–614

3. We release a public, reproducible system design[1] that can serve as a baseline for future Arabic multimodal classification tasks.

2 Background

Detecting hate speech in multimodal content has become a major research area, especially following the release of the *Hateful Memes* benchmark, which exposed the limitations of unimodal systems in handling cross-modal semantics (Kiela et al., 2021). Subsequent work has explored a range of fusion techniques, including early fusion (concatenating text and image embeddings before classification) (Galanakis et al., 2025), late fusion (combining predictions from unimodal models) (Snoek et al., 2005), and intermediate, attention-based approaches such as cross-attention and co-attention (Lu et al., 2017, 2019; Chen et al., 2020; Zhang et al., 2024).

In Arabic NLP, hate speech detection has mostly focused on text-only methods (Mubarak et al., 2023; Al-Saqqa et al., 2024) using pretrained language models such as AraBERT (Antoun et al., 2020), CAMeLBERT (Inoue et al., 2021), and MARBERTv2 (Abdul-Mageed et al., 2021). Vision–language pretraining models such as CLIP (Radford et al., 2021), SigLIP (Zhai et al., 2023), and Swin Transformer (Wang and Markov, 2024) have also shown promise for multimodal classification. However, their effectiveness for Arabic multimodal hate detection remains underexplored.

3 System Overview

In preliminary experiments on the development set, we found that combining MARBERTv2 for text with CLIP-Large for images performed best. Our final system is therefore built on this pairing, with the overall architecture described in Section 3.1. We also experimented with a uni-modal approach where each modality is used separately for the predictions (details can be found in Appendix E)

3.1 Model Components

Figure 2 illustrates the overall architecture of our system. The input meme consists of an image and its corresponding Arabic text. The image is processed by a visual encoder (CLIP-Large), producing image embeddings, while the text is processed by an Arabic text encoder (MARBERTv2) to produce text embeddings. These embeddings are then fed into one of four fusion mechanisms—transformer fusion, early concatenation, cross-attention, and bilinear pooling—which learn joint multimodal representations. The outputs of all fusion models are combined in an ensemble module that produces the final prediction as either *hate* or *not-hate*.

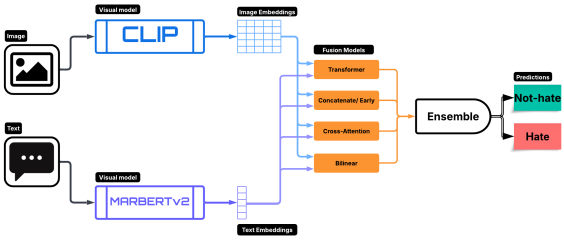

Figure 2: Framework overview

Unimodal Representations. We process the meme text[2] using **MARBERTv2**, a transformer-based language model pretrained on large-scale Arabic text from social media. We take the final hidden state of the `[CLS]` token as the text embedding.

For the image, we use **CLIP-Large (ViT-L/14)** (Radford et al., 2021) to generate visual features. We take the pooled output from CLIP's image encoder as the image embedding.

3.2 Fusion Mechanisms

We explore four fusion strategies, all of which fall under early or intermediate fusion: the text embedding $\mathbf{t} \in \mathbb{R}^{d_t}$ from the text encoder and the image embedding $\mathbf{v} \in \mathbb{R}^{d_v}$ from the vision encoder are merged into a joint representation.

Concatenation (Early Fusion). The text and image embeddings are concatenated into a single vector and passed through a feed-forward layer with ReLU activation and dropout before classification; see Eq. (1) in Appendix C. Here, $[\mathbf{t}; \mathbf{v}]$ denotes concatenation, W and W_o are weight matrices, and $\mathbf{b}$ and $\mathbf{b}_o$ are biases.

