

Towards Social Safety in Digital Platforms: Multimodal Hostile Content Detection and Explanation

Ph.D. Thesis

By

MOHAMMAD ZIA UR REHMAN



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY INDORE**

DECEMBER 2025

Towards Social Safety in Digital Platforms: Multimodal Hostile Content Detection and Explanation

A THESIS

submitted to the

INDIAN INSTITUTE OF TECHNOLOGY INDORE

in partial fulfillment of the requirements for
the award of the degree of

DOCTOR OF PHILOSOPHY

by

MOHAMMAD ZIA UR REHMAN



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY INDORE**

DECEMBER 2025

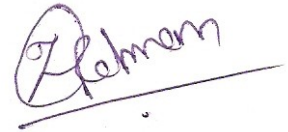


INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

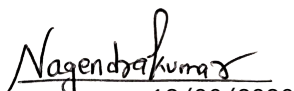
I hereby certify that the work which is being presented in the thesis entitled **Towards Social Safety in Digital Platforms: Multimodal Hostile Content Detection and Explanation** in the partial fulfillment of the requirements for the award of the degree of **Doctor of Philosophy** and submitted in the **Department of Computer Science and Engineering, Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from December 2021 to December 2025 under the supervision of Dr. Nagendra Kumar, Assistant Professor, Indian Institute of Technology Indore, Indore, India.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.


12/06/2026

Signature of the Student with Date
(**Mohammad Zia Ur Rehman**)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.


12/06/2026

Signature of Thesis Supervisor with Date
(**Dr. Nagendra Kumar**)

ACKNOWLEDGEMENT

First and foremost, I express my deepest gratitude to the Almighty Allah for granting me the strength, patience, and perseverance to successfully complete this doctoral journey. I dedicate this thesis to the loving memory of my late mother, whose blessings, sacrifices, and unconditional love continue to guide me in every step of my life. I am profoundly grateful to my father and my family for their unwavering support, encouragement, and constant faith in me throughout the course of my research. Their moral and emotional support has been an invaluable source of motivation during challenging times.

I would like to express my most sincere and heartfelt gratitude to my supervisor, **Dr. Nagendra Kumar**, for his invaluable guidance, continuous encouragement, and insightful feedback throughout my Ph.D. journey. His academic rigor, and constructive criticism played a pivotal role in shaping this research. I am deeply indebted to him for his patience, mentorship, and unwavering support, which greatly contributed to both my professional and personal growth.

I extend my sincere thanks to **Dr. Ranveer Singh** and **Dr. Mohd. Arshad** for their valuable suggestions, critical insights, and constructive feedback at various stages of my research. Their guidance helped refine the direction of this work and significantly improved its quality.

I gratefully acknowledge the financial support provided by the **University Grants Commission (UGC), Government of India**, through the Junior Research Fellowship (JRF) scheme, which enabled me to pursue this research and focus wholeheartedly on my doctoral studies.

I would also like to thank my labmates and colleagues for their cooperation, discussions, and encouragement. The stimulating research environment, collaborative spirit, and camaraderie within the lab made this journey both intellectually enriching and personally fulfilling.

Finally, I extend my gratitude to all those who directly or indirectly contributed to the successful completion of this thesis. Any omissions are entirely unintentional.

Mohammad Zia Ur Rehman
Indian Institute of Technology Indore
December 2025

Dedicated to My Parents

ABSTRACT

The exponential rise of user-generated content on social media has fostered connectivity but also facilitated the proliferation of hostile narratives, ranging from explicit abuse to subtle, implicit forms of hate speech and dark humor. While automated detection systems have advanced, significant gaps remain in addressing low-resource languages, multimodal complexity, and the interpretability of model decisions. This thesis proposes a comprehensive suite of frameworks to detect and understand hostile content across three modalities: text, memes, and video.

In the textual domain, the thesis addresses the problem in low-resource Indic languages. First, it introduces a user-aware framework for abusive content detection in multilingual and code-mixed environments. By integrating social context features, such as user history and post affinity, with cross-lingual textual embeddings, the proposed method significantly enhances detection performance in low-resource settings. Second, to bridge the gap between black-box predictions and human reasoning, the thesis presents X-MuTeST, an explainable framework for hate speech detection in Hindi, Telugu, and English. This work contributes benchmark datasets with token-level human rationales and employs a novel training strategy that combines Large Language Model (LLM) consultation with N-gram-based explainability to improve both plausibility and faithfulness.

Transitioning to static multimodal content, the thesis tackles the semantic complexity of internet memes. It proposes MM-ORIENT, a framework designed to mitigate noise propagation in multimodal fusion. By utilizing cross-modal relation graphs and Hierarchical Interactive Monomodal Attention (HIMA), the model effectively handles multiple downstream tasks, including sarcasm, humor, and offensive content detection, without explicit early-stage interaction. Furthermore, the thesis addresses gender-targeted hostility through a specialized context-aware attention and graph-based framework for misogyny detection, incorporating content-specific features like toxicity and image captions. Addressing the problem of dark humor, the thesis introduces D-Humor, a dataset and reasoning-augmented framework. By leveraging Vision-Language Models (VLMs) with a role-reversal self-loop mechanism, the system generates and refines structured explanations to detect the target and intensity of dark humor, moving beyond simple binary classification.

Finally, the thesis extends into the dynamic domain of video analysis with ImpliHateVid, a benchmark dataset specifically curated for implicit hate speech. To address the subtlety of

implicit hate in videos, a two-stage contrastive learning framework is proposed. This method aligns audio, visual, and textual representations in a shared embedding space, capturing the intricate, context-dependent cues often missed by unimodal systems.

Collectively, this research advances the state-of-the-art in hostile content detection by creating novel datasets for under-resourced domains, developing explainable, noise-resilient, and reasoning-aware multimodal architectures. The proposed frameworks demonstrate robust performance across diverse languages and modalities, offering scalable solutions for fostering safer online environments.

PUBLICATIONS FROM THE THESIS

In Refereed Journals

1. **Mohammad Zia Ur Rehman**, Devraj Raghuvanshi, Umang Jain, Shubhi Bansal, and Nagendra Kumar. “A multimodal-multitask framework with cross-modal relation and hierarchical interactive attention for semantic comprehension.” *Information Fusion* (2025): 103628, DOI: <https://doi.org/10.1016/j.inffus.2025.103628>, [Impact Factor: 15.5, H-index: 201].
2. **Mohammad Zia Ur Rehman**, Sufyaan Zahoor, Areeb Manzoor, Musharaf Maqbool, and Nagendra Kumar. “A context-aware attention and graph neural network-based multimodal framework for misogyny detection.” *Information Processing & Management* 62, no. 1 (2025): 103895, DOI: <https://doi.org/10.1016/j.ipm.2024.103895>, [Impact Factor: 6.9, H-index: 149].
3. **Mohammad Zia Ur Rehman**, Devraj Raghuvanshi, Harshit Pachar, Chandravardhan Singh Raghaw, and Nagendra Kumar. “Hierarchical Attention-enhanced Contextual CapsuleNet for Multilingual Hate Speech Detection.” *Expert Systems with Applications* 268 (2025): 126285, DOI: <https://doi.org/10.1016/j.eswa.2024.126285>, [Impact Factor: 7.5, H-index: 315].
4. **Mohammad Zia Ur Rehman**, Somya Mehta, Kuldeep Singh, Kunal Kaushik, and Nagendra Kumar. “User-aware multilingual abusive content detection in social media.” *Information Processing & Management* 60, no. 5 (2023): 103450, DOI: <https://doi.org/10.1016/j.ipm.2023.103450>, [Impact Factor: 6.9, H-index: 149].
5. **Mohammad Zia Ur Rehman**, Aditya Shah, and Nagendra Kumar. “Detecting implicit sexism in digital social networks via contrastive learning-based adaptive network.” *Computers and Electrical Engineering* 137 (2026): 111261, DOI: <https://doi.org/10.1016/j.compeleceng.2026.111261>, [Impact Factor: 4.9, H-index: 119].

In Refereed Conferences

1. **Mohammad Zia Ur Rehman**, Sai Kartheek Reddy Kasu, Shashivardhan Reddy Koppula, C Sai Rithwik Reddy, Shwetank Shekhar Singh and Nagendra Kumar. “X-MuTeST: A Multilingual Benchmark for Explainable Hate Speech Detection and A Novel LLM-consulted Explanation Framework.” *In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2026)*, 40(46), 39134–39142. DOI: <https://doi.org/10.1609/aaai.v40i46.41261>, (Core A*).

-
2. **Mohammad Zia Ur Rehman**, Anukriti Bhatnagar, Omkar Kabde, Shubhi Bansal, and Nagendra Kumar. “ImpliHateVid: A Benchmark Dataset and Two-stage Contrastive Learning Framework for Implicit Hate Speech Detection in Videos.” In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025)*, Volume 1: Long Papers, pp. 17209-17221, DOI: <https://doi.org/10.18653/v1/2025.acl-long.842>, (Core A*).
 3. **Mohammad Zia Ur Rehman**[†], Sai Kartheek Reddy Kasu[†], Shahid Shafi Dar, Rishi Bharat Junghare, Dhanvin Sanjay Namboodiri, and Nagendra Kumar. “D-HUMOR: Dark Humor Understanding via Multimodal Open-ended Reasoning.” In *Proceedings of IEEE International Conference on Data Mining (ICDM 2025)*, pp. 377-386, DOI: <https://doi.org/10.1109/ICDM65498.2025.00045>, (Core A*).
- ([†] Co-first authors)

The publication other than this doctoral dissertation is mentioned below:

1. **Mohammad Zia Ur Rehman**, Devraj Raghuvanshi, and Nagendra Kumar. "Kisan-QRS: A deep learning-based automated query-response system for agricultural decision-making." *Computers and Electronics in Agriculture* 213 (2023): 108180, DOI: <https://doi.org/10.1016/j.compag.2023.108180>, [Impact Factor: 8.9, H-index: 209].
2. Ashutosh Anshul, Gumpili Sai Pranav, **Mohammad Zia Ur Rehman**, and Nagendra Kumar. "A multimodal framework for depression detection during COVID-19 via harvesting social media." *IEEE Transactions on Computational Social Systems* 11, no. 2 (2023): 2872-2888, DOI: <https://doi.org/10.1109/TCSS.2023.3309229>, [Impact Factor: 4.9, H-index: 76].

Contents

Abstract	i
List of Publications	iv
Table of Contents	viii
List of Figures	xv
List of Tables	xvii
List of Algorithms	xviii
List of Abbreviations and Symbols	xix
1 Introduction	1
1.1 Multilingual Hostile Content Detection and Explanation	2
1.2 Multimodal Hostile Content Detection in Memes	3
1.3 Multimodal Implicit Hate Speech Detection in Videos	4
1.4 Scope and Objectives of the Thesis	5
1.4.1 Research Questions	5
1.5 Key Contributions	6
1.5.1 Textual Modality: Context and Explainability	6
1.5.2 Memetic Modality: Noise Reduction and Reasoning	7
1.5.3 Video Modality: Implicit Hate Detection	7
2 Literature Review	10
2.1 Text-based Hostility Detection and Explainable Methods	10

2.1.1	Existing Works on Multilingual Low-resource and Indic Languages	10
2.1.2	Explainability in Hate Speech Detection	12
2.2	Multimodal Methods to Detect Various Types of Hostility in Memes	13
2.2.1	Misogyny Detection	13
2.2.2	Attention-based Multimodal and Multitask Learning	15
2.2.3	Dark Humor and Related Provocative Content	16
2.3	Video-based Implicit Hostile Content Detection	17
3	Multilingual Hostile Content Detection	20
3.1	Introduction	20
3.2	Methodology	23
3.2.1	Data Preprocessing	24
3.2.2	Social Context Features	27
3.2.3	Text Context Features	32
3.2.4	Abuse Classification	35
3.2.5	Ensemble Learning	40
3.3	Experimental Evaluations	43
3.3.1	Experimental Setup	43
3.3.2	Experimental Results	47
3.4	Discussion	54
3.4.1	Limitations	55
3.5	Conclusion	55
4	Multilingual Hostile Content Explanation	57
4.1	Introduction	57
4.1.1	Motivation: Scarcity of Hateful Rationales	58
4.1.2	A Glance at the Proposed Solution	58
4.2	Dataset Description and Annotation	59
4.2.1	Dataset Source	59
4.2.2	Rationale Annotation Procedure	60
4.3	Methodology	62
4.3.1	Training Procedure	63
4.3.2	Explainability via N-gram Contributions	64

4.3.3	Final Explanations through LLM Consultation	65
4.4	Experimental Setup and Results	65
4.4.1	Metrics for Evaluation	65
4.4.2	Hyperparameter and System Details	66
4.4.3	Comparison Methods	67
4.4.4	Results Analysis for Telugu	67
4.4.5	Results Analysis for Hindi	68
4.4.6	Results Analysis for English	69
4.4.7	Parameter Sensitivity Analysis & Ablation	69
4.4.8	Generalizability	71
4.5	Conclusion	71
5	Multimodal Hostile Content Detection	73
5.1	Introduction	73
5.1.1	Women on Social Media: A Statistical View	73
5.1.2	Existing Approaches and Challenges	74
5.1.3	Research Objectives	75
5.1.4	A Glimpse into the Proposed Solution Framework	76
5.1.5	Key Contributions	76
5.2	Methodology	77
5.2.1	Data Preprocessing	78
5.2.2	Feature Extraction	79
5.2.3	Multimodal Attention Module (MANM)	81
5.2.4	Graph-based Feature Reconstruction Module (GFRM)	87
5.2.5	Content-specific Feature Learning Module (CFLM)	89
5.2.6	Classification Module	90
5.3	Experimental Evaluations	92
5.3.1	Experimental Setup	92
5.3.2	Results	95
5.3.3	Qualitative Analysis	102
5.4	Conclusion	103

6	Multimodal-Multitask Method for Hostile Content Detection	106
6.1	Introduction	106
6.1.1	Existing Approaches and Challenges	107
6.1.2	Navigating Our Approach: Proposed Solution at a Glance	108
6.1.3	Key Contributions	109
6.2	Problem Definition and Objectives	110
6.2.1	Main Problem	110
6.2.2	Objective 1: Cross-modal Feature Interaction	111
6.2.3	Objective 2: Hierarchical Attention-based Feature Interaction	111
6.3	Methodology	112
6.3.1	Data Preprocessing	112
6.3.2	Data Augmentation	114
6.3.3	Feature Extraction	115
6.3.4	Hierarchical Interactive Monomodal Attention	117
6.3.5	Cross-modal Relation Learning	121
6.3.6	Multifeature Learning and Classification	128
6.4	Experimental Evaluations	129
6.4.1	Datasets	129
6.4.2	Baselines for Comparison	131
6.4.3	Results	131
6.4.4	Ablation Analysis	136
6.4.5	Qualitative Analysis	143
6.5	Limitations	145
6.6	Conclusion	146
7	Multimodal Hostile Content Detection via LLM Integration	148
7.1	Introduction	148
7.2	Dataset Description and Annotation Process	151
7.2.1	Dataset Collection	151
7.2.2	Annotator Training	151
7.2.3	Annotation in Batch Mode	153
7.2.4	Inter-Annotator Agreement	153

7.2.5	Dataset Statistics	154
7.3	Methodology	156
7.3.1	Structured Reasoning Generation	156
7.3.2	Feature Extraction	159
7.3.3	Tri-stream Cross-Reasoning Network (TCRNet)	159
7.3.4	Feature Aggregation	160
7.3.5	Hyperparameters and System Description	160
7.4	Results and Discussion	161
7.4.1	Experimental Findings	162
7.5	Ablation Study	163
7.6	Conclusion	164
8	Hostile Content Detection in Videos	167
8.1	Introduction	167
8.2	Data Collection and Annotation	169
8.2.1	Platforms for Video Collection	169
8.2.2	Dataset Statistics	169
8.2.3	Annotation Guidelines	170
8.2.4	Annotation Process	170
8.3	Methodology	171
8.3.1	Preprocessing	171
8.3.2	Feature Extraction	171
8.3.3	Two-stage Contrastive Learning	172
8.3.4	Multimodal Classification	174
8.4	Results	175
8.4.1	Datasets	175
8.4.2	Experimental Setup	175
8.4.3	Compared Methods	175
8.4.4	Effectiveness Comparison	176
8.4.5	Ablation Analysis	178
8.5	Conclusion	180
8.6	Limitations	181

9	Conclusions and Future Research Directions	183
9.1	Summary of the Thesis	183
9.1.1	Multilingual Hostile Content Detection and Explanation	183
9.1.2	Multimodal Hostile Content Detection in Memes	184
9.1.3	Multimodal Hostile Content Detection in Videos	184
9.2	Contributions	184
9.3	Limitations	185
9.4	Future Research Directions	186
	Bibliography	189

List of Figures

3.1	System Architecture of Multilingual Abusive Content Detection Framework	24
3.2	Representation of Data Preprocessing Steps	27
3.3	Architecture of Abuse Classification Module	36
3.4	Ablation Analysis Results on SCIDN Dataset	51
3.5	Ablation Analysis Results on MACI Dataset	52
4.1	Example of Human and LLM Rationales*	58
4.2	Iterative Annotation Procedure	61
4.3	Architecture of LLM-consulted Explanation Framework	62
4.4	Influence of α Parameter on Classification Performance in Stage-2 Training	70
4.5	Ablation on Two Stages of Training of X-MuTeST	71
5.1	Example of Misogynistic and Non-misogynistic Content	75
5.2	System Architecture of Multimodal Misogyny Detection Method	77
5.3	Example of Cleaned Images	78
5.4	Adaptive Gating-based Cross Attention for MCAN	83
5.5	Graph Construction and Node Update	87
5.6	Classification Module	91
5.7	Test-Time Augmentation	91
5.8	Influence of Different Modules	98
5.9	Influence of Different Attention Mechanisms	99
5.10	Influence of Different Modalities	100
5.11	Influence of Additional Features	101
5.12	Qualitative Analysis of Misogyny Detection Method	103

6.1	Multimodal Machine Cognition and Noise Propagation	109
6.2	Overall Architecture of MM-ORIENT	113
6.3	In-modal Relation Graph construction and Node Update	125
6.4	Cross-modal Graph Construction and Node Update	126
6.5	Influence of Different Modalities on Binary classification in MM-ORIENT	137
6.6	Influence of Different Attention Mechanisms on Binary Classification in MM-ORIENT	139
6.7	Influence of Different Attention-based Fusions on Binary Classification in MM-ORIENT	140
6.8	Influence of Different Graph Components on Binary Classification in MM- ORIENT	141
6.9	Influence of Features on Binary Classification in MM-ORIENT	142
6.10	Qualitative Analysis of MM-ORIENT	143
7.1	Comparison of Clean and Dark Humor	149
7.2	Example of Dark Humor, Target, and Intensity	152
7.3	Architecture of Reasoning-augmented Multimodal Dark Humor Detection Method	156
7.4	Self-loop Mechanism using Role Reversal (RR) Prompting	158
7.5	Ablation Study of TCRNet on Dark Humor Meme Analysis	163
8.1	Implicit and Explicit Hate Videos Example	168
8.2	ImpliHatevid Method Architecture	172
8.3	Impact of Different Modality Combinations on Binary Classification on Im- pliHateVid Dataset	179
8.4	Impact of Different Features on Binary Classification on ImpliHateVid Dataset	180

List of Tables

3.1	Examples of comments	22
3.2	Point Biserial correlation of features with class labels in SCIDN dataset . .	32
3.3	Point Biserial correlation of features with class labels in MACI dataset . . .	32
3.4	SCIDN dataset description	44
3.5	Language-wise description of SCIDN dataset	44
3.6	MACI dataset description	45
3.7	Language-wise description of MACI dataset	45
3.8	Comparison with classical and state-of-the-art baseline methods on SCIDN dataset	48
3.9	Comparison with classical and state-of-the-art baseline methods on MACI dataset	50
3.10	Language-wise comparison on SCIDN dataset	53
3.11	Language-wise comparison on MACI dataset	54
4.1	Summary of hate and non-hate records	60
4.2	Kappa agreement scores	62
4.3	Hyperparameters for 2nd-stage training in X-MuTeST	67
4.4	Classification and explainability performance of various methods on Telugu	68
4.5	Classification and explainability performance of various methods on Hindi .	69
4.6	Classification and explainability performance of various methods on English	70
5.1	Performance comparison over MAMI dataset	95
5.2	Performance comparison over MMHS150K dataset	97
5.3	Ablation analysis on different modules	98

5.4	Ablation analysis on attention mechanisms	99
5.5	Ablation analysis on modalities	100
5.6	Ablation analysis on additional features	101
6.1	Memotion Dataset Statistics	130
6.2	Performance Comparison on Memotion Dataset	132
6.3	Performance Comparison on MMHS150K Dataset	134
6.4	Performance Comparison on HarMeme Dataset	135
6.5	Ablation Analysis on Different Modalities in MM-ORIENT	137
6.6	Ablation Analysis on Attention Components in MM-ORIENT	138
6.7	Effect of Different Attention-based Fusion Mechanisms in MM-ORIENT .	140
6.8	Ablation Analysis on Graph Components in MM-ORIENT	141
6.9	Ablation Analysis on Additional Features in MM-ORIENT	142
7.1	Inter-annotator Agreement Scores of D-Humor Dataset. A: Annotator, DH: Dark Humor, Unwt: Unweighted, Wt: Weighted, C Kappa: Cohen’s Kappa, F Kappa: Fleiss’ Kappa	154
7.2	Summary of D-Humor Dataset	155
7.3	Key Hyperparameter Configurations in D-Humor	161
7.4	Performance Analysis of Different Methods on D-Humor Dataset	161
8.1	ImpliHateVid Dataset Statistics	170
8.2	Effectiveness Comparison for Binary Classification	176
8.3	Effectiveness Comparison for Multiclass Classification	177
8.4	Impact of Different Modalities on Multiclass Classification on ImpliHateVid Dataset	178
8.5	Impact of Different Features on Multiclass Classification on ImpliHateVid Dataset	179

List of Algorithms

3.1	User-Post Affinity	30
3.2	Data Augmentation	34
3.3	Majority Voting with Confidence Decision	41
5.1	Graph Construction	88
5.2	Additive Random Perturbation	92
6.1	Hierarchical Interactive Monomodal Attention	118
6.2	Cross-modal Relation Learning	122

List of Abbreviations and Acronyms

List of Abbreviations

AGM	Adaptive Gating Module
BiLSTM	Bidirectional Long Short-Term Memory
BLIP	Bootstrapping Language-Image Pre-training
BoW	Bag of Words
CFLM	Content-specific Features Learning Module
CLIP	Contrastive Language-Image Pre-training
CMRL	Cross-Modal Relation Learning
D-HUMOR	Dark Humor Understanding via Multimodal Open-ended Reasoning
GFRM	Graph-based Feature Reconstruction Module
GNN	Graph Neural Network
HateXplain	Hate Speech Explanation Benchmark
HIMA	Hierarchical Interactive Monomodal Attention
ImpHateVid	Implicit Hate Speech Detection in Videos
IOU	Intersection Over Union
LIME	Local Interpretable Model-agnostic Explanations

LLM	Large Language Model
LR	Logistic Regression
M-BERT	Multilingual BERT
MACI	Multilingual Abusive Comment Identification
MAMI	Multimedia Automatic Misogyny Identification
MANM	Multimodal Attention Module
MM-ORIENT	Multimodal-Multitask framework with Cross-modal Relation and Hierarchical Interactive Attention
MSAN	Multihead Self-Attention
MSL	Misogyny-Specific Lexicon
MultiOFF	Multimodal Offensive Language Identification
MuRIL	Multilingual Representations for Indian Languages
RQ	Research Question
RRT	Relative Reporting Tendency
SCIDN	ShareChat IndoML Datathon NSFW
SCL	Supervised Contrastive Learning
SVM	Support Vector Machine
TCRNet	Tri-stream Cross-Reasoning Network
TF-IDF	Term Frequency-Inverse Document Frequency
ViT	Vision Transformer
VLM	Vision-Language Model
X-MuTeST	Explainable Multilingual Hate Speech Detection

List of Abbreviations and Acronyms

XLM-R XLM-RoBERTa

Chapter 1

Introduction

The unprecedented growth of social media platforms has fundamentally transformed the way individuals communicate, express opinions, and participate in public discourse. Platforms such as Twitter/X, Facebook, Instagram, Reddit, YouTube, and emerging video platforms have enabled rapid dissemination of user-generated content at an unprecedented scale. While these platforms foster connectivity and democratize expression, they have also become fertile grounds for the spread of hostile content, including abusive language, hate speech [1], misogyny [2], and dark humor [3]. Such content not only undermines the quality of online discourse but also causes tangible psychological, social, and societal harm, ranging from emotional distress to the normalization of discrimination and violence.

Hostile content is a broad umbrella term that encompasses multiple overlapping yet distinct phenomena. Abusive content often involves insults, profanity, or harassment directed at individuals. Hate speech targets individuals or groups based on protected characteristics such as race, religion, gender, caste, or ethnicity [1], often with the intent to demean or incite hostility. Misogyny constitute gender-targeted hostility, disproportionately affecting women [4]. Dark humor represents a more subtle and ambiguous category, where humor is constructed around sensitive, taboo, or traumatic themes, often blurring the boundary between humor and harm. The subjective and context-dependent nature of these categories makes automated detection a highly challenging task. From a societal perspective, the unchecked proliferation of hostile content has serious implications. Studies have shown that exposure to online abuse and hate speech can lead to anxiety, depression, and social withdrawal, particularly among vulnerable populations [5, 6]. At a broader level, such content contributes to polarization,

radicalization, and erosion of trust in digital spaces. Consequently, governments, policy-makers, and social media companies increasingly rely on automated content moderation systems to identify and mitigate harmful content at scale. However, the complexity, diversity, and evolving nature of hostile expressions continue to outpace existing computational approaches.

1.1 Multilingual Hostile Content Detection and Explanation

A major limitation of existing systems is their strong bias toward high-resource languages, particularly English. In multilingual societies such as India, users frequently communicate in low-resource Indic languages and in code-mixed or code-switched forms [7], often written in Roman script. Such text exhibits spelling variations, informal grammar, and non-standard vocabulary, which significantly degrades the performance of conventional text-based models [8]. Addressing abusive and hateful content in these settings requires models that are robust to linguistic noise and capable of leveraging limited annotated data. This challenge motivates the need for multilingual and cross-lingual approaches tailored to low-resource languages.

Most state-of-the-art hostile content detection systems rely on deep neural networks, which function as black boxes. While these models may achieve high accuracy [9], their decisions are often opaque, making it difficult for users, moderators, and policymakers to trust or audit them. Explainability is especially critical in sensitive applications such as hate speech detection [10], where incorrect predictions can have serious ethical and legal consequences. Aligning model explanations with human reasoning is therefore an essential requirement for practical deployment.

The first part of the thesis focuses on textual hostile content detection, with an emphasis on low-resource Indic languages. The first contribution introduces a user-aware multilingual framework for abusive content detection that integrates textual features with social context signals such as user history and post affinity. By jointly modeling linguistic content and social behavior, the proposed approach significantly improves detection performance in multilingual and code-mixed environments, addressing key limitations of text-only models.

While improved accuracy is essential, it is insufficient in isolation. The second textual contribution addresses the explainability gap through X-MuTeST, an explainable multilingual hate speech detection framework. This work introduces benchmark datasets with token-level human rationales for Hindi, Telugu, and English, filling a critical resource gap for under-resourced languages. X-MuTeST combines large language model (LLM) explanations with n-gram-based explainability and attention-guided training, resulting in models that are both accurate and interpretable. Extensive evaluation using plausibility and faithfulness metrics demonstrates improved alignment between machine-generated explanations and human reasoning.

1.2 Multimodal Hostile Content Detection in Memes

The second part of the thesis transitions to static multimodal content, with a focus on memes. Memes are a dominant medium for expressing sarcasm [11], humor [12], and hostility online [13], yet their interpretation requires joint reasoning over text and images.

Modern social media content is inherently multimodal. Memes combine images and text to convey meaning that, sometimes, cannot be inferred from either modality alone [14, 15]. Simple early-fusion or late-fusion strategies often fail to capture the nuanced cross-modal interactions necessary for understanding such content [16]. Moreover, noise in one modality can propagate to others during fusion, leading to incorrect predictions [17]. Designing noise-resilient multimodal architectures that preserve discriminative unimodal information while enabling effective cross-modal reasoning remains a key challenge. To address noise propagation and modality dominance in multimodal fusion, the thesis introduces MM-ORIENT, a multimodal-multitask framework based on cross-modal relation graphs and Hierarchical Interactive Monomodal Attention (HIMA). Unlike traditional fusion methods that rely on explicit early interactions, MM-ORIENT reconstructs unimodal features in a cross-modal manner, reducing noise and preserving discriminative information. The framework demonstrates strong performance across multiple tasks, including offensive content detection, sarcasm, and humor recognition.

Building on general offensive content detection, the thesis further explores gender-targeted hostility through a specialized multimodal misogyny detection framework. This

work introduces context-aware attention, graph-based feature reconstruction, and content-specific features such as toxicity scores and image captions. Evaluated on benchmark datasets, the proposed approach achieves substantial improvements over existing methods, highlighting the importance of task-specific modeling for nuanced forms of hostility.

Dark humor represents a particularly challenging form of hostile content due to its reliance on implicit meaning, cultural context, and sensitive themes [3]. To address the lack of resources in this area, the thesis introduces D-Humor, a novel multimodal dataset annotated for dark humor presence, target categories, and intensity levels. Beyond dataset creation, a reasoning-augmented framework is proposed that leverages vision–language models to generate and iteratively refine structured explanations through a role-reversal self-loop. A tri-stream cross-reasoning network then fuses textual, visual, and reasoning representations, enabling fine-grained dark humor understanding beyond binary classification.

1.3 Multimodal Implicit Hate Speech Detection in Videos

Hostility on social media is not always explicit [18]. Users increasingly employ implicit strategies, such as sarcasm, euphemisms, coded language, and cultural references, to evade moderation. Implicit hate speech is particularly difficult to detect because it often appears innocuous when analyzed in isolation [19]. Correct interpretation requires contextual understanding, background knowledge, and sometimes multimodal cues. This issue is further amplified in video content, where hateful intent may emerge only from the interaction of speech, visuals, and audio cues over time.

The final part of the thesis extends hostile content detection to the video modality, which remains underexplored in prior work. Videos pose unique challenges due to their temporal nature and the need to integrate audio, visual, and textual information. This thesis introduces ImpliHateVid, one of the first large-scale benchmark datasets specifically curated for implicit hate speech detection in videos. To effectively model subtle and context-dependent hate expressions, a two-stage contrastive learning framework is proposed. The approach aligns modality-specific representations in a shared embedding space and refines them using cross-encoders, significantly improving performance on both implicit and explicit hate detection tasks.

1.4 Scope and Objectives of the Thesis

This thesis aims to address the aforementioned challenges through a comprehensive investigation of hostile content detection across three major modalities: text, memes (image-text), and videos. The thesis is structured around six peer-reviewed research contributions, each targeting a specific gap in the literature while collectively advancing the state of the art in hostile content analysis.

1.4.1 Research Questions

Guided by the challenges identified in hostile content detection and the overarching goal of developing robust, explainable, and multimodal moderation systems, this thesis is structured around the following core research questions (RQs). These questions collectively shape the methodological choices, dataset curation, and evaluation strategies adopted across the six constituent studies.

RQ1: How can abusive and hateful content be effectively detected in low-resource and multilingual textual settings, particularly in the presence of code-mixing, spelling variation, and informal language?

This question investigates the limitations of text-only and monolingual approaches when applied to Indic and other low-resource languages. It motivates the exploration of cross-lingual representations and the integration of social context features such as user history and post affinity to compensate for data scarcity and linguistic noise.

RQ2: How can automated hate speech detection systems be made more interpretable and aligned with human reasoning, especially for under-resourced languages?

This question addresses the growing need for explainability in sensitive moderation tasks. It examines whether incorporating human-annotated rationales, n-gram-based explainability, and large language model (LLM) consultation can jointly improve both classification performance and the faithfulness of model explanations.

RQ3: How can multimodal fusion architectures be designed and further expanded to reduce noise propagation while preserving discriminative information within individual modalities?

RQ4: How can temporal dynamics and audio-visual cues be effectively leveraged to detect hostility in dynamic video content?

Moving beyond static text and images, this question addresses the complexity of video hate speech detection. It investigates methods for modeling temporal dependencies and synchronizing audio tracks with visual frames to capture evolving hostile narratives that are not discernible from a single snapshot or transcript alone.

RQ5: What methodologies are required to curate high-quality, diverse, and annotated benchmarks for low-resource and multimodal hostile content detection?

Recognizing that data scarcity is a primary bottleneck for under-resourced languages, this question focuses on the creation of robust evaluation frameworks. It outlines strategies for ethical data scraping, ensuring cultural diversity, and managing annotation quality to produce standardized datasets that facilitate future research in the field.

1.5 Key Contributions

The primary contributions of this thesis are categorized by the modalities and methodological advancements they represent:

1.5.1 Textual Modality: Context and Explainability

- **User-Aware Multilingual Framework:** We proposed a novel framework for abusive content detection in low-resource Indic languages (including Hindi, Telugu, and Bengali) that integrates social context features (user history, post affinity) with cross-lingual textual embeddings [8].
- **Explainable Hate Speech Benchmark (X-MuTeST):** We developed a benchmark dataset containing token-level human rationales for hate speech in Hindi, Telugu, and English. We introduced a novel two-stage training framework that combines LLM consultation with N-gram-based explainability to enhance both model plausibility and faithfulness [20].

1.5.2 Memetic Modality: Noise Reduction and Reasoning

- **Context-Aware Misogyny Detection:** We developed a unified framework for detecting misogyny and sexism that employs adaptive gating-based context-aware attention. This framework uniquely integrates content-specific features, such as toxicity scores and image captions, with graph-based feature reconstruction [21].
- **Noise-Resilient Multimodal Framework (MM-ORIENT):** We designed a Multimodal-Multitask framework that utilizes cross-modal relation graphs and Hierarchical Interactive Monomodal Attention (HIMA). This approach reconstructs monomodal features to mitigate the impact of noise during fusion, significantly improving performance on downstream tasks like sarcasm and humor detection [16].
- **Dark Humor Understanding (D-Humor):** We introduced D-Humor, a novel dataset annotated for dark humor intensity and target categories. We proposed a reasoning-augmented framework using Vision-Language Models (VLMs) with a “Role-Reversal Self-Loop” mechanism to generate and refine structured explanations for implicit humor [3].

1.5.3 Video Modality: Implicit Hate Detection

- **ImpliHateVid Dataset:** We curated ImpliHateVid, one of the first large-scale datasets specifically dedicated to implicit hate speech in videos, comprising over 2,000 samples annotated for implicit vs. explicit hate [22].
- **Two-Stage Contrastive Learning Framework:** We proposed a novel video detection architecture that first trains modality-specific encoders (audio, text, image) and then aligns them via cross-modal encoders using supervised contrastive loss. This method effectively captures the subtle, multimodal signals required to detect implicit hate speech.

The remainder of this thesis is organized as follows. Chapter 2 reviews related work across text, image, and video-based hostile content detection. Chapters 3 and 4 detail the

1.5 Key Contributions

individual research contributions corresponding to the multilingual hostile content detection and Explanation, respectively. Next, Chapters 5, 6, and 7 detail the multimodal hostile content detection methods in memes. Subsequently, Chapter 8 detail the hostile content in detection in videos. Finally, Chapter 9 concludes the thesis with a summary of findings and directions for future research.

Chapter 2

Literature Review

In this chapter, we present existing works related to hostile content detection, such as abusive, hateful and dark humor. First, we provide an overview of existing works on Indic and multilingual methods related to abusive/hateful content detection and explainability. Next, we discuss a few works on misogyny detection in memes and multimodal-multitask learning methods. Subsequently, we discuss existing works on dark humor and hate speech detection in videos.

2.1 Text-based Hostility Detection and Explainable Methods

2.1.1 Existing Works on Multilingual Low-resource and Indic Languages

In existing works, researchers have used classical as well as transformer-based methods for text feature extraction. Classical methods of text feature extraction include rule-based approach, count-based and frequency-based methods such as Bag of words (BoW) & TF-IDF, and non-contextual word-embeddings [1]. The study conducted by Akram et al. [23] centers around the detection of hatred in the Urdu language. They utilize N-gram as one of the feature extractors and compare its performance with other recent feature extractors. Their findings reveal that the N-gram technique falls behind in terms of performance. In Marreddy

et al. [24], authors develop Telugu BoW and Telugu TF-IDF along with word-embedding techniques. They apply their method to various downstream tasks, including hate speech detection. Their observation indicates that classical techniques show lower performance as compared to other techniques. FHA et al. [25] focuses on detecting and analyzing hate speech in Tamil-English code-mixed content. They use the TF-IDF text feature extraction technique with different Machine Learning methods for classification. Naive-Bayes (NB) achieves the highest F1-score, and Support Vector Machine (SVM) achieves the highest accuracy. In the aforementioned classical feature extractors, a major problem is that they are non-contextual which hampers the performance of the hate detection systems.

To overcome the drawbacks of classical methods, techniques have been developed to generate contextual embeddings [26]. The major upgrade in NLP techniques is attributed to Google which introduced transformers with attention mechanism. Transformer-based methods have proved to be extremely useful in hate speech detection for low-resource languages [27]. Ghosal et al. [28] employs an extended fuzzy SVM classifier in conjunction with transformer-based M-Bert embeddings to classify hate speech in the Bengali language. Their research showcases that M-Bert embeddings exhibit superior performance compared to classical word2vec embeddings. This improvement can be attributed to the transformer-based method's enhanced ability to capture contextual information effectively. Madhu et al. [9] and Sharma et al. [1] demonstrate that transformers-based methods outperform classical methods of feature extraction for hate speech detection in Hindi-English code-mixed text. Velanker et al. [29] demonstrates that transformer-based monolingual techniques, specifically MahaBERT, outperform their multilingual counterparts like M-Bert. They apply MahaBERT-based embeddings for Marathi hate speech detection. One advantage of the transformer-based model is the contextual embeddings; another major advantage is transfer-learning, i.e., sharing and re-utilizing the model weights developed during training [30]. If training is done on a corpus for a language with a substantial resource base, its knowledge can help the process of training for low-resource languages that are still being researched.