Transformer Fusion (Single-Stream). A lightweight transformer jointly processes projected text ($\mathbf{t}$) and image ($\mathbf{v}$) embeddings of equal dimension d, augmented with modality type embeddings. The two-token sequence passes

[1] https://github.com/YassirELATTAR/task3-mahed2025

[2] The extracted text was provided as part of the task data.

through L self-attention layers, and the pooled token is classified with a small MLP.[3]

Cross-Attention (Dual-Stream). Two single-head cross-attention blocks let text attend to image features and vice versa, aligning modalities more explicitly than concatenation but typically requiring more data to generalize.

Bilinear Fusion. Multimodal Compact Bilinear (MCB) pooling (Fukui et al., 2016) models multiplicative interactions between $\mathbf{t}$ and $\mathbf{v}$ in a compressed space, enabling richer feature combinations at the cost of higher overfitting risk on small datasets.

3.2.1 Ensemble

We combine the predictions of all fusion models using:

- **Majority Vote:** Label predicted by most models.
- **Equal Weighted:** Mean-pooling of class probabilities before selecting the argmax.
- **Transformer-Weighted:** Weighted average giving higher weight to transformer fusion[4].

3.3 Dealing with Imbalance

A major challenge in this task is the severe class imbalance in the training data (90.05% *not-hate* vs. 9.95% *hate*). To address this, we experimented with several training-time strategies.

Class-Weighted Training Loss. We use weighted cross-entropy with inverse-frequency class weights; see Eq. (2) in Appendix C. This increases the penalty for errors on the minority class.

Focal Loss. We also test focal loss (Lin et al., 2018) to focus more on hard examples (Eq. (3) in Appendix C), where γ controls hard-example emphasis and α is set to the minority-class prior.

Regularization. To reduce overfitting to the majority class, we applied stronger dropout (0.3 in encoders, 0.2 in fusion layers), weight decay (10^{-4}), and early stopping (patience 5).

Targeted Data Augmentation. To balance the dataset, we augmented the *hate* class with both modified images and texts. For images, we applied rotation, scaling, perspective warp, color jitter, gamma adjustment, noise/blur, geometric distortions, shadows/fog, and crop–resize. For text, we used OCR-extracted text from augmented images (70% probability when confidence was high), synonym replacement, light character dropout, and cautious AR→EN→AR back-translation. We designed the augmentation to preserve the original semantic intent. We paired augmented images and text in three ways: (i) replacing the text with the newly extracted text, (ii) appending new text to the original, and (iii) substituting a few words without altering the meaning. (We show a few examples in Appendix B.)

4 Experimental Setup

4.1 Data and Evaluation

The task is to determine whether a multimodal meme—comprising an image and embedded Arabic text—conveys a hateful message (*hate*) or not (*not hate*). This phenomenon often involves *meaning multiplication*: even if neither the text nor the image alone is hateful, their combination can create a hateful meaning. Effective fusion of the two modalities is therefore crucial, and in this work we explore different fusion strategies.

We use the official splits from the Prop2Hate-Meme dataset (Alam et al., 2024b,a), which follow the shared task protocol for training, development, and testing. The training split is highly imbalanced, with 90.05% *not-hate* and only 9.95% *hate* examples. This motivates the imbalance-handling strategies described in Section 3.3. No external labeled data are used. The official evaluation metric for the shared task, and for all our experiments, is **Macro-F1**, which is preferred over accuracy because it balances performance across classes in the presence of severe class imbalance.

4.2 Training and Evaluation

We trained the following models in our experiments:

- **MARBERTv2** (Abdul-Mageed et al., 2021) as the Arabic text encoder[5].
- **CLIP-Large (ViT-L/14)** (Radford et al., 2021) as the visual encoder[6].

[3] This was the strongest single-model method in preliminary validation.

[4] This choice is based on its stronger validation performance compared to other models.

[5] https://huggingface.co/UBC-NLP/MARBERTv2

[6] https://huggingface.co/openai/clip-vit-large-patch14

Fusion	Accuracy	Macro-F1 (Test)	Macro-F1 (Gold)
Ensemble (All)	0.90	0.72	**0.75**
Transformer	0.91	0.72	0.75
Concatenation	0.89	0.74	0.73
Cross-Attn.	0.88	0.69	0.68
Bilinear	0.89	0.63	0.66

Table 1: Performance on evaluation-phase test (Test) and official leaderboard (Gold) splits. Ensemble gain over Transformer = +0.005 on Gold.