Researchers have investigated detecting abusive and hateful content by considering social context features, such as user index, followers, the number of posts, and other post metadata. In Krupalija et al. [31], incorporating user features have shown the enhanced performance of the model. They utilize user followers, user status count, and user hate speech index. Chopra

et al. [32] employs graph embeddings for author profiling for hate detection in Hindi-English text. Their methods exhibit superior performance compared to state-of-the-art approaches. Pitlis et al. [33] uses the user’s tendency to detect racism and sexism. Authors observe improvement in the performance of the model when user features are used with the ensemble method [34]. Work of Sánchez-Rada et al. [35] uses social context features for sentiment analysis. Mosca et al. [36] and Unsvag et al. [37] conclude that the utilization of user features can lead to a better understanding of the model and aid in the detection of unintended bias.

2.1.2 Explainability in Hate Speech Detection

Explainability in hate speech detection has garnered significant attention, with recent advancements highlighting both its potential and underlying challenges. The works, such as Zaidan et al. [38] and Yessenalina et al. [39] utilized rationales for improving sentiment classification, which set the stage for explainable approaches in more domains [40], including hate speech [10, 41]. The work of Mathew et al. [10] provides one of the first datasets to include human-annotated rationales for hate speech. For each sample, annotators not only label whether the text contains hate speech but also highlight the specific tokens that are relevant to that judgment. The authors highlight that even though a model may be accurate in classification, its reasoning may not align with human logic or provide meaningful explanations. Recent works in LLMs and attention-based models also highlight critical challenges in explainability, as seen in works by Roy et al. [42], Clarke et al. [43], and Arshad and Shahzad [44]. These studies underscore difficulties in ensuring faithful rationales, even when powerful models achieve high classification accuracy.

Kapil et al. [45] and Sawant et al. [46] emphasize the importance of precise annotations and transparency in low-resource languages such as Hindi, where cultural aspects affect how hate speech is perceived and classified. Similarly, Yadav et al. [47] address the complexities of code-mixed languages, highlighting how explainability helps clarify model predictions in multilingual settings. In this context, Geleta et al. [48] introduce a multilingual framework for assessing hate speech intensity levels using explainable AI, further underlining the need for transparent hate speech detection models. By leveraging explainable methodologies, these studies collectively enhance the detection of hate speech across linguistic boundaries, ensuring that cultural and linguistic differences are accounted for in model interpretation.

2.2 Multimodal Methods to Detect Various Types of Hostility in Memes

2.2.1 Misogyny Detection

Zhang & Wang [4] explore the relationship between vision and language by studying the effectiveness of multimodal pretrained models. Findings from the study indicate that multimodal fine-tuning generally performs better than unimodal approaches for both UNITER and CLIP, with CLIP excelling due to extensive pretraining. XGBoost classifier competes with fine-tuning, particularly by leveraging CLIP’s image features. Muti et al. [49] address a task focused on detecting misogyny in memes using both text and image data. The proposed MMBT model, which integrates text and image embeddings, employed BERT for text and CLIP for image encoding. This model demonstrates superior performance compared to models utilizing only one modality. Gu et al. [50] introduce an approach to detect and classify misogyny in multimedia content using both text and images. The proposed multimodal-multitask variable autoencoder (MMVAE) model combines an image-text embedding module, variable autoencoder module, and multitask learning module. Pretrained models like BERT, ResNet-50, and CLIP are used for embedding, and the VAE combines them into a latent representation. The MMVAE outperforms single-modal baselines such as BERT and ResNet-50. Gomez et al. [51] present an innovative contribution to hate speech detection by introducing the MMHS150K dataset, containing 1,50,000 Twitter image-text pairs annotated as hate speech or not. One of the labels in the dataset is sexism. The study introduces the Feature Concatenation Model (FCM), Spatial Concatenation Model (SCM), and Textual Kernels Model (TKM). A few other existing works focus on different approaches for multimodal hate speech detection, such as transformer-based [52], ensemble learning [53], and pretrained models [54]. Other than misogyny detection, few multimodal approaches have been proposed for other tasks such as depression detections [55] and hashtag recommendation [56].

Most of the aforementioned approaches have shown improvement over unimodal approaches except the approach presented by Gomez et al. [51]. However, these approaches utilize text-embeddings and visual features only while neglecting important features such

as toxicity features and image captions. These features may provide useful insights into the content. Additionally, the generalization of predictions of test data is also one of the limitations that can be overcome by TTA.

Guo et al. [57] employ a transformer method to detect sexism in textual content. BERT+BiLSTM is used for local and global features, and focal loss is used to improve the performance further. Data augmentation is additionally employed to augment the sample count. Due to their ability to provide contextual embeddings, transformer-based techniques have been extensively investigated for misogyny detection in the existing works. Finetuning of transformer-based models for downstream tasks has shown performance improvement [58–60]. Muti et al. [61] provide an approach for misogyny detection in text-based content for English, Spanish, and Italian languages. They train ten transformer-based models for different combinations of languages. Dehingia et al. [62] compare the BERT model with Machine Learning (ML) methods such as Naive Bayes, Logistic Regression (LR), and Support Vector Machine (SVM). They utilize the Term Frequency - Inverse Document Frequency (TF-IDF) approach for ML methods. Their observation indicates that BERT performs better than ML methods. Rehman et al. [8] employs a transformer-based approach along with a confidence decision for predictions from multiple transformers. A recent paradigm shift in hate in speech detection methods is the use of Large Language Models (LLMs) such as GPT3.5, GPT-4, LLaMA, and Mistral. These methods are being employed for analyzing and identifying hate speech within text data. Han et al. [63] utilize GPT-3 LLM for hate speech detection using prompts without updating the parameters of GPT. LLM accomplishes this by receiving a test instance, a description of the new task, and a small set of training examples (such as input-label pairs) as inputs. Luo et al. [64] also utilize LLMs for hate speech detection in the legal domain. They employ a prompt-based classification in zero-shot and few-shot settings, which include LLMs such as GPT-3.5, GPT-4, and LLaMA-2, to name a few. The work of Christodoulou [65] explores parameter finetuning and prompt-tuning of Mistral LLM for hate speech detection. The results show that the prompt-tuned method outperforms other methods in most of the tasks performed. Unimodal approaches, particularly those relying solely on text, have a limited ability to understand the broader context surrounding a statement. As online content becomes increasingly multimedia-rich, unimodal approaches struggle to analyze images, videos, or a combination of different modalities ef-

fectively. Offensive content can be shared through visual elements or a combination of text and images, requiring a broader approach.

2.2.2 Attention-based Multimodal and Multitask Learning

Hossain et al. [66] propose a context-aware alignment framework that aligns visual and textual features before fusion, rather than relying on standard cross-attention. Their MCA-SCF model computes modality-guided context vectors using attention, which are then fused along with residual modality features. This strategy improves hateful meme detection performance on both English (MultiOFF) and Bengali (MUTE) datasets and demonstrates robustness in low-resource settings. Yadav and Singh [67] present HateFusion, an explainability-enhanced system to detect implicit hate speech using fine-tuned BERT and DistilBERT. Their model employs attention mechanisms for classification and integrates interpretability tools such as LIME and SHAP to highlight influential tokens. HateFusion shows strong performance across binary, 3-way, and 7-way tasks, and includes a GUI for real-time feedback and flagging, enabling dynamic model refinement.

Yao et al. [68] propose a multimodal, multiinteractive, and multihierarchical neural network utilizing insights from neuroanatomy and neuropsychology for sarcasm detection in images and text. The method incorporates gating and guided attention for feature interaction. Zhang et al. [69] propose a multimodal-multitask framework for sarcasm detection utilizing single and multi-level decoders. This approach employs intramodal and intermodal attention to comprehend contextual dependencies among text, visual, and acoustic features. Ghosh et al. [70] design a commonsense-aware framework for emotion recognition for multimodal mental health conversations. This method integrates commonsense reasoning and devises a gating mechanism to eliminate unwanted noise. Lv et al. [71] observe that the lack of synchronization between modalities adds complexity to the task of achieving effective multimodal fusion. Consequently, the study devises an approach to handle the issue of asynchrony of modalities in multimodal streams for sentiment and emotion detection. A few works use a transformer as the backbone for multimodal hate content detection [72] and multimodal representation for languages [73].

The aforementioned works mostly rely on the attention mechanism for feature interaction among different modalities. However, feature interaction through a cross-attention mech-

anism may result in inter-modal incongruity due to different learning dynamics between modalities and diverse noise patterns.

2.2.3 Dark Humor and Related Provocative Content

Dark humor, while increasingly present in online discourse, remains underexplored in computational linguistics. Unlike general humor detection, dark humor poses unique challenges due to its reliance on morally deviant, taboo, or emotionally provocative content. Ruch et al. [74] identify four rhetorical styles central to dark humor, sarcasm, cynicism, satire, and irony, each involving complex linguistic and psychological cues. While studies directly targeting dark humor are rare, significant advances have been made in detecting its constituent forms.

Sarcasm detection has seen considerable progress, particularly through multimodal approaches. Bao et al. [75] proposed HIAN, which uses BiLSTM, attention, and transformers to fuse text, image, and class-based features, capturing cross-modal inconsistencies indicative of sarcasm. Gupta et al. [76] extended this with MLFTI, a weighted multimodal fusion model combining text and visual cues in a shared latent space, achieving 93% accuracy on Twitter data. Jia et al. [77] addressed generalization challenges with DMSD-CL, which uses counterfactual data augmentation and contrastive learning to reduce bias from spurious correlations. Liang et al. [78] introduced MMGCL, a graph-based contrastive learning model that integrates text, OCR, and visual objects, effectively modeling sentiment incongruity, an important signal in sarcastic and darkly humorous content. Irony detection has also been actively explored. Saroj and Pal [79] demonstrated the importance of domain adaptation with an ensemble of contextual models fine-tuned for political discourse. Maladry et al. [80] achieved strong results in Dutch by combining transformer models with engineered sentiment-clash features. In a low-resource setting, Bölücü and Can [81] employed UCCA-based graph neural networks for Turkish irony detection, achieving an F1-score of 94.85%, and demonstrating the value of structured semantic modeling. Satire, though less studied, has seen recent computational advances. Goyal et al. [82] contrasted satire with sarcasm in a multimodal framework, combining BERT, ResNet, and I3D to capture linguistic and visual nuances. Ortega-Bueno et al. [83] proposed MvAttLSTM, a multilingual model that integrates linguistic, sentence-level, and contextual embeddings to detect satire and irony in

Spanish. Kamal et al. [84] introduced BiSAT, a BiLSTM with self-attention and handcrafted linguistic features, achieving strong results on short satirical texts across multiple datasets.

While these studies provide a robust foundation, several limitations remain. Most notably, research tends to treat sarcasm, irony, satire, and related styles as isolated tasks, overlooking the fact that they frequently co-occur or interact within a single instance of humor, particularly in dark humor. Though there are a few works [85, 86], cynicism remains virtually unexamined as a computational task. Furthermore, many models are tailored to narrow domains or specific modalities, limiting their generalizability across contexts, cultures, and moral boundaries. These gaps highlight the necessity of framing sarcasm, irony, satire, and cynicism not as independent phenomena but as interrelated components of a broader category, dark humor. A unified modeling approach is essential to capture the layered, morally provocative, and affectively ambiguous nature of dark humor in real-world discourse.

2.3 Video-based Implicit Hostile Content Detection

Hate speech detection in videos is an emerging field, yet many existing works focus only on explicit hate. For example, HateMM [87] introduced a dataset of 1,083 English BitChute videos aimed at binary hate classification only. Similarly, MultiHateClip [88] provides a multilingual dataset of 2,000 YouTube and Bilibili videos with fine-grained labels in English and Chinese, but these labels predominantly target explicit hate content. Additionally, studies by Al- cântara et al. [89] and Wu and Bhandary [90] have advanced video-based hate speech detection; however, they too tend to overlook the subtleties of implicit hate. This gap underscores the need for comprehensive approaches that not only detect overt hateful content in videos but also capture the nuanced and often overlooked manifestations of implicit hate speech. Recent studies on implicit hate speech detection have primarily focused on text-based approaches. For instance, ElSherief et al. [18] and Kim et al. [91] have advanced the field using linguistic analysis to capture subtle hate cues, while Ocampo et al. [92] and Guo et al. [19] further enhanced detection by integrating emotion, ambiguity, and multi-feature fusion techniques. Additionally, efforts such as ToxiGen [93] have leveraged machine-generated data to improve model robustness. Despite these advances, the majority of existing work relies solely on textual information, neglecting the rich, multimodal

2.3 Video-based Implicit Hostile Content Detection

context inherent in video content. In contrast, our study is the first to explore implicit hate speech detection in videos, incorporating not only text but also visual and audio modalities to capture a more comprehensive spectrum of hateful content.

Chapter 3

Multilingual Hostile Content Detection

3.1 Introduction

This chapter focuses on multilingual hostile content detection in low-resource Indic languages. The major reasons for undetected abusive content are complexities that arise due to multilingualism and code-mixed content [7]. The objective of this chapter is to deal with the challenges that exist in detecting multilingual abusive content on social media.

Due to the multitude of languages spoken globally, it is common for individuals to communicate in multiple languages on social media. Some of these languages are low-resource, which further complicates the challenging task of abuse identification. Languages with a relatively limited amount of data available for training are referred to as low-resource languages, with many Asian and African languages falling under this category. Indic languages such as Telugu and Bengali are a few examples of low-resource languages [94], [95]. Many of the West-European languages, such as English, French, Spanish, and Italian are high-resource languages, with English being the most well-resourced language [96]. The practice of writing content in native low-resource languages using Roman script is widespread today. Transcribing a native language using Roman alphabets results in several textual inconsistencies¹. For instance, the Romanized Hindi word *humara* can be transcribed in multiple spelling structures as *humaraa*, *hamara*, with all forms being acceptable. These multiple spelling structures of words hamper the performance of abusive content detection methods as they fail to recognize each occurrence of these words. Thus, it is essential to develop an

¹K. Guha, Why writing hindi in r oman rather than devanagri would be a disaster (2015).

efficient technique to detect the proliferation of abusive content in Romanized multilingual text.

Most social media platforms have made it convenient for users to write their content in complex code-mixed text; that is, people post their content using words of more than one language. For instance, the sentence *Main lunch break me call karunga tumhe*, which translates to English as *I will call you during the lunch break*, is a Hindi-English code-mixed sentence. This code-mixed sentence uses English words *lunch break* and *call* in a Hindi structured sentence. Code-mixed content is vastly used for low-resource languages, but the grammatical structure used for code-mixing may vary. The grammatical structure can be either in English or the native language depending on the user’s taste [97]. The above-mentioned code-mixed sentence follows the Hindi grammatical structure. In contrast, the sentence, *Mere maa-papa are the best in the world*, which translates to English as *My parents are the best in world*, follows English grammatical structure. Both of the above-mentioned sentences are Hindi-English code-mixed sentences that follow different grammatical structures. The intricate nature of such cases presents a challenge in recognizing and controlling inappropriate content in low-resource code-mixed texts.

The emergence of advanced text feature extraction techniques, such as transformer-based methods, has led to a significant increase in sole focus on textual data among many researchers [9]. Even though text is the primary feature, social context features such as likes and reports can also provide useful insights about the nature of the content [36]. To cite an instance of the same, we observe in the SCIDN dataset that abusive comments receive more reports and fewer likes than non-abusive comments. Such insights can be utilized alongside textual modality. Moreover, incorporating users’ history and disposition of posts can be beneficial for abusive content detection. Disposition of a post or post affinity refers to the distribution of abusive and non-abusive comments on the post. Earlier works do not capitalize on the collective benefits of all the aforementioned perspectives for abusive content detection [98].

Our motivation to work with Indic languages can be attributed to the vast multilingualism in India. As per the census, there are a total of 1,369 rationalized mother tongues in India, that are spoken by 10,000 or more speakers, and they are further grouped into a total of 121

3.1 Introduction

languages, out of which 22 are officially recognized by the constitution of India ². Also, India has the second highest number of internet as well as social media users in the world. Recently, WhatsApp banned over 2.6 million accounts in India through its abuse detection system³, which shows the massive amount of abusive content present on the internet, a lot of which is propagated in Indic languages. In light of the above, it has become essential to focus on a robust method to detect abusive content in Indic languages.

Table 3.1: Examples of comments

Comments	Translation	Label
1 Ninkkal eppaṭi irukkirirkal... all good ?	<i>How are you...all good?</i>	Non-abusive
2 Busy hu main, work load zyada hai.	<i>I am busy due to heavy work load</i>	
3 Ye sab kya hai? samajh nhi aaya.	<i>What is this? could not understand.</i>	
1 Dont comment here , anaathai kaluthai.	<i>Dont comment here, Orphaned Do**key.</i>	Abusive
2 Iska face to dog jaisa hai.	<i>His face looks like a dog</i>	
3 Kya ch***pa hai ye!!!	<i>What is this nonsense!!</i>	

Table 3.1 demonstrates examples of abusive and non-abusive content in Indic languages. The first example in both categories demonstrates Inter-sentential code-mixing in Tamil-English, where two sentences in a comment are written in two different languages. The second example in both categories represents Intra-sentential code-mixing in Hindi-English. The third example is written only in Hindi in both categories. To this end, we have proposed a cross-lingually trained transformer-based method to tackle these cases by extracting contextual embeddings from user-generated content. The following points summarize our key contributions to the above-mentioned challenges:

1. The suggested method leverages both social context and textual features by first training them in two separate modules to create a high-level representation of the features. It is shown that combining the outputs of these modules results in a significant improvement in performance.
2. We propose a novel multilingual abuse identification framework for multiple Indic languages. Unlike earlier techniques which focus either on monolingual corpora of

²C. of India, Census Data 2001 : General Note,[Online; accessed 19-July-2001] (2001). URL <https://censusindia.gov.in/census.website/data/census-tables/>

³WhatsApp bans over 37 lakh accounts in India in November — telegraphindia.com, <https://www.telegraphindia.com/business/whatsapp-bans-37-16-lakh-accounts-in-india-in-november/cid/1905109>, [Accessed 22-Dec-2022].

regional languages [29], or bilingual corpus of code-mixed language [32], our method achieves state-of-the-art results for a total of 12 major Indic languages.

3. User-generated content on social media usually includes many words that do not add significantly to the information carried by such content. It is imperative to filter out these words during data preprocessing. With this in consideration, we have created a collection of 2,250 words for Indic languages.
4. Another challenge is dealing with spelling errors in user-generated content. Instead of the costly operation of focusing on all spelling errors, our method focuses only on spelling errors of abusive words. To deal with this, we have constructed a comprehensive set of abusive words for Indic languages. The set contains more than 17K abusive words, including multiple spelling structures for every single abusive word. This set has been used for data augmentation and for assisting in the calculation of user and post affinity pertaining to abuse.
5. To deal with the code-mixed aspect of the content, we employ a cross-lingual method for text embedding extraction. Further, multiple embeddings are used in the ensemble framework to exploit the combined benefits of transformer-based methods.
6. We have conducted extensive experiments on two datasets to assess the efficacy of the proposed approach. The experimental results demonstrate that our method outperforms state-of-the-art methods by incorporating information from the text as well as social context features.

3.2 Methodology

A detailed discussion of our proposed methodology for abuse identification is presented in this section. Figure 3.1 demonstrates the system architecture of our proposed user-aware abusive content detection system. The proposed system comprises five major components: a) data preprocessing; b) social context features; c) text context features; d) abuse classification, and e) ensemble learning. In the data preprocessing module, the comment text set $C = \{c^i\}_{i=0}^K$ is cleaned and then transliterated to Roman script. Here c^i denotes the i th comment

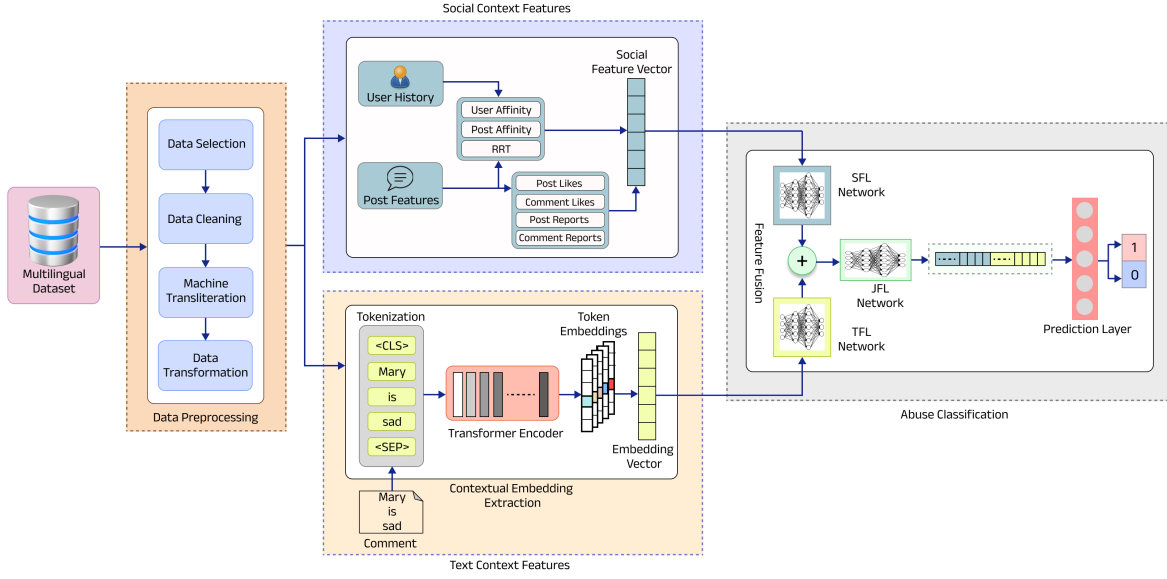


Figure 3.1: System Architecture of Multilingual Abusive Content Detection Framework

in set C , K is the cardinality of C . We denote the transliterated set of comments with $T_c = \{t_c^i\}_{i=0}^K$, where t_c^i is i th transliterated comment. Transliteration is a salient feature of our proposed system. Comments from set T_c are used to get the contextual text embedding vector $\hat{v}$ for each comment $t_c^i \in T_c$ through transformer-based multilingual methods. We employ three transformer-based methods, Muril [99], M-Bert [100] and XLM-R [101] for contextual embedding extraction. Parallely, another module is trained with social context features. High-level representations of text context features and social context features are then fused to get a joint representation $\hat{I}$. The architecture of the social feature learning (SFL) module, text feature learning module (TFL), and joint feature learning (JFL) module is given in Figure 3.3. Finally, a majority voting with confidence decision is employed for final classification.

3.2.1 Data Preprocessing

The data preprocessing step is pivotal as it sets the data up for future in-depth analysis in the most meaningful way possible. A pictorial representation of data preprocessing is given in Figure 3.2. Data preprocessing steps carried out in the proposed approach are elaborated as follows:

3.2.1.1 Data Selection

The ShareChat IndoML Datathon NSFW (SCIDN) dataset contains comments from more than 15 languages [102]. Some of them are dialects of Hindi, such as Haryanvi and Bhojpuri. We select comments from 12 prominent Indic languages, namely, Assamese, Bengali, English, Gujarati, Hindi, Kannada, Malayalam, Marathi, Odia, Punjabi, Tamil and Telugu⁴. We take comments of 10 similar languages from the IIIT-D Multilingual Abusive Comment Identification dataset (MACI) except for English and Punjabi as comments from these two languages are not present in that dataset [103]. Table 3.4,3.5,3.6 and 3.7 in Section 3.3.1.1 give a detailed overview of both datasets.

3.2.1.2 Data Cleaning

In this step, we remove unexpected erroneous entries from the datasets, such as rows with missing comments and missing feature values. Then comment-level cleaning is performed, where we remove extra spaces, and punctuation. We observe digits in a good number of comments. When digits are observed in particular patterns or formats, they can carry valuable meaning. For instance, in the context of named entity recognition, the detection of patterns like dates e.g., *02/11/2023* plays a vital role in extracting relevant information. Nevertheless, it is crucial to acknowledge that the relevance of digits in text classification problems is contingent upon the particular problem at hand, the dataset used, and the domain under consideration. Hence, we conduct experiments with and without digits. For the SCIDN dataset, we achieve an F1-score of 95.38% without digits and 95.33% with digits. For the MACI dataset too, results are slightly better without digits with 91.58% F1-score as compared to 91.51% F1-score with digits. Based on the results, we decide to remove digits as the results are very similar (slightly less) with the digits. Next, insignificant words are filtered out from comments utilizing our prepared set of such words. English words are taken from the NLTK library [104]. For other languages, we translate English words into Indic languages. We also collect a few words from different internet sources to create the set.

⁴C. of India, Census Data 2001 : General Note,[Online; accessed 19-July-2001] (2001). URL <https://censusindia.gov.in/census.website/data/census-tables/>

3.2.1.3 Machine Transliteration

Transliteration is a process to convert sentences written in one script to another script based on phonetic similarity. Many social media platforms give the option to write comments in native scripts as well as in Roman script. In our experiments, we observe that embeddings from transliterated comments achieve better results than the rest of the models. In our pipeline, all comments are transliterated into Roman script. For a given comment $c \in \mathcal{C}$, t_c denotes the comment after transliteration where $t_c \in \mathcal{T}_c$.

$$t_c = \text{IndicXlit}(c) \quad (3.1)$$

We use IndicXlit [105] method for transliteration. Roark et al. [106] also puts forward a transliteration model for South Asian languages, which exhibits an average accuracy of 51.83%. In contrast, IndicXlit surpasses this performance with an accuracy of 72.12% due to the implementation of a re-ranking mechanism. Consequently, we opt to adopt IndicXlit. The largest publicly accessible parallel corpus with 26 million word pairings from Indic languages, Aksharantar, serves as the training data for IndicXlit. Dakshina and Aksharantar benchmarks are used for its evaluation. It employs six layers of encoders and decoders, 256-D input embedding vectors with four attention heads, and a 1024 feedforward dimension for a total of 11M parameters. It currently supports 21 languages. We use transliterated comments for contextual embedding extraction for M-Bert and Muril, while for XLM-R we take comments in their original script as it shows better performance with them.

3.2.1.4 Data Transformation

Plenty of comments in the dataset contain emojis such as 😊, 😞 or emoticons such as :), :-]. These days people mostly use emojis to express anger, disgust, and happiness while posting comments [107]. Emojis can be utilized to get valuable emotional insights from a comment. We, therefore, transform all emojis into text. Next, uppercase characters are transformed into lowercase characters.

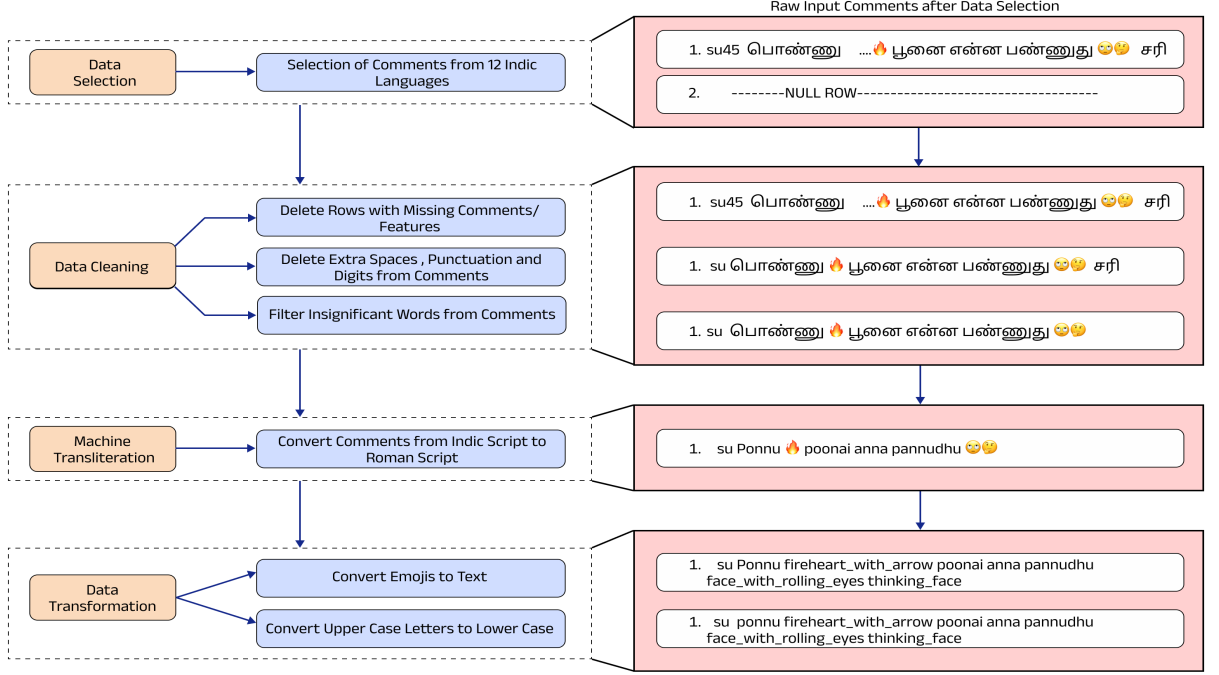


Figure 3.2: Representation of Data Preprocessing Steps

3.2.2 Social Context Features

In numerous existing works related to abuse detection, only textual features have been extensively used with Machine Learning and Deep Learning models [36], [37]. A few incorporated user behavior as well [32], [33]. One objective of this chapter is to investigate the impact of both social context and textual features on abusive content detection methods. Broadly, social context features include post features and user history. We present correlation scores of these features later in this section to establish their importance. Correlation scores show that our proposed feature, that is user-post affinity, has the strongest correlation with the class labels.

In the dataset, we have comments made on different posts, made by different users. Let's assume a set of posts $P = \{p^i\}_{i=0}^X$ and a set of users $U = \{u^j\}_{j=0}^Y$, where X represents the cardinality of set P and Y denotes the cardinality of set U . There may be more than one comment on a single post and more than one comment made by a single user in the dataset. Hence, all the comments in the dataset can be represented either as the union of the set of comments made on each post or the union of the set of all comments made by each user as

follows:

$$C = \{C_p^0\} \cup \{C_p^1\} \cup \{C_p^2\} \dots \dots \{C_p^X\} \quad (3.2)$$

$$C = \{C_u^0\} \cup \{C_u^1\} \cup \{C_u^2\} \dots \dots \{C_u^Y\} \quad (3.3)$$

where, C denotes the set of all comments, C_p^i denotes the set of comments made on a post p^i and C_u^j denotes the set of comments made by a user u^j .

We calculate two features, user affinity, and post affinity. For that, we use the set of abusive words $A = \{a^i\}_{i=0}^{Q_1}$ collected by us, where a denotes a single abusive word and Q_1 is the cardinality of set A . We then prepare an extended set, $A_{ext} = \{a_{ext}^i\}_{i=0}^{Q_2}$, with different spellings of each abusive word from set A by replacing a few letters, where a_{ext}^i denotes a single abusive word and Q_2 is the cardinality of set A_{ext} . Abusive words of Hindi are taken from Mathur et al. [97] and English words are taken from the better profanity repository⁵. For other languages, we collect the abusive words from different internet sources, and we also translate the Hindi and English words into the remaining Indic languages. Next, we make different spellings of all words by replacing vowels with each other and by replacing a few letters with similar-sounding letters. This is done to cover the potential spelling mistakes that can be made by users. Next, we remove the duplicates from the list, thus getting 17K abusive words.

3.2.2.1 Post Features

Post features provide valuable information about the content, such as the number of likes and reports. Generally, the pattern of abusive content pertaining to the number of likes and reports would be different from non-abusive content [108]. Therefore, we include post features in training. Following post features are available in both the datasets, namely, *post id* (p), *report count comment* (r_u), *report count post* (r_p), *like count comment* (l_c) and *like count post* (l_p), where $p \in P$. Apart from these, we determine two more post features, *post affinity* and *relative reporting tendency*, as follows:

⁵https://github.com/snguyenthnh/better_profanity/blob/master/better_profanity/profanity_wordlist.txt.

Relative Reporting Tendency (RRT): For any given comment c , let r_c be the number of times it was reported, and, let r_p be the number of times all the comments on that post were reported. Then *relative reporting tendency* (r_{rt}) can be determined as follows:

$$r_{rt} = \frac{r_c}{r_p} \quad (3.4)$$

Post Affinity: Post affinity gives the likelihood of a post attracting an abusive or non-abusive comment. For a given set of comments C_p^i on a post p^i , the number of all abusive comments in this set can be written as $|\sigma_{L=1} C_p^i|$ and the number of all non-abusive comments as $|\sigma_{L=0} C_p^i|$. Where $L = 1$ and $L = 0$ denote an abusive and non-abusive comment respectively and σ denotes the selection operator. *Post affinity* (ϕ_p) can be calculated as:

$$\phi_p = \frac{|\sigma_{L=0} (C_p^i)| - |\sigma_{L=1} (C_p^i)|}{m} \quad (3.5)$$

where m is the total of number comments on post p^i . Similar steps, as shown in Algorithm 3.1 for the user affinity feature can be repeated for determining post affinity.

3.2.2.2 User History

User history can be utilized to determine the likelihood of a comment being abusive or non-abusive. Generally, a comment written by a user, who has made more abusive comments in the past, is more likely to be abusive [109].

User Affinity: User affinity feature gives the user's tendency to make an abusive or non-abusive comment independent of the nature of the post. For a given set of comments C_u^j made by a user u^j , the number of all abusive comments in this set can be written as $|\sigma_{L=1} C_u^j|$ and the number of all non-abusive comments as $|\sigma_{L=0} C_u^j|$. *User affinity* (ϕ_u) can be calculated as:

$$\phi_u = \frac{|\sigma_{L=0} (C_u^j)| - |\sigma_{L=1} (C_u^j)|}{n} \quad (3.6)$$

where n is the total number of comments made by the user u^j . Algorithm 3.1 gives the

3.2 Methodology

steps for calculation of ϕ_u .

Algorithm 3.1 User-Post Affinity

Input: $T_{c,u}^j$: set of comments made by a user u^j
 $T_{c,p}^i$: set of comments on a post p^i
 A_{ext} : set of abusive words

Output: ϕ_u : User affinity of user u
 ϕ_p : Post affinity of post p
 ϕ_{up} : Combined user and post affinity

function: $\text{user_post_affinity}(A_{ext}, T_{c,u}^j, T_{c,p}^i)$

1: $\phi_u \leftarrow \text{user_affinity}(A_{ext}, T_{c,u}^j)$

2: $\phi_p \leftarrow \text{post_affinity}(A_{ext}, T_{c,p}^i)$

3: $\phi_{up} \leftarrow \alpha(\phi_u) + (1 - \alpha)\phi_p$

4: **return** $\phi_{up}, \phi_u, \phi_p$

function: $\text{user_affinity}(A_{ext}, T_{c,u}^j)$

1: $\text{count}_{abuse} \leftarrow 0$

2: $\text{count}_{non} \leftarrow 0$

3: **for all** $t_c \in T_{c,u}^j$ **do**

4: $\text{flag} \leftarrow 0$

5: **for all** $a_{ext} \in A_{ext}$ **do**

6: **if** a_{ext} **in** t_c **then**

7: $\text{flag} = 1$

8: **break**

9: **end if**

10: **end for**

11: **if flag is 1 then**

12: $\text{count}_{abuse} \leftarrow \text{count}_{abuse} + 1$

13: **else**

14: $\text{count}_{non} \leftarrow \text{count}_{non} + 1$

15: **end if**

16: **end for**

17: $\phi_u^{ext} \leftarrow \frac{\text{count}_{non} - \text{count}_{abuse}}{\text{count}_{non} + \text{count}_{abuse}}$

18: $\phi_u \leftarrow \max(\phi_u^{ext}, \phi_u^{cls})$

19: **return** ϕ_u

3.2.2.3 Combined User-Post Affinity

A user's behavior may depend on the type of post. For a post that receives mostly abusive comments, any arbitrary user will likely write an abusive comment on the post [110]. However, it may not always be true as user tendency would also have a role to play in determining this. So, we calculate a combined user-post affinity feature that considers both affinities with

appropriate weights given to each. User-post affinity is calculated as follows:

$$\phi_{up} = \alpha (\phi_u) + (1 - \alpha) \phi_p \quad (3.7)$$

We perform the experiments with different values of α ranging from 0 to 1 in increments of 0.1. We observe that it gives relatively better results at 0.4 and 0.5. To perform additional testing, we varied the value of α for this feature within the range of 0.4 to 0.5 with an incremental step of 0.01. Our observation indicates that the best results are achieved when α is set to 0.47. One of the reasons for α providing better results at 0.47 when post affinity is given marginally higher weight (0.53) is that post affinity has a stronger correlation with the class label than the user affinity, as given in Table 3.2.

To calculate ϕ_u and ϕ_p , we count the number of abusive and non-abusive comments for each user and post. Test data may not necessarily have comments from the same users and posts as training data. So, we take the maximum of affinities determined using pre-classifier and lexicon matching approaches for test data. For this, we prepare a large set of abusive words comprising words from all 12 languages. With the help of this set, we determine the values of ϕ_p^{ext} and ϕ_u^{ext} . To determine affinities ϕ_p^{cls} and ϕ_u^{cls} , Muril is taken as the pre-classifier. ϕ_u^{ext} and ϕ_p^{ext} denote the user and post affinities respectively, determined using an extended abusive set. ϕ_u^{cls} and ϕ_p^{cls} denote the user and post affinities respectively, determined using the pre-classifier. For the MACI dataset, we use only the post affinity feature as user-ids are not available in that dataset.

Correlation: We use Point Biserial correlation [111] to represent the association of features with class labels. Point Biserial correlation coefficient is used to measure the correlation between a continuous and a dichotomous variable. It returns values that range from -1 to 1. A coefficient value of 1 represents a perfect positive correlation, -1 represents a perfect negative correlation, and 0 represents no correlation. Results are presented in Table 3.2 and Table 3.3 for SCIDN and MACI datasets, respectively. User-post affinity shows the strongest correlation, whereas post-id has the weakest correlation with class labels in SCIDN. In MACI, post affinity has the strongest correlation, while post-id has the lowest correlation. Correlation for embeddings is calculated for Muril’s predicted labels.