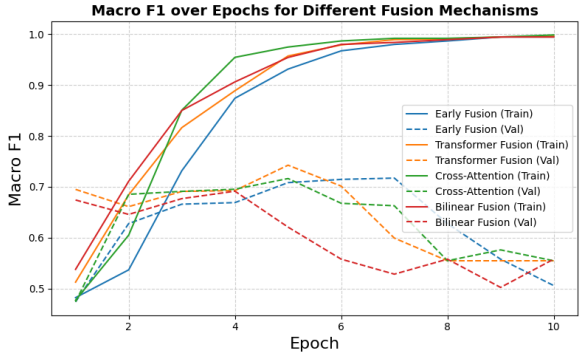

Figure 3: Macro-F1 progression across epochs on Train (solid) and Development (dotted). *Takeaway:* Transformer fusion is the most stable and highest-performing; bilinear overfits quickly.

- Four fusion architectures: concatenation (early fusion), transformer fusion, cross-attention (dual-stream), and bilinear fusion.
- An ensemble combining the predictions of all four fusion models.

We trained all models on the official train split and tuned them on the development set, using **Macro-F1** as the model selection criterion. Details of the hyperparameters are reported in Appendix D.

5 Results

Table 1 presents the main results on MAHED Subtask 3. We report **Macro-F1** on the test split provided during the evaluation phase, and **Macro-F1*** on the gold test set. The latter corresponds to the official leaderboard score. Macro-F1 is the primary evaluation metric of the shared task because it balances performance across classes in the presence of severe class imbalance (Section 3.3).

Figure 3 visualizes the progression of Macro-F1 over training epochs for each fusion mechanism on the train and dev splits. To better illustrate the overfitting behaviour, we train for 10 epochs without early stopping, while keeping all other hyperparameters the same.

Observations. Transformer fusion offers the best single-model trade-off between capacity and stability. The ensemble slightly improves Macro-F1 (+0.005 on the test split) but at the cost of a small drop in accuracy. Cross-attention underperforms transformer fusion, likely due to limited training data, while bilinear fusion tends to overfit. For imbalance handling, class-weighted loss yields the most consistent improvements. Focal loss reduces the impact of easy majority-class cases and can slightly improve minority recall, but the gain is marginal. Data augmentation does not improve performance—in fact, the model often overfits to the augmented data, reaching perfect scores on the training set but dropping significantly on dev. A possible explanation is that the augmented samples introduce superficial patterns that the model can exploit without learning meaningful cross-modal interactions.

Example predictions for the two samples shown in Figure 1 are provided in Appendix A.

6 Limitations

Our system depends on pre-extracted texts from memes, which may miss stylized text; sarcasm/irony and culture-specific references remain challenging. The dataset's class imbalance and limited size constrain generalization, with bilinear and cross-attention models prone to overfitting. We did not perform Arabic-specific vision–language pretraining, which could improve alignment.

7 Conclusion

We explored different fusion strategies combining an Arabic text encoder and a visual encoder for Arabic hate meme detection. We find that an ensemble that aggregates the individual predictions is most effective, yielding a Macro-F1 score of **0.75** on the official test set and ranking **second** on the shared task leaderboard. We also examined approaches to mitigate class imbalance, including class-weighted loss, focal loss, and regularization, and find class-weighted loss to be the most effective. Future work could investigate culture-aware prompts and Arabic-focused vision–language pretraining. Our findings can guide the development of future Arabic multimodal hate detection systems.

Acknowledgments

Thanks are extended to the MAHED 2025 organizers and the dataset providers. A heartfelt thanks

also to Prof. Dr. Carina Silberer for her guidance and assistance and to the Institute of Natural Language Processing (IMS), University of Stuttgart, for providing the working environment and GPU server resources used in this work.