We concatenate the selected features and create a social feature vector $\hat{s}$. Next, the SFL

Table 3.2: Point Biserial correlation of features with class labels in SCIDN dataset

Feature	Point Biserial
Post id	0.002
Like count comment	-0.097
Like count post	0.136
Report count comment	-0.010
Report count post	0.105
Relative reporting tendency	-0.019
Post affinity	-0.906
User id	0.157
User affinity	-0.752
User-post affinity	-0.911
Contextual Embeddings	0.873

Table 3.3: Point Biserial correlation of features with class labels in MACI dataset

Feature	Point Biserial
Post id	-0.007
Like count comment	-0.080
Like count post	-0.055
Report count comment	-0.022
Report count post	-0.007
Relative reporting tendency	-0.028
Post affinity	-0.857
Contextual Embeddings	0.752

module is trained on $\hat{s}$. We give a detailed description of the SFL module in Section 3.2.4.1.

3.2.3 Text Context Features

In this section, we present a detailed overview of the text context features. It comprises two subsections: a) data augmentation, and c) contextual embedding extraction. Data augmentation is employed by inserting additional synthetic comments in the dataset using a random sample of original comments from the training data. It is done with the objective of dampening the effect of misspelled abusive words. Next, contextual embeddings are extracted and fed to the TFL module. The rest of this section describes these steps in detail.

3.2.3.1 Data Augmentation

We observe that the same words were transliterated differently in different comments. Users, too, tend to make spelling mistakes while writing comments. To handle this problem, we use the data augmentation technique. For that, we use the extended set of abusive words A_{ext} . Set T_c is divided into train set T_c^{train} and test set T_c^{test} as shown in Equation 3.8.

$$T_c = T_c^{train} \cup T_c^{test} \quad (3.8)$$

From the train set T_c^{train} , we sample a random set of non-abusive comments having the same cardinality as A_{ext} . Each abusive word $a_{ext} \in A_{ext}$ is inserted into a comment. Original comments are also retained in the train set T_c^{train} . Since the insertion of the abusive word makes the comment abusive, the label is also inverted for these synthetic comments. This process is repeated for each language and we get an augmented set, T_c^{aug} , which is combined with the train set to get the final augmented training set $T_c^{train_aug}$ as shown in Equation 3.9.

$$T_c^{train_aug} = T_c^{train} \cup T_c^{aug} \quad (3.9)$$

This set of additional comments, T_c^{aug} , helps the model in two aspects, first, training samples have increased for each language while covering spelling errors in abusive words, secondly, the model can give more attention to words which make a comment abusive since the training data contains the non-abusive copy of the same comments as well. We argue that synthetic comments still make sense semantically because an abusive word is added at the start of each sentence. Generally, most of the abusive words are person or group-oriented [112], hence, adding them at the start does not alter the grammar of the sentences, and semantics of the sentences are retained while the polarity changes from non-abusive to abusive. Steps of data augmentation are presented in Algorithm 3.2.

In Algorithm 3.2, line 1 initializes T_c^{aug} as an empty set. In lines 2-6, for each language, we iterate over a fixed number of comments equal to the number of abusive words for that language. Then, we insert an abusive word in the comment and the comment is added to T_c^{aug} set. In line 7, we add the augmented set T_c^{aug} to the training set. Finally, the augmented

3.2 Methodology

Algorithm 3.2 Data Augmentation

Input: T_c^{train} : training set of comments u_j
 A_{ext} : extended set of abusive words
Output: $T_c^{train_aug}$: augmented training set of comments
function: $\text{augment}(T_c^{train}, A_{ext})$
1: $T_c^{aug} \leftarrow \{ \}$
2: **for** $i \in \text{language}$ **do**
3: **for each** $a_{ext} \in A_{ext}^i$ **and** $t_c \in T_c^{train,i}$ **do**
4: $T_c^{aug} \leftarrow T_c^{aug} \cup t_c.\text{insert}(a_{ext})$
5: **end for**
6: **end for**
7: $T_c^{train_aug} \leftarrow T_c^{train} \cup T_c^{aug}$
8: **return** $T_c^{train_aug}$

training set $T_c^{train_aug}$ is returned.

3.2.3.2 Contextual Embedding Extraction

We employ three different methods to generate embeddings for a tokenized sequence of words, namely, Muril, XLM-R, and M-Bert. First, tokenization of a transliterated comment $t_c \in T_c$ is done. A tokenized comment is represented as a sequence of tokens $W = \{w^i\}_{i=0}^{len}$, where W denotes the set of tokens, w is a single token, and len denotes the total number of tokens in the comment. Next, two unique tokens, the class [CLS] token and separator [SEP] token, are inserted at the beginning and at the end, respectively. The set of tokens is then converted into a set of integers represented by $word_ids$ in Equation 3.10.

$$words_ids = tokens_to_ids(W) \quad (3.10)$$

We perform experiments for input sequence lengths of 128 and 64, in the ensemble framework. Any comment with the number of tokens exceeding the sequence length is truncated. For comments shorter than the sequence length, padding tokens are added. Then, we create $input_mask$ by adding 1s a number of times which is equal to the length of the tokenized sequence, followed by padding by 0s to the appropriate length, if required. This helps the model to distinguish between original tokens and padding tokens. Then, we create $input_type_ids$ which helps in determining the positional encoding of a sentence. Finally, we pass all three inputs to the encoder as shown in Equation 3.11, which generates the sen-

tence embedding $E \in \mathbb{R}^D$ and the last hidden state matrix $H \in \mathbb{R}^{l \times D}$.

$$E, H \leftarrow \text{encoder}(\text{words_ids}, \text{input_mask}, \text{input_type_ids}) \quad (3.11)$$

Here, l is the sequence length and D is the dimension of each token embedding which is 768, $\mathbb{R}$ symbolizes the set of real numbers. We use the last hidden state to create the text embedding vector which is discussed in detail in Section 3.2.4.2. We use encoders of the three aforementioned methods to generate the embeddings.

3.2.4 Abuse Classification

In this section, we discuss the abuse classification module. Various components of the abuse classification module are shown in Figure 3.3. Four components constitute this module, namely, a) social feature learning; b) text feature learning; c) joint feature learning, and d) prediction layer.

3.2.4.1 Social Feature Learning (SFL)

We use six social context features, namely, report count post (r_p), like count comment (l_c), like count post (l_p), relative reporting tendency (r_{rt}), and combined user-post affinity (ϕ_{up}). Here r_{rt} and ϕ_{up} are the manufactured features. For the MACI dataset, we use report count comment (r_c) instead of r_p , as its correlation coefficient is higher as shown in Table 3.3. We first scale these features using min-max normalization. Min-max normalization method is represented in Equation 12. Let x , x_{min} , and x_{max} denote the original value of a feature instance, the minimum value for that feature, and the maximum value for that feature, respectively. Then, min-max normalized value x_{norm} of x would be:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3.12)$$

Next, a joint representation of social context features is created by concatenating these features as shown in Equation 3.13. We call this joint representation a social feature vector, denoted by $\hat{s}$.

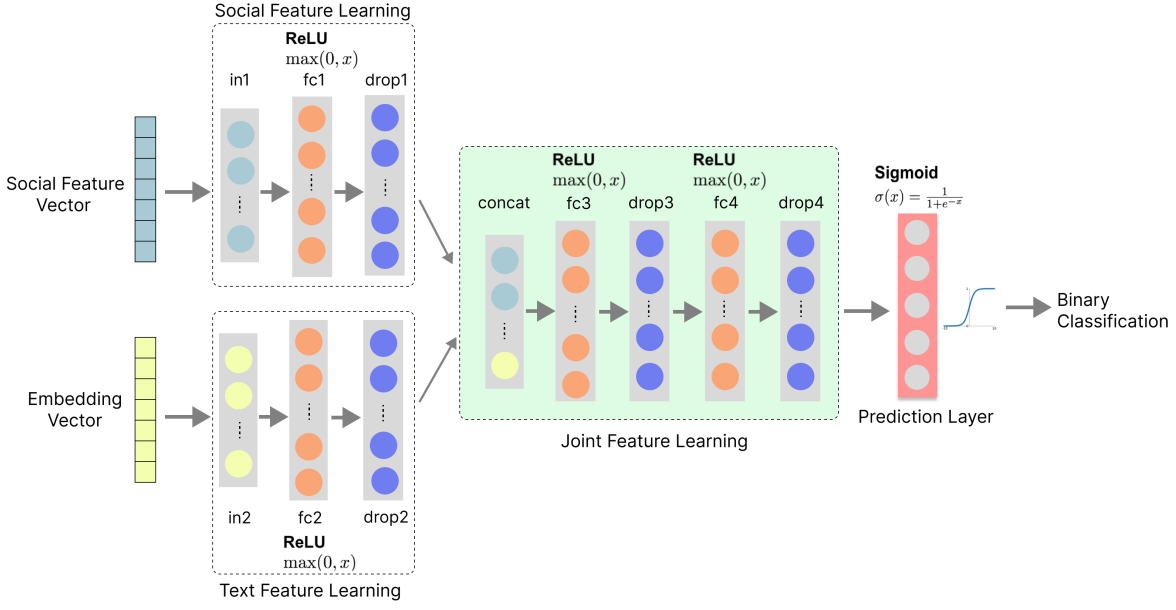


Figure 3.3: Architecture of Abuse Classification Module

$$\hat{s} = f_{cat}(r_p \oplus l_c \oplus l_p \oplus r_{rt} \oplus \phi_{up}) \quad (3.13)$$

Here, $\hat{s} \in \mathbb{R}^M$, M denotes the dimension of $\hat{s}$ vector, f_{cat} denotes the concatenation function and $\oplus$ denotes the concatenation operation. Vector $\hat{s}$ is then passed to input layer $in1$ of the SFL module which instantiates the tensor $\overline{T}(\hat{s})$. We pass the tensor $\overline{T}(\hat{s})$ to a fully connected dense layer $fc1$ to obtain the high-level representation, $\hat{S}$, of the social feature vector $\hat{s}$ as given in Equation 3.14.

$$\hat{S} = fc1(units = D_1)(\overline{T}(\hat{s})) \quad (3.14)$$

The layer $fc1$ performs dot product between $\overline{T}(\hat{s})$ and weight matrix $W_1^{(r)}$. The operations performed by $fc1$ are represented by Equation 3.15.

$$fc1: \quad \hat{S} = \sigma_{relu}\left(W_1^{(r)} \cdot \overline{T}(\hat{s}) + b_1\right) \quad (3.15)$$

Here, $\hat{S} \in \mathbb{R}^{D_1}$, D_1 denotes the dimension of $\hat{S}$ which is 16, $W_1^{(r)}$ is the weight matrix of dimension r which is $D_1 \times M$ i.e., 16×5 , b_1 is bias vector of size D_1 and σ_{relu} denotes

the activation function. We use Rectified Linear Unit (ReLU) as the activation function for $fc1$ [113]. It adds non-linearity to the model. The ReLU function is represented in Equation 3.16 with x as input.

$$\sigma_{relu}(x) = \begin{cases} 0, & \text{if } x < 0 \\ x, & \text{if } x \geq 0 \end{cases} \quad (3.16)$$

ReLU makes the computation of gradient extremely simple as it gives either 0 or x as output which depends on the sign of x . 0 is returned for non-positive values, otherwise, x is returned. It helps in speeding up the training of the neural network. To prevent overfitting, 20% of the neurons are randomly dropped. Next, $\hat{S}$ is fused with the high-level representation of the text embedding vector $\hat{v}$ for joint learning. Joint feature learning is explained in detail in Section 3.2.4.3.

3.2.4.2 Text Feature Learning (TFL)

This section illustrates the text feature learning module. As mentioned in Section 3.2.3.2, transformer-based multilingual methods are used for contextual embedding extraction [114]. They generate the sentence embedding $E \in \mathbb{R}^D$ and the last hidden state matrix $H \in \mathbb{R}^{l \times D}$. We use the last hidden state $H \in \mathbb{R}^{l \times D}$ for our experiments. As shown in Equation 3.17, all the token embeddings from the last hidden matrix H are stacked horizontally to create embedding vector $\hat{v}$.

$$\hat{v} = Reshape(H) \quad (3.17)$$

Here, $\hat{v} \in \mathbb{R}^N$, where N is the dimension of the text embedding vector. N , dimension of vector $\hat{v}$, depends on the sequence length. We illustrate the TFL module using a sequence length of 128. Thus N becomes 98,304. Next, embedding vector $\hat{v}$ is passed to the input layer $in2$ of the TFL module which instantiates the tensor $\overline{T}(\hat{v})$. We pass the tensor $\overline{T}(\hat{v})$ to the fully connected dense layer $fc2$ to obtain the high-level representation, $\hat{V}$, of the

3.2 Methodology

embedding vector $\hat{v}$ as given in Equation 3.18.

$$\hat{V} = fc2(units = D_2)(\bar{T}(\hat{v})) \quad (3.18)$$

The layer $fc2$ performs dot product between $\bar{T}(\hat{v})$ and weight matrix $W_2^{(y)}$. All the operations performed by $fc2$ are represented by Equation 3.19.

$$fc2 : \quad \hat{V} = \sigma_{relu} \left(W_2^{(y)} \cdot \bar{T}(\hat{v}) + b_2 \right) \quad (3.19)$$

Here, $\hat{V} \in \mathbb{R}^{D_2}$, D_2 denotes the dimension of $\hat{V}$ which is 768, $W_2^{(y)}$ is the weight matrix of dimension y which is $D_2 \times N$, i.e., $768 \times 98,304$, b_2 is bias vector of size D_2 and σ_{relu} denotes the activation function. ReLU is used in $fc2$ too as an activation function. As in the SFL module, we randomly drop 20% of the neurons in TFL.

3.2.4.3 Joint Feature Learning (JFL)

This module employs a feature fusion mechanism to obtain a joint representation of the learned social feature vector $\hat{S}$ and the text embedding vector $\hat{V}$. Then, the module learns this joint representation [115]. As shown in Equation 3.20, we first pass $\hat{V}$ and $\hat{S}$ to the *concat* layer which concatenates both vectors and outputs a joint feature vector $\hat{I}$.

$$\hat{I} = concat(\hat{V} \oplus \hat{S}) \quad (3.20)$$

Here, $\hat{I} \in \mathbb{R}^{D_3}$, D_3 is the dimension of the joint feature vector which is 784, *concat* denotes the concatenation layer and $\oplus$ denotes the concatenation operation. This 784- D vector is passed to two fully connected layers sequentially, with a dropout layer after each fully connected layer. Equations 3.21, 3.22, 3.23, and 3.24 represent further operations on the joint vector.

$$\hat{i}_1 = fc3(units = D_4)(\hat{I}) \quad (3.21)$$

$$fc3 : \quad i_1 = \sigma_{relu} \left(W_3^{(z_1)} \cdot \hat{I} + b_3 \right) \quad (3.22)$$

$$\hat{i}_2 = fc4(units = D_4) \left(\hat{i}_1 \right) \quad (3.23)$$

$$fc4 : \hat{i}_2 = \sigma_{relu} \left(W_4^{(z_2)} \cdot \hat{i}_1 + b_4 \right) \quad (3.24)$$

We obtain two intermediate vector outputs of dimension D_4 , $\hat{i}_1$ and $\hat{i}_2$, from $fc3$ and $fc4$ layers respectively. Based on our experiments, we set D_3 as 100. $W_3^{(z_1)}$ and $W_4^{(z_2)}$ are weight matrices for $fc3$ and $fc4$ of dimension z_1 and z_2 which are 100×784 and 100×100 respectively. b_3 and b_4 biases of $100-D$ are used in $fc3$ and $fc4$. Next, we pass the intermediate vector $\hat{i}_2$ to the prediction layer for final classification.

3.2.4.4 Prediction Layer

We generate class labels for each comment in multilingual comment set C . To obtain the predicted class label, we pass the output of the JFL module, i.e., intermediate vector $\hat{i}_2$, to the prediction layer as represented in Equation 3.25.

$$Y_{pred} = Dense(units = 1) \left(\hat{i}_2 \right) \quad (3.25)$$

Here, Y_{pred} is the predicted probability, and $Dense$ denotes the prediction layer. Next, we apply logistic sigmoid activation on the prediction layer to get a predicted probability value between 0 and 1 [113]. This type of sigmoid activation function can be described by the mathematical formula given in Equation 3.26.

$$\sigma_{sig}(x) = \frac{1}{1 + e^{-x}} \quad (3.26)$$

This sigmoid function accepts any real-valued input and returns a real value between 0 and 1. Next, the loss is calculated to check the difference between the predicted probability Y_{pred} and the ground truth L_{gt} . As we have just two classes, we take binary cross entropy loss L_{BCE} as the loss function for our model. The binary cross entropy loss function computes

loss by taking the following average:

$$L_{BCE} = \frac{1}{B} \sum_{i=0}^B (L_{gt}^i - 1) \log(1 - Y_{pred}^i) - L_{gt}^i \log(Y_{pred}^i) \quad (3.27)$$

where B denotes the batch size. We use the Adam optimizer to obtain the gradients of the loss function and weights are updated accordingly. Then, the probability is converted to one of the binary labels 0 and 1 using the *set_label* function as represented in Equation 3.28.

$$Label = set_label(Y_{pred} \geq threshold, 1, 0) \quad (3.28)$$

If probabilities are greater than or equal to the threshold, they are given the label 1, otherwise, the label 0 is assigned. Label 1 represents the abusive class and label 0 represents the non-abusive class.

3.2.5 Ensemble Learning

Ensemble learning is used to achieve better predictive performance by capitalizing on the predictive effectiveness of different classification algorithms. It can be considered a technique to create a strong classifier by combining multiple base classifiers. In this section, we discuss ensemble learning in detail. This section consists of three subsections: a) methods used in ensembling; b) majority voting and c) confidence decision. Algorithm 3.3 presents the sequence of steps performed in ensemble learning.

3.2.5.1 Methods

We employ three methods for embedding extraction for ensemble learning. We take two text embedding vectors for two different sequence lengths from each of the three different transformer-based methods as depicted by Equation 3.29.

$$\hat{v} = Reshape(TF(t_c)_{128,64}) \quad (3.29)$$

Here, $TF = \{Muril, MBert, XLMR\}$ represents all the contextual embedding generation methods used to transform the comments into embeddings. These two embedding vectors

are generated for 64 and 128 input sequence lengths. Predictions are taken for all the text embedding vectors, thus giving us a total of six abuse classification methods, one for each embedding. Six results, therefore, are taken for a single comment $t_c \in T_c$.

Algorithm 3.3 Majority Voting with Confidence Decision

Input: L^i : set of predicted labels for i th comment
 Y_p^i : set of predicted probabilities of classes
Output: F_{label}^i : final label for i th comment

```

function: majority_voting ( $L^i, Y_p^i$ )
1: if  $\sum_{j=0}^G Label_1^j > 3$  then
2:    $F_{label}^i \leftarrow 1$ 
3: else if  $\sum_{j=0}^G Label_1^j \leq 2$  then
4:    $F_{label}^i \leftarrow 0$ 
5: else
6:    $F_{label}^i \leftarrow confidence\_decision(Y_p^i)$ 
7: end if
8: return  $F_{label}^i$ 

function: confidence_decision( $Y_p^i$ )
1: if  $\sum_{j=0}^3 dist(Y_{pred_1}^j) > \sum_{j=0}^3 dist(Y_{pred_0}^j)$  then
2:   return 1
3: else if  $\sum_{j=0}^3 dist(Y_{pred_1}^j) < \sum_{j=0}^3 dist(Y_{pred_0}^j)$  then
4:   return 0
5: else
6:   return best_model( $Y_p^i$ )
7: end if
  
```

3.2.5.2 Majority Voting

We use majority voting for the final classification. If the majority of the models agree on a label, that label is taken as the final label. Majority voting is mainly used when models' outputs are independent. As shown in Equation 3.30, we pass the set of predicted labels $L^i = \{Label^j\}_{j=0}^G$ and predicted probabilities set for the i th comment to the *majority_voting* function to obtain the final label.

$$F_{label}^i = majority_voting (L^i, Y_p^i) \quad (3.30)$$

Here, F_{label}^i is the final label for comment c^i , L^i denotes the set of predicted labels for a single comment c^i , $Label^j$ denotes the j th label from the set L^i , and G is the cardinality of the set

which is 6. Y_p^i is the set of predicted probabilities for the i th comment. Since we have an even number of models, there may be a tie among the models. For this situation, we employ confidence decision which is described next.

3.2.5.3 Confidence Decision

In the case of an even number of models in ensemble learning, a situation of tie occurs in binary classification when half of the models predict one class, and the other half predict the other class. We employ the confidence decision technique based on the sum of the distance of probabilities from the threshold. If for a label, this sum is higher than that of the other label, this label is selected as the final label.

$$F_{label}^i = confidence_decision(Y_p^i) \quad (3.31)$$

The function of the *confidence_decision* can be written as:

$$f(Y_p^i) = \begin{cases} 1, & \text{if } \sum_{j=0}^3 \text{dist}(Y_{pred_1}^j) > \sum_{j=0}^3 \text{dist}(Y_{pred_0}^j) \\ 0, & \text{otherwise} \end{cases} \quad (3.32)$$

where $Y_p^i = \{Y_{pred}^j\}_{j=0}^G$ denotes the resultant set of probabilities for comment c^i for 6 models, G is the cardinality of the set which is 6. Y_{pred_1} is the probability for 1 to be the label, and Y_{pred_0} is the probability for 0 to be the label. Distance function *dist* calculates the distance of a probability from the threshold value. In the case of tie in sum, which is rare, we take the label given by the Muril embedding model of 128 sequence length as the final label as it is the best-performing individual model.

In the majority voting function of Algorithm 3.3, Line 1 and Line 3 compare and calculate the number of 1s from the set of predicted labels for a single comment. Line 6 calls the confidence decision function in case of a tie between the number of labels of the two classes. In the confidence decision function of Algorithm 3.3, line 1 and line 3 calculate and compare the sum of the distance of probabilities from the threshold for two labels. Based on these comparisons, a label is returned, otherwise, line 6 returns the label given by the best model.

3.3 Experimental Evaluations

This section first discusses the experimental setup and then shows the experimental results to demonstrate the effectiveness of our technique.

3.3.1 Experimental Setup

In this section, we first describe datasets used for experimental evaluations. Next, different methods used for comparative analysis are discussed. We also discuss the evaluation metrics used for evaluating the performance of different methods. The model was trained using the Adam optimiser with a batch size of 128, a learning rate of $1e-5$, and 150 training epochs. All experiments were conducted on a system equipped with an Intel® Xeon® Silver 4215R 3.20 GHz processor, 384 GB RAM, and an NVIDIA Tesla T4 GPU.

3.3.1.1 Summarization of Datasets

We perform abusive language detection on two different datasets, namely, SCIDN and MACI. SCIDN is a publicly available multilingual dataset provided by ShareChat. MACI is a similar multilingual text-based dataset taken from Moj.

- a) *SCIDN*: The SCIDN dataset has more than 1.5M (15,00,838) human annotated comments. After data preprocessing, we are left with 6,63,059 comments. We first take comments for 12 different Indic languages, Assamese (as), Bengali (bn), English (en), Gujarati (gu), Hindi (hi), Kannada (kn), Malayalam (ml), Marathi (mr), Odia (or), Punjabi (pn), Tamil (ta) and Telugu (te). Next, we remove rows with erroneous and null comments. Table 3.4 and 3.5 present a detailed overview of the SCIDN dataset after data preprocessing.

ShareChat is a social media and content-sharing platform that allows users to create, share and discover content in 15 languages of India. Founded in 2015, ShareChat has become one of the most popular social networking platforms in India with over 180 million registered users⁶.

⁶<https://economictimes.indiatimes.com/tech/technology/sharechats-virtual-gifting-earns-50-million/articleshow/94689816.cms?from=mdr>

3.3 Experimental Evaluations

The platform allows users to share short videos, images, and text posts in their preferred languages, including Hindi, Bengali, Telugu, Marathi, and more. It also provides a chat feature that allows users to communicate with others in their preferred language. ShareChat app is part of a parent company that also operates other short video platforms such as Moj and TakaTak. In total, these three platforms have a combined user base of over 400 million users⁷.

Table 3.4: SCIDN dataset description

Label	Percentage
Non-Abusive	3,33,902 (50.36%)
Abusive	3,29,157 (49.64%)

Table 3.5: Language-wise description of SCIDN dataset

Language	Samples	Language	Samples
Hindi	2,66,537	Kannada	29,679
English	79,618	Bengali	22,237
Telugu	61,292	Odia	17,355
Punjabi	54,964	Marathi	16,177
Malayalam	52,790	Gujarati	10,969
Tamil	49,989	Assamese	1,453

b) *MACI*: IIIT-D Multilingual Abusive Comment Identification dataset was taken from the IEEE BigMM challenge which aims to improve abusive content detection on social media for low-resource Indic languages. Comments are collected from the Moj application. MACI contains comments from more than 10 Indic languages. The dataset is fairly balanced for instances of the abusive and non-abusive classes. The raw dataset comprises 6,65,041 human-annotated comments labeled as abusive or non-abusive. After data preprocessing, we are left with 6,65,032 comments. In Table 3.6 and 3.7, we give a detailed description of the MACI dataset.

⁷<https://www.moneycontrol.com/news/business/sharechat-closes-520-million-funding-round-at-5-billion-valuation-8697271.html>

Table 3.6: MACI dataset description

Label	Percentage
Non-Abusive	3,52,376 (52.98%)
Abusive	3,12,656 (47.02%)

Table 3.7: Language-wise description of MACI dataset

Language	Samples	Language	Samples
Hindi	3,07,179	Telugu	97,011
Marathi	72,044	Bengali	22,835
Tamil	69,496	Odia	10,973
Kannada	13,943	Malayalam	40,959
Gujarati	8,828	Assamese	2,780

3.3.1.2 Comparison Methods

We perform a comparative analysis to evaluate the performance of our model. For this purpose, we compare the results of our model with the following methods:

- a) *Classical Methods*: For comparison with classical methods, we first take textual features from two classical methods, count-based and frequency-based surface-level feature methods. These features are then used with SVM, LR, and Random-Forest Machine Learning models. We present results only for the best-performing method for a given classical feature set. SVM is found to give better results with character-level features for both feature extraction techniques. Next, we take the word embedding-based method for text feature extraction. Unlike count-based and frequency-based methods which generate large sparse vectors, word embeddings are dense vectors of fixed size. For comparison, we take non-contextual word embeddings which are specifically trained on our datasets. These embeddings are fed to MLP and LSTM Deep Learning models for predictions. MLP is found to outperform LSTM. The best-performing method's result is shown for word embeddings too. Results are shown in Table 3.8 and Table 3.9 in Section 3.3.2.
- b) *State-of-the-art Baseline Methods*: We evaluate the performance of our method against recent transformer-based multilingual baseline methods. Muril, XLM-R, and M-Bert embeddings are taken and used for classification. Muril is found to give better results

for SCIDN and XLM-R performs the best on MACI. Comparison results are shown in Table 3.8 and Table 3.9 in Section 3.3.2. Next, we compare the efficacy of the proposed method with existing research works. Results of our method are compared with Sazzed et al. [116], Modha et al. [117] and Bansal et al. [98] methods. Detailed results are shown in Table 3.8 and Table 3.9. Lastly, we perform a language-wise comparative analysis between transformer-based methods and the proposed method. The proposed method outperforms existing methods for every language as shown in Table 3.10 and Table 3.11.

3.3.1.3 Evaluation Metrics

Abusive content detection is mainly modeled as a classification problem [1]. Comments are classified into two different classes, abusive and non-abusive. We aim to show that our method gives results with the same efficacy across multiple evaluation metrics. Hence, we use four evaluation metrics for comparison, namely, Accuracy (Acc), Precision (P), Recall (R), and F1-score (F1). F1-score can be a good measure even in the case of unbalanced classes.

Let us suppose that L_{TP} , L_{FP} , L_{TN} , and L_{FN} are the sets of labels predicted correctly as abusive, predicted falsely as abusive, predicted correctly as non-abusive and predicted incorrectly as non-abusive, respectively. Precision is calculated by dividing the number of elements in the set of correctly predicted abusive labels L_{TP} by the total number of actual abusive comments. Equation 3.33 provides the formula for precision.

$$\text{Precision} = \frac{|L_{TP}|}{|L_{TP}| + |L_{FP}|} \quad (3.33)$$

The recall is determined by dividing the number of elements in the set of correctly predicted abusive labels L_{TP} by the sum of the number of elements in L_{TP} set and falsely predicted non-abusive set L_{FN} . The formula for Recall is shown in Equation 3.34.

$$\text{Recall} = \frac{|L_{TP}|}{|L_{TP}| + |L_{FN}|} \quad (3.34)$$

The F1-score metric considers both precision and recall. It is determined as the harmonic

mean of precision and recall. F1-score can be calculated as shown in Equation 3.35.

$$F1 - \text{score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (3.35)$$

The accuracy metric is determined by dividing the total number of correctly predicted comments, be they abusive or non-abusive, by the total number of comments in the dataset. Since our dataset is fairly balanced, we use accuracy too for evaluation. The formula for Accuracy is given in Equation 3.36.

$$\text{Accuracy} = \frac{|L_{TP}| + |L_{TN}|}{|L_{Tp}| + |L_{FN}| + |L_{TN}| + |L_{FP}|} \quad (3.36)$$

Next, we discuss experimental results on both datasets using the aforementioned evaluation metrics.

3.3.2 Experimental Results

In this section, we present experimental results and comparisons. First, the results of the proposed method are compared with classical and baseline state-of-the-art methods. Next, an ablation analysis is shown using different features with transliterated comments. Finally, language-wise comparison is shown with state-of-the-art methods.

3.3.2.1 Performance Comparison on SCIDN Dataset

To validate the efficacy of our method, we compare the results of the proposed method with the existing methods. Comparison results on the SCIDN dataset are shown in Table 3.8.

It can be seen in the table that our method obtains an accuracy score of 95.40%, precision of 95.18%, recall of 95.59%, and F1-score of 95.38%. Our method outperforms classical as well as state-of-the-art baselines by significant margins. The proposed method achieves a performance gain of 6.25%, 3.25%, 9.95%, and 6.7% over the count-based method, and 6.22%, 3.58%, 9.43%, and 6.58% over the frequency-based method in terms of accuracy, precision, recall, and F1-score, respectively. This is due to the fact that count-based and frequency-based methods create text features based on the count of occurrence of each word

3.3 Experimental Evaluations

Table 3.8: Comparison with classical and state-of-the-art baseline methods on SCIDN dataset

	Methods	Acc	P	R	F1
Classical Methods	Count-based [118]	89.15	91.92	85.64	88.67
	Frequency-based [119]	89.18	91.60	86.16	88.79
	Embedding-based [120]	90.26	93.28	86.68	89.86
State-of-the-art Baseline Methods	Muril [1]	92.05	91.09	93.13	92.10
	M-Bert [100]	90.40	91.06	89.62	90.34
	XLM-R [101]	91.64	90.97	92.47	91.71
	Modha et al. [117]	92.21	93.29	90.88	92.07
	Sazzed et al. [116]	89.86	92.12	87.07	89.53
	Bansal et al. [98]	91.97	91.08	92.98	92.02
	Proposed Method	95.40	95.18	95.59	95.38

and frequency of each word in the corpus, respectively. Such text features do not consider the context and similarity of words. On the other hand, we use contextual text features which consider both the context and similarity of words. Besides these baselines, we also take non-contextual word embeddings for comparison with our method. These word embeddings are trained on the SCIDN dataset during the fitting of the model. Our method performs significantly better than the non-textual word embedding-based method by attaining 5.14%, 1.9%, 8.9%, and 5.51% higher accuracy, precision, recall, and F1-score, respectively. Transformer-based models such as Muril, M-Bert, and XLM-R have proved to be very effective for multilingual text classification problems. The major advantage of these methods is that they provide contextual embeddings. Our approach capitalizes on the benefits of these transformer-based methods and also includes post-tendency and user history. Our method outperforms Muril by 3.35%, 4.09%, 2.46%, and 3.28%, M-Bert by 5%, 4.12%, 5.96%, and 5.03% in terms of accuracy, precision, recall and F1-score, respectively. XLM-R is beaten by 3.76%, 4.21%, 3.12%, and 3.67% in accuracy, precision, recall, and F1-score metrics, respectively. We compare our results with Modha et al. [117] which uses an embedding layer with an attention mechanism. Their neural network contains two layers of LSTM as forward and backward layers. Our method achieves a performance gain of 3.19%, 1.89%, 4.71%, and 3.31% over Modha et al. in terms of accuracy, precision, recall and F1-score, respectively. A significant performance gain across all four evaluation metrics establishes the effectiveness of our proposed method. We also compare the results of our method with

Sazzed et al. [116] which employs SVM with TF-IDF. They remove rows with words written in the native script. They do not deal with the code-mixed aspect. We, on the other hand, use cross-lingually trained word embeddings in the ensemble framework to deal with code-mixing, and a large set of abusive words in 12 Indic languages to deal with spelling errors. Our method outperforms Sazzed et al. by 5.54%, 3.06%, 8.52%, and 5.85% in accuracy, precision, recall, and F1-score metrics, respectively. On comparison with Bansal et al. [98], our method achieves an increase of 3.43%, 4.1%, 2.61%, and 3.36% in terms of accuracy, precision, recall, and F1-score, respectively. Bansal et al. uses XLM-R with transliterated comments and emoji embeddings.

3.3.2.2 Performance Comparison on MACI Dataset

We compare the results of our method on MACI with the same set of previously mentioned methods. We use post-affinity for MACI instead of the combined user-post affinity feature due to the unavailability of user features in this dataset. On this dataset as well, our method exhibits significantly better performance compared to other methods. This is due to the reason that the post affinity feature has a strong association with abusive content. All the results are shown in Table 3.9. As shown in the table, our method achieves 91.97% accuracy, 90.12% precision, 93.08% recall, and an F1-score of 91.58%. We achieve an absolute performance gain of 10.21%, 4.52%, 19.37%, and 12.37% over the count-based method, and 10.08%, 3.42%, 20.37%, and 12.49% over the frequency-based method in terms of accuracy, precision, recall, and F1-score, respectively. Another baseline, the embedding-based method, which refers to non-contextual word embeddings, is outperformed too by our method on the MACI dataset. Our method obtains 7.54%, 3.07%, 14.31%, and 8.88% higher accuracy, precision, recall, and F1-score, respectively. The transformer-based method Muril is outperformed by 9%, 2.95%, 17.98% and 10.9%, M-Bert by 8.5%, 0.2%, 19.78%, and 10.81%, XLM-R by 6.13%, 4.96%, 8.18%, and 6.56%, in terms of accuracy, precision, recall and F1-score, respectively. Besides these, results are compared with the existing methods Modha et al. [117], Sazzed et al. [116] and Bansal et al. [98]. Our method achieves a performance gain of 10.12%, 5.52%, 18.15%, and 12.11% over Modha et al. in terms of accuracy, precision, recall and F1-score evaluation metrics, respectively. In comparison with Sazzed et al., our method achieves an increase of 9.66%, 3.96%, 18.46%, and 11.6% in accuracy, precision,

3.3 Experimental Evaluations

recall, and F1-score, respectively. Bansal et al. performed their experiments on the same dataset, and our proposed method outperforms their method by 4.27%, 2.42%, 7.28%, and 5.18% in terms of accuracy, precision, recall, and F1-score evaluation metrics, respectively.

Table 3.9: Comparison with classical and state-of-the-art baseline methods on MACI dataset

	Methods	Acc	P	R	F1
Classical Methods	Count-based [118]	81.76	85.60	73.71	79.21
	Frequency-based [119]	81.89	86.70	72.70	79.09
	Embedding-based [120]	84.43	87.04	78.77	82.70
State-of-the-art Baseline Methods	Muril [1]	82.97	87.17	75.09	80.68
	M-Bert [100]	83.47	89.92	73.30	80.77
	XLM-R [101]	85.84	85.15	84.89	85.02
	Modha et al. [117]]	81.85	84.60	74.92	79.47
	Sazzed et al. [116]	82.31	86.16	74.62	79.98
	Bansal et al. [98]	87.70	87.70	85.80	86.40
	Proposed Method	91.97	90.12	93.08	91.58

Our proposed approach has shown significant improvement over other methods on both datasets. Our proposed method makes predictions based on two aspects of the content, i.e., social context features, including user history and post affinity, and textual features. Results presented in Table 3.8 and Table 3.9 show that the utilization of user history and post affinity improves the performance of our proposed method remarkably.