References

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Samar Al-Saqqa, Arafat Awajan, and Bassam Hammo. 2024. A survey of hate speech detection for arabic social media: Methods and datasets. *Procedia Computer Science*, 251:224–231.

Firoj Alam, Md Rafiul Biswas, Uzair Shah, Wajdi Zaghouani, and Georgios Mikros. 2024a. Propaganda to hate: A multimodal analysis of arabic memes with multi-agent llms. In *International Conference on Web Information Systems Engineering*, pages 380–390. Springer.

Firoj Alam, Abul Hasnat, Fatema Ahmad, Md. Arid Hasan, and Maram Hasanain. 2024b. ArMeme: Propagandistic content in Arabic memes. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21071–21090, Miami, Florida, USA. Association for Computational Linguistics.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. In *Proceedings of the LREC*.

Fariha Tahosin Boishakhi, Ponkoj Chandra Shill, and Md. Golam Rabiul Alam. 2021. Multi-modal hate speech detection using machine learning. In *2021 IEEE International Conference on Big Data (Big Data)*, page 4496–4499. IEEE.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. *Preprint*, arXiv:1909.11740.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 457–468, Austin, Texas. Association for Computational Linguistics.

Ioannis Galanakis, Rigas Filippos Soldatos, Nikitas Karanikolas, Athanasios Voulodimos, Ioannis Voyiatzis, and Maria Samarakou. 2025. Early and late fusion for multimodal aggression prediction in dementia patients: A comparative analysis. *Applied Sciences*, 15(11).

Go Inoue, Muhammad Abdul-Mageed, and Mahmoud El-Haj. 2021. Camelbert: Transformer-based arabic language models. In *Proceedings of WANLP*.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. 2021. The hateful memes challenge: Detecting hate speech in multimodal memes. *Preprint*, arXiv:2005.04790.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2018. Focal loss for dense object detection. *Preprint*, arXiv:1708.02002.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Preprint*, arXiv:1908.02265.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2017. Hierarchical question-image co-attention for visual question answering. *Preprint*, arXiv:1606.00061.

Hamdy Mubarak, Sabit Hassan, and Shammur Absar Chowdhury. 2023. Emojis as anchors to detect arabic offensive language and hate speech. *Natural Language Engineering*, 29(6):1436–1457.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020.

Cees Snoek, Marcel Worring, and Arnold Smeulders. 2005. Early versus late fusion in semantic video analysis. pages 399–402.

Yeshan Wang and Ilia Markov. 2024. 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)*, pages 73–78, St. Julians, Malta. Association for Computational Linguistics.

Wajdi Zaghouani, Md Rafiul Biswas, Mabrouka Bessghaier, Shimaa Ibrahim, Georgios Mikros, Abul Hasnat, and Firoj Alam. 2025. Overview of the mahed shared task: Multimodal detection of hope and hate emotions in arabic content. In *The Third Arabic Natural Language Processing Conference (Arabic-NLP 2025)*, Suzhou, China. Association for Computational Linguistics.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *Preprint*, arXiv:2303.15343.

Yinghui Zhang, Tailin Chen, Yuchen Zhang, and Zeyu Fu. 2024. Enhanced multimodal hate video detection via channel-wise and modality-wise fusion. In *2024 IEEE International Conference on Data Mining Workshops (ICDMW)*, page 183–190. IEEE.

Appendix

A Example Predictions

Table 2 shows the predictions from different fusion models for the two examples in Figure 1.

Model	Example 1	Example 2
(Ground truth)	not-hate	hate
Concatenation	not-hate	not-hate
Transformer	not-hate	hate
Cross-Attn.	not-hate	hate
Bilinear	not-hate	hate
Ensemble	not-hate	hate

Table 2: Example predictions from different models.

B Augmentation Examples

Figure 4 illustrates examples of image augmentations applied to the *hate* class. The associated text augmentations are shown below each image.

Figure 4: Examples of image augmentations for the *hate* class.