3.3.2.3 Ablation Analysis on SCIDN Dataset

In this section, we discuss and contrast the results utilizing different features with preprocessed transliterated comments while performing the experiments on the SCIDN dataset. Overall, our proposed model achieves an accuracy of 95.40%, precision of 95.18%, recall of 95.59%, and an F1 score of 95.38%, which is 3.35%, 4.09%, 2.46% and 3.28% higher respectively, in comparison to the results taken on raw comments. Figure 3.4 depicts the graph which compares the results for different feature combinations for the SCIDN dataset. For the ablation study, we first take results with raw comments using their contextual embeddings from the transformer-based model, which gives an accuracy of 92.05%, precision of 91.09%, recall of 93.13%, and an F1-score of 92.10%. We then perform the experiments with preprocessed transliterated comments using their contextual embeddings, and here we

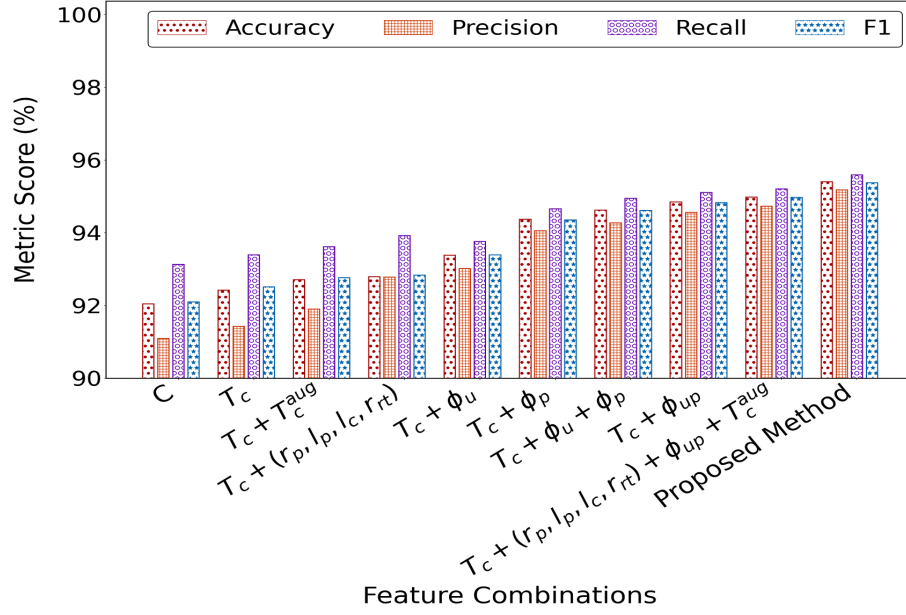


Figure 3.4: Ablation Analysis Results on SCIDN Dataset

obtain 0.37%, 0.34%, 0.26%, and 0.41% better results than the results of raw comments in terms of accuracy, precision, recall, and F1-score, respectively. We then take results after data augmentation, and here also we get a slight improvement in performance as compared to raw comments. Next, along with the contextual embeddings from the transliterated comments, we use the four post features and through them, an improvement of 0.74%, 0.68%, 0.79%, and 0.74% is observed in accuracy, precision, recall, and F1-score evaluation metrics, respectively. It can be seen in the graph that the highest improvement in results, by using an individual feature along with the embeddings, is achieved by combined user-post affinity as compared to the results of raw comments. Here, we get significantly higher accuracy, precision, recall, and F1-score of 94.85%, 94.56%, 95.11%, and 94.83%, respectively. These results are 2.79%, 3.47%, 1.98%, and 2.73% better than the results on raw comments in terms of accuracy, precision, recall, and F1-score, respectively. This performance gain is expected because user-post affinity has the strongest correlation with class labels, as shown in Table 3.2 in Section 3.2.2. Each feature contributes to performance gain when used in combination with the contextual embeddings, thus, finally providing state-of-the-art results as compared to the results for a single modality, i.e., text. Social context features, specifically user and post affinities, are recommended for abusive content detection.

3.3 Experimental Evaluations

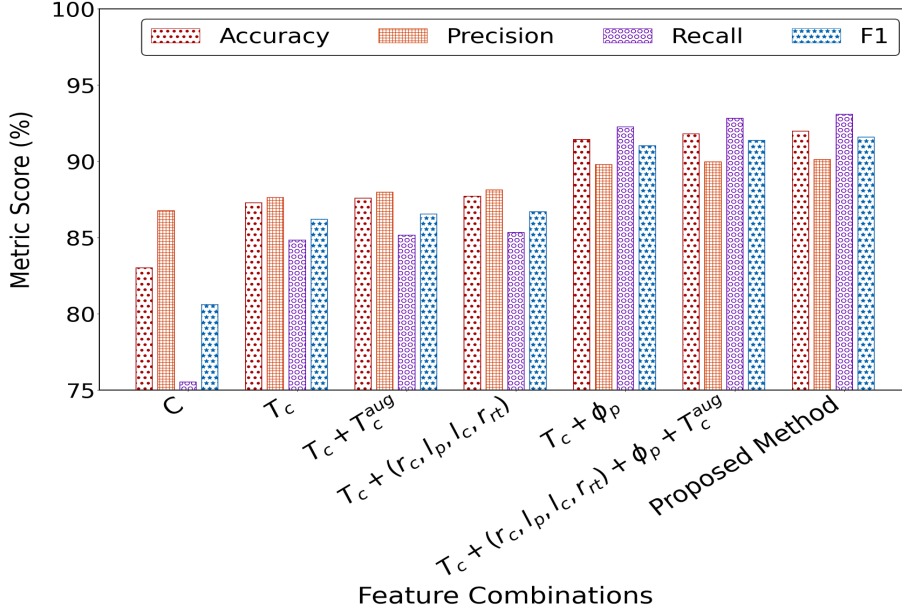


Figure 3.5: Ablation Analysis Results on MACI Dataset

3.3.2.4 Ablation Analysis on MACI Dataset

Figure 3.5 illustrates a graph that compares the results obtained from various feature combinations for the MACI dataset. By incorporating all features, the proposed model achieves an accuracy of 91.97%, precision of 90.12%, recall of 93.08%, and an F1-score of 91.58%, which is 8.96%, 4.37%, 17.56% and 10.98% higher respectively, in comparison to the results for raw comments. For the MACI dataset, post affinity provides the best results for any individual feature used in combination with text embeddings. This is because post affinity has the strongest correlation with class labels as shown in Table 3.3 in Section 3.2.2. Using post affinity, we achieve 91.43%, 89.78%, 92.26%, and 91.01% accuracy, precision, recall and F1-score, respectively. This is an improvement of 8.42%, 3.03%, 16.74%, and 10.41% over raw comments in terms of respective evaluation metrics.

3.3.2.5 Language-wise Performance on SCIDN

To validate the performance of our proposed method across different Indic languages, we conduct a language-wise comparison. The results demonstrate that our method outperforms all the compared methods by significant margins. In our comparisons, we utilize transformer-based multilingual methods as benchmarks.

Results for language-wise comparison corresponding to SCIDN are presented in Ta-

Table 3.10: Language-wise comparison on SCIDN dataset

Language	Muril			XLM-R			M-Bert			Proposed Method		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
As	81.30	88.77	84.87	90.56	88.88	89.72	88.46	85.18	86.79	94.00	95.91	94.94
Bn	89.52	91.59	90.54	88.39	92.27	90.29	89.53	83.85	86.59	92.46	92.58	92.52
En	90.20	93.39	91.77	91.43	93.39	92.40	92.00	92.66	92.33	95.46	95.77	95.61
Gu	89.49	90.70	90.09	87.50	92.98	90.15	86.20	87.71	86.95	93.86	93.43	93.65
Hi	92.44	94.19	93.31	93.11	93.34	93.22	92.73	92.14	92.43	96.39	97.25	96.82
Ka	84.87	88.84	86.81	87.89	85.48	86.62	88.33	80.76	84.38	95.49	88.51	91.87
MI	79.27	90.39	84.47	73.52	75.48	74.49	78.30	51.69	62.28	95.41	89.46	92.34
Mr	86.47	95.14	90.59	84.92	94.66	89.53	84.17	88.67	86.36	94.80	96.80	95.79
Od	90.60	92.16	91.37	91.90	90.32	91.11	92.43	89.54	90.96	95.70	92.53	93.95
Pa	93.16	95.96	94.54	92.57	96.01	94.26	92.21	94.93	93.55	94.30	96.83	95.55
Ta	86.88	85.13	86.00	83.83	82.75	83.29	84.27	67.92	75.22	91.84	89.94	90.88
Te	87.89	94.04	90.86	87.84	92.61	90.16	87.51	89.89	88.69	95.74	93.68	94.70

ble 3.10. Our method outperforms Muril, XLM-R, and M-Bert for Assamese by 10.07%, 5.22%, and 8.15%, respectively, in terms of the F1-score. For the Bengali language, our model achieves better F1-scores as compared to the results of the aforementioned models by 1.98%, 2.23%, and 5.93%, respectively, in terms of the F1-score metric. Corresponding to the same models, we have achieved an improvement through our own model, of 3.84%, 3.21%, and 3.28%, for English. An improvement of 3.56%, 3.50%, and 6.70%, is observed for Gujarati, of 3.51%, 3.6%, and 4.39%, for Hindi, and of 5.06%, 5.25%, and 7.49%, for Kannada, in terms of F1-score. For Malayalam, our model has improved abuse identification by 7.87%, 17.85%, and 30.06%, in comparison to the state-of-the-art models referenced above, by 5.20%, 6.26%, and 9.43%, for Marathi, and by 2.58%, 2.84%, & 2.99%, for Odiya, for the F1-score metric. For Punjabi, the margin of performance gain is 1.01%, 1.29%, and 2%, for Tamil, it is 4.88%, 7.59%, and 15.66%, and for Telugu, it is 3.84%, 4.54%, and 6.01% in terms of F1-score. For precision and recall too, our method outperforms all the compared methods with significant margins as depicted in the table.

3.3.2.6 Language-wise Performance on MACI

In this section, a language-wise performance comparison is presented for the MACI dataset. Comparisons are made with transformer-based methods, Muril, M-Bert, and XLM-R. For the MACI dataset too, our method outperforms all the compared methods for all languages. Table 3.11 shows the results of language-wise comparisons on the MACI dataset. In Assamese, our model beats the performance of Muril, XLM-R, and M-Bert by 13.07%, 15.02%, and 15.6%, respectively, in terms of the F1-score. Our model has achieved a performance gain

3.4 Discussion

of 10.25%, 5.81%, and 9.53% in F1-score for the Bengali language, as compared to the aforementioned models. In Gujarati, our approach outperforms these state-of-the-art models by 13.9%, 6.84%, and 8.25%, in Hindi by 9.91%, 8.76%, and 11.61%, and in Kannada by 14.49%, 7.34%, and 9.32%, respectively, in terms of F1-score. The experimentation we conducted has shown that for Malayalam, our model is better than the aforementioned models by 55.92%, 23.20%, and 62.77%, while for Marathi, it is better than them by 22.56%, 13.44%, and 29.27%, respectively, for the F1-score metric. We achieve an improvement in F1-score of 13.92%, 5.73%, and 12.97%, for Odiya, in comparison to other models and for Tamil and Telugu, we are outperforming these models by 18.23%, 3.44%, and 7.82%, and 7.66%, 6.06%, and 7.22%, respectively, in terms of F1-score. Our model outperforms Muril, XLM-R, and M-Bert in language-wise comparisons pertaining to the precision and recall metrics as well, as depicted in the table.

Table 3.11: Language-wise comparison on MACI dataset

Language	Muril			XLM-R			M-Bert			Proposed Method		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
As	87.65	80.68	84.23	82.75	81.81	82.28	88.15	76.13	81.70	96.78	97.83	97.30
Bn	85.31	86.57	85.94	89.89	90.88	90.38	92.33	81.65	86.66	94.58	97.86	96.19
Gu	77.28	86.18	81.49	89.29	87.82	88.55	88.47	85.85	87.14	93.29	97.59	95.39
Hi	88.35	81.94	85.02	84.76	87.63	86.17	89.58	77.87	83.32	92.57	97.42	94.93
Ka	89.05	75.62	81.78	87.42	90.49	88.93	87.13	86.77	86.95	95.58	96.96	96.27
Ml	71.49	23.79	35.70	72.80	64.54	68.42	72.04	18.04	28.85	92.47	90.78	91.62
Mr	76.69	54.01	63.38	77.26	68.82	72.80	85.08	42.82	56.97	87.60	84.92	86.24
Od	82.88	72.06	77.09	82.96	87.72	85.28	84.03	72.84	78.04	89.47	92.61	91.01
Ta	93.67	61.63	74.34	91.03	87.31	89.13	91.64	78.83	84.75	90.05	95.24	92.57
Te	87.38	81.22	84.19	86.93	84.76	85.79	93.21	77.50	84.63	88.89	95.01	91.85

3.4 Discussion

This article introduces a technique for abusive content detection on social media. The proposed method is specifically built for multiple low-resource Indic languages. Our method integrates social context features with textual features using the late fusion technique, where two categories of features are fed to two separate networks before concatenating them to get the joint feature for final classification. Overall comparison results show that the proposed method outperforms the classical as well as state-of-the-art methods by a significant margin.

3.4.1 Limitations

One of the limitations of the proposed method is that it requires social context features; therefore, in the absence of those, the proposed method may not be as effective as it is with the social context features. The proposed method has a further limitation in that it may not always recognize an abusive comment if the spelling of a non-abusive word matches that of an abusive word. Similarly, as India is a multilingual country with rich diverse cultures, there may be a few words that are abusive in one language but non-abusive in another. This limitation occurs when the method fails to capture the context of the words in the comment accurately. Ensuring the model’s resilience to such errors needs additional work beyond the scope of our work.

3.5 Conclusion

This study examines the challenges of abusive content detection in multiple Indic languages and proposes a novel method to detect such content. The study indicates that social context features and textual features, when trained in dedicated modules before integration help identify abusive content. The study finds that careful data augmentation can be an effective approach to address spelling inconsistencies that arise from the use of Roman script in multilingual social media content. Further, this study suggests that cross-lingually trained text embeddings can be useful in addressing the challenge of code-mixed content in abusive content detection. We show, with extensive testing on two datasets, that our proposed method outperforms the baseline and state-of-the-art methods. Our model achieves better performance across all the evaluation metrics as established by the experimental results. Future work involves addressing the previously stated limitations, such as context identification of similar words. Additionally, we plan to examine how various factors, such as geographic location and time window, influence abusive content detection. Addressing them may further improve the model’s performance.

Chapter 4

Multilingual Hostile Content Explanation

4.1 Introduction

The rise in online media usage has heightened exposure to hate speech, making detection systems increasingly essential. While initial research focused only on the detection of hateful content [121], more recent efforts have shifted towards explainability and addressing the rationales behind labelling content as hateful [43]. Advanced methods such as Large Language Models (LLMs) and their predecessors, such as BERT [100], can provide explanations for such decisions. LLMs can be prompted to provide the explanations in the form of an identified hateful list of words [42] for a given sentence, whereas methods such as BERT can provide explanations in the form of attention given to each word of the sentence [10]. However, in the case of under-resourced languages, where overall sequence classification performance has improved significantly over time [8], machine-provided rationales, many a time, do not align well with human rationales. For instance, in Figure 4.1, the translation of the first and third rationales provided by the human annotator would be “to feel offended” and “to bark”, respectively, which are very offensive terms in the Hindi language. Though the LLMs correctly identify the explicit term “dog”, they do not identify first and third rationale terms as hateful or offensive, which can be attributed to the fact that societal and cultural aspects of languages are often overlooked by these methods for under-resourced languages. This gap between machine-provided rationales and human rationales can be bridged by providing rationale resources for these languages.

4.1 Introduction

	Rationales
Human	जिसे मिर्ची लगे वो भी कमेंट करे कुत्तों को भोकने की इजाजत है
GPT-3.5	जिसे मिर्ची लगे वो भी कमेंट करे कुत्तों को भोकने की इजाजत है
Llama-3.1	जिसे मिर्ची लगे वो भी कमेंट करे कुत्तों को भोकने की इजाजत है
Mistral-24.07	जिसे मिर्ची लगे वो भी कमेंट करे कुत्तों को भोकने की इजाजत है
Translation	Whoever feels offended, let them comment, dogs are allowed to bark.

Figure 4.1: Example of Human and LLM Rationales*

4.1.1 Motivation: Scarcity of Hateful Rationales

Though numerous rationale and reasoning-based resources for the English language [122, 123] are compiled in ERASER [124] in various domains spanning from movie reviews [125], questions with multiple correct answers [126], and to other miscellaneous domains [127, 128], there is a scarcity of rationale-based resources in the domain of hate speech detection. In 2021, Mathew et al. [10] introduced the HateXplain dataset, which features word and phrase-level span annotations for hate speech in English, claiming to be the first of its kind. However, similar datasets for under-resourced languages such as Hindi and Telugu are nonexistent to the best of our knowledge. Even for English, HateXplain is one of the few datasets offering human rationales for hate speech detection. To address this gap, we provide word-level human rationales for hate speech detection datasets in Telugu, Hindi, and English. Our primary focus was on Indic languages. However, we believe that evaluating models on the English dataset alongside HateXplain enhances the generalizability of future approaches.

4.1.2 A Glance at the Proposed Solution

We propose a novel explainability framework that combines LLM-based and n-gram-based explanations to generate final explanations. We propose an **eXplainable Multilingual haTe Speech deTection** (X-MuTeST) method. X-MuTeST is a two-stage explainable hate speech detection framework designed to enhance both classification performance and explainability. It incorporates a novel explainability method based on n-grams to identify relevant to-

*Offensive content warning: The presented example is just for illustration. We do not promote the views presented in the image.

kens. In the first stage, the model’s attention is guided by human-annotated rationales to align the model’s focus with key tokens identified by humans. In the second stage, training is guided by the n-gram-based explainability method, which generates attention masks based on model-driven token importance. This approach balances human rationales and the model’s insights, improving both explainability and classification performance. Finally, union of explanations is taken from LLama 3.1 and X-MuTeST. Through experiments on three datasets, Hindi, Telugu, and English, we demonstrate the efficacy of the proposed framework.

The key contributions of this chapter are as follows:

- We contribute benchmark human-annotated rationales for under-resourced languages, such as Hindi and Telugu, alongside English. This will provide a valuable resource for future research in explainable hate speech detection across multiple languages.
- We propose a novel LLM-consulted and n-gram-based hybrid explainability method that combines LLMs with traditional transformer-based attention improvement. The proposed method enhances explainability performance.
- We propose a novel explainable hate speech detection framework with a two-stage training method that balances classification and explainability performance.

4.2 Dataset Description and Annotation

4.2.1 Dataset Source

The Hindi and English datasets utilized in this research are derived from the Hate Speech and Offensive Content Identification (HASOC) contest, which is dedicated to the detection of hate speech and offensive language across various languages and dialects. We collect datasets from HASOC 2020 [129,130] and HASOC 2021 [131] for Hindi and English, which include samples sourced from Twitter featuring annotated instances of hate speech and offensive content. We mix datasets of two years to increase the number of samples. We select binary labels that categorize the sentences into *HATE* and *NOT HATE* categories. Subsequently, the testing set is split, where 15% of the samples are retained for testing for both

4.2 Dataset Description and Annotation

Table 4.1: Summary of hate and non-hate records

Metric	Telugu	Hindi	English
Total Records	4,492	6,004	6,334
Hate Records	2,131	2,026	3,767
Non-Hate Records	2,361	3,978	2,567
Avg Word Count	6.46	18.84	19.20
Avg Character Count	68.98	110.93	100.47
Avg Rationale Length	3.10	3.26	3.78

languages. Dataset details are shown in Table 4.1.

The Telugu dataset is collected from the Hate and Offensive Language Detection in Telugu Codemixed Text (HOLD-Telugu) task [132, 133] organized as part of Dravidian-LangTech 2024 [134]. The comments in the dataset are collected from YouTube and annotated for binary class labels, *HATE* and *NON-HATE*. The original dataset contains text in Latin script; we transliterate the text into Telugu script using the IndicXlit method [135]. A testing set is provided in the dataset. Dataset details are shown in Table 4.1.

4.2.1.1 Profile of Annotators

To ensure the reliability and consistency of the data annotation process, we employed a rigorous selection and training procedure for our annotators. Five annotators were hired based on their linguistic proficiency, subject matter expertise, and prior experience in annotation tasks. Annotators were selected such that we had three annotators for each language. All annotators underwent a structured training phase to familiarize themselves with the annotation guidelines. Periodic quality checks were conducted, and discrepancies were resolved through consensus discussions [22]. Given the nature of our dataset, we carefully select annotators above the age of 18 from diverse regions across the country to mitigate potential demographic biases.