C Additional Modeling Equations

Concatenation (early fusion).

$$\begin{aligned}\mathbf{h} &= \mathrm{ReLU}\big(W[\mathbf{t};\mathbf{v}] + \mathbf{b}\big),\\ \hat{\mathbf{y}} &= \mathrm{softmax}\big(W_o\ \mathrm{Dropout}(\mathbf{h}) + \mathbf{b}_o\big).\end{aligned} \tag{1}$$

Table 3: Baseline summary on the test set (accuracy and macro F1).

Approach	Acc (Weighted)	Macro-F1 (Weighted)	Acc (Focal)	Macro-F1 (Focal)
Text only	0.80	0.67	0.76	0.57
Image only	0.77	0.57	0.77	0.59
Confidence combine	0.78	0.59	0.78	0.55

Table 4: Text-only (Weighted) – classification report (test).

Class	Precision	Recall	F1	Support
not-hate	0.81	0.96	0.88	452
hate	0.74	0.34	0.46	154
Accuracy		0.80		606
Macro avg	0.78	0.65	0.67	606
Weighted avg	0.79	0.80	0.77	606

Weighted cross-entropy.

$$\mathcal{L}_{\mathrm{wCE}} = -\,w_1\, y \log p \;-\; w_0\,(1-y)\,\log(1-p). \tag{2}$$

Focal loss.

$$\begin{aligned}\mathcal{L}_{\mathrm{focal}} = &-\,\alpha\,(1-p)^{\gamma}\, y \log p\\ &-\,(1-\alpha)\,p^{\gamma}\,(1-y)\,\log(1-p).\end{aligned} \tag{3}$$

D Framework Training Details and Hyperparameters

The main hyperparameters used: batch size 16, 40 training epochs, AdamW optimizer, base learning rate 2×10^{-5} with a linear scheduler, and weight decay of 10^{-4}. We applied dropout of 0.3 in encoders and 0.2 in fusion layers, and used early stopping with patience 5 to prevent overfitting.

E Unimodal Experiments

We evaluate three simple baselines: (i) text only, (ii) image only, and (iii) a confidence-based combination of the two unimodal systems (if the two disagree, pick the class from the model with higher softmax confidence). Each is trained/evaluated under class-weighted cross-entropy and Focal Loss. We report test accuracy and macro F1, then the final classification reports.

Table 5: Text-only (Focal) – classification report (test).

Class	Precision	Recall	F1	Support
not-hate	0.77	0.96	0.86	452
hate	0.60	0.18	0.28	154
Accuracy		0.76		606
Macro avg	0.69	0.57	0.57	606
Weighted avg	0.73	0.76	0.71	606

Table 6: Image-only (Weighted) – classification report (test).

Class	Precision	Recall	F1	Support
not-hate	0.78	0.97	0.86	452
hate	0.66	0.18	0.28	154
Accuracy		0.77		606
Macro avg	0.72	0.57	0.57	606
Weighted avg	0.75	0.77	0.71	606

Table 7: Image-only (Focal) – classification report (test).

Class	Precision	Recall	F1	Support
not-hate	0.78	0.97	0.87	452
hate	0.70	0.19	0.30	154
Accuracy		0.77		606
Macro avg	0.74	0.58	0.58	606
Weighted avg	0.76	0.77	0.72	606

Table 8: Confidence-based combination (Weighted) – classification report (test).

Class	Precision	Recall	F1	Support
not-hate	0.78	0.98	0.87	452
hate	0.76	0.19	0.30	154
Accuracy		0.78		606
Macro avg	0.77	0.58	0.59	606
Weighted avg	0.78	0.78	0.72	606

Table 9: Confidence-based combination (Focal) – classification report (test).

Class	Precision	Recall	F1	Support
not-hate	0.77	1.00	0.87	452
hate	0.91	0.14	0.24	154
Accuracy		0.78		606
Macro avg	0.84	0.57	0.55	606
Weighted avg	0.81	0.78	0.71	606