4.2.2 Rationale Annotation Procedure

To ensure the quality and accuracy of annotations, we engage native language experts fluent in Hindi, English, and Telugu. This is crucial for preserving the societal and cultural aspects within the dataset. Each post is annotated by three annotators. Annotators are tasked with identifying token-level rationales that explain why a particular text is classified as hateful or

non-hateful. For instance, in a sentence marked as hateful, the rationale could be an offensive word or phrase that justifies the hate speech classification. Human annotators provide these rationales by marking hate-indicating tokens with a label of 1 and non-hate tokens with 0, enhancing both model interpretability and explainability. In non-hate samples, all tokens are marked as 0. During the initial annotation rounds, we observe *moderate to substantial agreement* (61-80%). Since our dataset is intended to support explainability in downstream models, maintaining high annotation consistency is crucial. Although annotators are initially provided with detailed guidelines, the agreement scores indicate the need for further refinement. As supported in prior works [136, 137], iterative annotation can significantly improve label quality and annotator alignment. To address this, we employ an iterative annotation process, as shown in Figure 4.2, conducting additional guidance sessions and having annotators re-annotate the dataset, which increased the agreement among annotators. As a token of appreciation and to ensure fair compensation, annotators are provided with free access to our GPU for 150 hours, valid for one year. Each dataset is annotated by three annotators for rationales. Table 4.2 shows the inter-annotator agreement scores using Cohen’s Kappa method between any two annotators, whereas overall agreement is computed using Fleiss method. From Table 4.2, all three languages show substantial agreement. English exhibits the highest consistency (85.10), while Hindi shows slightly lower pairwise scores. These results confirm that iterative refinement notably improved annotation reliability across languages.

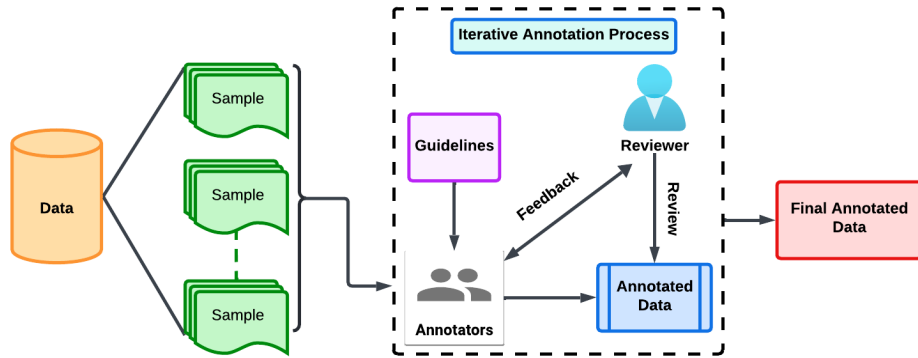


Figure 4.2: Iterative Annotation Procedure

Table 4.2: Kappa agreement scores

Language	A1 & A2	A2 & A3	A3 & A1	Overall
Telugu	82.50	87.75	88.25	81.00
Hindi	86.45	88.00	79.20	83.15
English	87.60	82.00	89.30	85.10

4.3 Methodology

We propose **X-MuTeST**, an explainable hate speech detection framework that leverages both human rationales and a novel explainability formulation to train a model. Training proceeds in two stages as shown in Figure 4.3. In Stage-1, the model is trained for three epochs with human rationales guiding its *attention*. In Stage-2, the target attention mask is derived from our explainability method, allowing the model to balance human annotations with tokens deemed salient by the model itself. This dual approach enhances both classification accuracy and interoperability. For the final, explainability, we take the union of explanations provided by LLama-3.1 and our attention-based method. We integrate LLMs, as they have shown promise in semantic and contextual reasoning [3]. Specifically, LLaMA-3.1 is selected based on empirical evidence, as it demonstrated a higher token-level alignment with human rationales than other LLMs, for Telugu and Hindi.

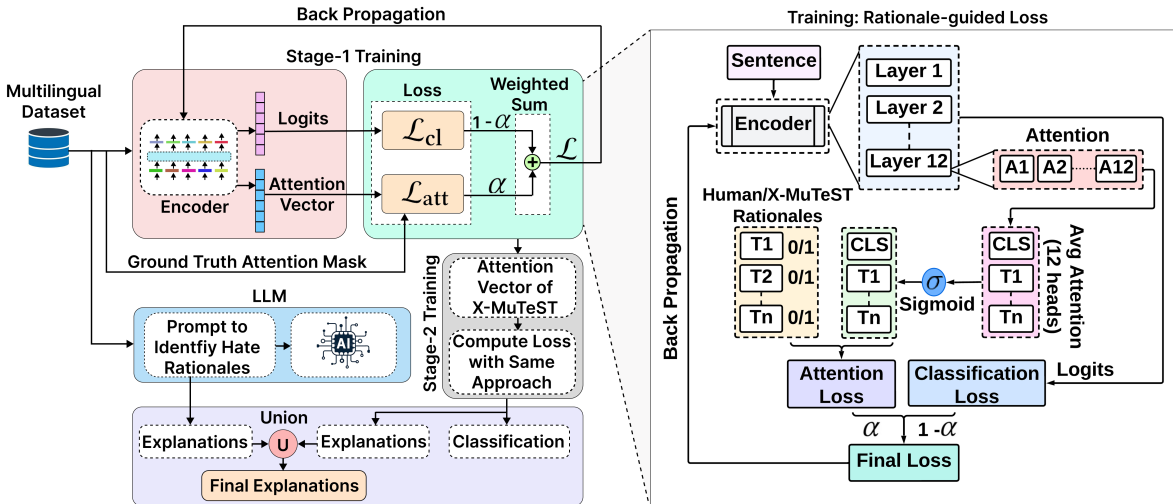


Figure 4.3: Architecture of LLM-consulted Explanation Framework

For each language, we select the encoder that provides the best classification performance after finetuning. Hence, for X-MuTeST framework, we employ Muril [99] for Telugu and

English and XLMR [101] for Hindi.

4.3.1 Training Procedure

In Stage-1, the model’s attention is guided by human rationales. Let a_i denote the normalized attention score assigned to the i th token (word) by the model and R_i the corresponding binary human rationale. The attention weights are computed using the dot product between token representations and the $[CLS]$ token, followed by softmax normalization:

$$a_i = \frac{\exp(h_i^\top h_{[CLS]})}{\sum_{j=1}^L \exp(h_j^\top h_{[CLS]})} \quad (4.1)$$

where h_i and $h_{[CLS]}$ are the hidden states of the i th token and the $[CLS]$ token respectively.

The attention alignment loss is computed using cross-entropy:

$$\mathcal{L}_{att} = - \sum_{i=1}^L R_i \log a_i \quad (4.2)$$

The overall loss function combines this attention loss with the standard classification loss $\mathcal{L}_{cl}$ as follows:

$$\mathcal{L}_1 = \alpha \mathcal{L}_{att} + (1 - \alpha) \mathcal{L}_{cl} \quad (4.3)$$

where α is a balancing coefficient (set to 0.3 in Stage-1). This helps the model to jointly learn task-specific classification while aligning its attention with human-provided rationale supervision.

In Stage-2, the human rationales are replaced by model-generated attention targets derived from our explainability method. For a sequence S of length L , we identify the top k tokens based on their explainability scores, where

$$k = \begin{cases} 5, & \text{if } L \geq 10 \\ \lceil L/2 \rceil, & \text{otherwise} \end{cases} \quad (4.4)$$

The loss function in this stage remains the same as in Equation (4.3), but with updated α

values: 0.6 for Telugu and English, and 0.7 for Hindi. These values are selected through empirical tuning for best validation performance.

4.3.2 Explainability via N-gram Contributions

Our explainability method quantifies the importance of each token based on its contributions across overlapping n-grams. Let the model’s output logits for the original sequence S be:

$$P_{\text{orig}} = f(S) \quad (4.5)$$

For each n-gram ng ($n = 1, 2, 3$), we compute:

$$P_{ng} = f(ng) \quad (4.6)$$

and define the logit drop as:

$$\Delta P_{ng} = |P_{\text{orig}} - P_{ng}| \quad (4.7)$$

The unnormalized importance score for a token t is then computed as:

$$E[t] = \sum_{n=1}^3 w_n \cdot \frac{1}{N_t^{(n)}} \sum_{ng \ni t} \Delta P_{ng} \quad (4.8)$$

where $w_1 = 0.5$, $w_2 = 0.3$, and $w_3 = 0.2$ are the weights assigned to unigrams, bigrams, and trigrams respectively, and $N_t^{(n)}$ is the number of n-grams of length n containing token t .

Finally, the scores are normalized:

$$\tilde{E}[t] = \frac{E[t]}{\sum_j E[j]} \quad (4.9)$$

This formulation captures both token-level and contextual relevance, providing more comprehensive guidance to the model in Stage-2.

4.3.3 Final Explanations through LLM Consultation

As shown in Figure 4.3, the final explanation is derived by combining rationales from two sources: our in-model n-gram based method and LLM-generated rationales. Let $\mathcal{E}_X$ denote the set of top tokens selected by the X-MuTeST explainability scores, and $\mathcal{E}_{LLM}$ the rationales returned by LLaMa-3.1. The final explanation set is given by:

$$\mathcal{E}_{\text{final}} = \mathcal{E}_X \cup \mathcal{E}_{LLM} \quad (4.10)$$

To assess alignment between the two explanation sources, we also compute an agreement score:

$$\text{Agreement} = \frac{|\mathcal{E}_X \cap \mathcal{E}_{LLM}|}{|\mathcal{E}_X \cup \mathcal{E}_{LLM}|} \quad (4.11)$$

This union-based approach ensures both syntactic and semantic coverage, combining task-specific saliency with LLM-derived contextual reasoning. LLaMa-3.1 is used for LLM consultation due to its superior interpretability, as discussed in the results section.

4.4 Experimental Setup and Results

4.4.1 Metrics for Evaluation

4.4.1.1 Performance-Based Metrics

We use standard performance-based metrics, including accuracy, F1 score, and macro-F1, to evaluate the classification performance.

4.4.1.2 Explainability-Based Metrics

We assess explainability using two metrics: *plausibility* (how believable the explanation is) and *faithfulness* (how well it reflects the model’s decision process) [10, 138].

Plausibility

Plausibility is measured via token-level F1-score and Intersection-over-Union (IOU) F1-score. IOU is computed as the ratio of overlapping tokens to the union of predicted and ground truth rationales, with a match defined when the overlap exceeds 0.5 [10, 124].

Faithfulness

Faithfulness is evaluated using *comprehensiveness* and *sufficiency* [10, 124].

Comprehensiveness quantifies the reliance on rationales by measuring the drop in prediction confidence when they are removed:

$$\text{Comprehensiveness} = P(x_i)_c - P(x_i \setminus r_i)_c \quad (4.12)$$

where $P(x_i)_c$ is the probability for class c using the full input x_i , and $P(x_i \setminus r_i)_c$ is the probability when the predicted rationale r_i is masked.

Sufficiency evaluates whether the rationale alone supports the prediction:

$$\text{Sufficiency} = P(x_i)_c - P(r_i)_c \quad (4.13)$$

with $P(r_i)_c$ computed from the rationale r_i only. Lower sufficiency indicates that the rationale is informative.

For both metrics, tokens are replaced with the $\langle MASK \rangle$ token, and the top-5 tokens are used as the predicted rationales.

4.4.2 Hyperparameter and System Details

Table 4.3 shows the hyperparameters used for experiments. Values in bold were taken as final hyperparameters.

All the experiments were performed on a DGX A100 server with 1000 GB RAM, 40 GB GPU, and an AMD EPYC 7742 64-Core Processor.

Hyperparameter	Value
Learning rate	(10^{-3} , 10^{-4} , 10^{-5})
Batch size	(16, 32 , 64, 128)
Epochs	(10, 20, 30, 50 , 100)

Table 4.3: Hyperparameters for 2nd-stage training in X-MuTeST

4.4.3 Comparison Methods

We employ different multilingual transformer encoders for classification, whereas LIME and X-MuTeST methods are used for explainability. Additionally, we employ three LLMs, GPT-4o [139], LLaMa-3.1 [140], and Mistral-24.07 [141], using the prompt-based zero-shot method. We use token-F1 and IOU-F1 to evaluate the explainability of LLMs. As prompt-based methods do not provide probability scores for their decision, comprehensiveness and sufficiency scores could not be computed for LLMs. We also compared our method with BERT-HateXplain-LIME [10] and with BERT-ILP method [142]. We evaluate transformer encoders as follows:

We employ two multilingual encoders Muril [99] and XLMR [101] for comparison. First, the base encoder model is finetuned for the sequence classification objective, and the LIME and the X-MuTeST approaches are used to evaluate explainability performance after training. Next, the base encoder is trained by integrating human rationales, and subsequently, LIME and the X-MuTeST method are employed for explainability.

4.4.4 Results Analysis for Telugu

Table 4.4 presents the classification and explanation performance of various models on the Telugu dataset. The LLM-consulted X-MuTeST outperforms all other models across metrics. It achieves the highest accuracy (0.8881), F1 score (0.8762), and Macro-F1 (0.8849), surpassing the next-best baseline, Muril-RX, by 1.41% in accuracy and 1.29% in Macro-F1. These results highlight the effectiveness of integrating LLM-consulted explanations combined with X-MuTeST.

On explainability metrics, X-MuTeST yields the best plausibility performance, with token-F1 of 0.6231 and IOU-F1 of 0.3189, improvements of 4.77% and 2.76%, respectively, over Muril-RX. This indicates stronger alignment of the generated rationales with human an-

4.4 Experimental Setup and Results

Table 4.4: Classification and explainability performance of various methods on Telugu

Model	Performance			Plausibility		Faithfulness	
	Acc $\uparrow$	F1 $\uparrow$	Macro-F1 $\uparrow$	Token-F1 $\uparrow$	IOU-F1 $\uparrow$	Comp $\uparrow$	Suff $\downarrow$
XLMR-LIME	0.7927	0.8030	0.7921	0.5143	0.2232	0.6470	0.3094
XLMR-XMuTeST	0.7927	0.8030	0.7921	0.5256	0.2261	0.6539	0.2568
XLMR-Rationale-LIME	0.8455	0.8355	0.8450	0.5492	0.2385	0.7395	0.1380
XLMR-Rationale-XMuTeST	0.8455	0.8355	0.8450	0.5634	0.2417	0.7267	0.1035
Muril-LIME	0.8598	0.8595	0.8591	0.5515	0.2840	0.6676	0.1287
Muril-XMuTeST	0.8598	0.8595	0.8591	0.5583	0.2919	0.6852	0.0947
Muril-Rationale-LIME	0.8740	0.8640	0.8733	0.5608	0.2941	0.6968	0.0802
Muril-Rationale-XMuTeST	0.8740	0.8640	0.8733	0.5754	0.3013	0.7243	0.0747
GPT-4o	0.6484	0.5929	0.6417	0.3893	0.1815	-	-
Llama-3.1	0.6362	0.6716	0.6319	0.5154	0.2806	-	-
Mistral-24.07	0.6402	0.4451	0.5895	0.3453	0.1228	-	-
BERT-HateXplain-LIME	0.8089	0.8075	0.8084	0.5560	0.2233	0.6851	0.0516
BERT-ILP	0.8213	0.8183	0.8221	0.5236	0.2440	0.5833	0.0849
X-MuTeST with LLM	0.8881	0.8762	0.8849	0.6231	0.3189	0.7456	0.0448

notations. In terms of faithfulness, X-MuTeST obtains a lower (better) sufficiency score of 0.0448 compared to 0.0747 for Muril-RX, suggesting that its identified rationales are more essential to the model’s prediction. LLMs such as GPT-4o, Llama-3.1, and Mistral-24.07 show lower performance in both classification and explainability, likely due to inaccurate transliteration from Latin to Telugu script.

4.4.5 Results Analysis for Hindi

Table 4.5 shows that X-MuTeST with LLM achieves the highest classification and explanation scores for Hindi, with an accuracy of 0.8745 and token-F1 of 0.4344. Compared to the next-best model (Muril-RX), it gains 3.88% in accuracy and 2.27% in token-F1. It also improves sufficiency (0.4768 vs. 0.5042), showing more faithful explanations. While LLMs like Llama-3.1 show high token-F1 (0.4263), their classification drops sharply by 35.73%, indicating inaccurate token lists. XLMR and BERT-based baselines also trail across all metrics. These results confirm that the LLM-guided two-stage training in X-MuTeST produces more accurate and human-aligned explanations.

Table 4.5: Classification and explainability performance of various methods on Hindi

Model	Performance			Plausibility		Faithfulness	
	Acc $\uparrow$	F1 $\uparrow$	Macro-F1 $\uparrow$	Token-F1 $\uparrow$	IOU-F1 $\uparrow$	Comp $\uparrow$	Suff $\downarrow$
XLMR-LIME	0.8535	0.7988	0.8418	0.3068	0.1494	0.6651	0.7296
XLMR-XMuTeST	0.8535	0.7988	0.8418	0.3948	0.1799	0.7122	0.6528
XLMR-Rationale-LIME	0.8590	0.8000	0.8456	0.3127	0.1564	0.7011	0.5314
XLMR-Rationale-XMuTeST	0.8590	0.8000	0.8456	0.4007	0.1832	0.7261	0.5612
Muril-LIME	0.8269	0.7679	0.8149	0.3124	0.1588	0.5585	0.5517
Muril-XMuTeST	0.8269	0.7679	0.8149	0.4182	0.1734	0.5486	0.5615
Muril-Rationale-LIME	0.8357	0.7716	0.8217	0.3160	0.1724	0.6890	0.4936
Muril-Rationale-XMuTeST	0.8357	0.7716	0.8217	0.4117	0.1906	0.6945	0.5042
GPT-4o	0.5461	0.6071	0.5542	0.3942	0.2266	-	-
Llama-3.1	0.5172	0.5829	0.5049	0.4263	0.3063	-	-
Mistral-24.07	0.7159	0.6684	0.7099	0.3975	0.2832	-	-
BERT-HateXplain-LIME	0.8557	0.7884	0.8376	0.2822	0.1470	0.6481	0.2111
BERT-ILP	0.8395	0.7918	0.8302	0.3941	0.2618	0.6299	0.2704
X-MuTeST with LLM	0.8745	0.8168	0.8623	0.4344	0.2667	0.7483	0.4768

4.4.6 Results Analysis for English

Consistent with the findings for Telugu and Hindi, X-MuTeST achieves the highest classification performance in English as shown in Table 4.6. In terms of explainability, X-MuTeST leads with a token-F1 score of 0.5125 and demonstrates strong faithfulness with a high comprehensiveness of 0.8433 and the lowest sufficiency score of 0.0507. LLMs show moderate classification performance, with GPT-4o scoring relatively high in token-F1 of 0.4613 but lower in overall classification accuracy (0.7802). Mistral-24.07 achieves the highest IOU-F1 of 0.3176, though its overall performance lags behind X-MuTeST. Overall, X-MuTeST consistently outperforms other models in classification while maintaining competitive scores in explainability for the English dataset, too.

4.4.7 Parameter Sensitivity Analysis & Ablation

The graphs in Figure 4.4 show the influence of α on the classification performance in different languages. The α parameter denotes the weights given to attention loss, whereas $(1-\alpha)$ weights are given to classification loss. The attention loss is computed between the attention mask generated through the X-MuTeST explainability method and the predicted average attention score for each token relative to the $[CLS]$ token.

For Hindi and English, the best results are achieved when α is set to 0.6, denoting that

4.4 Experimental Setup and Results

Table 4.6: Classification and explainability performance of various methods on English

Model	Performance			Plausibility		Faithfulness	
	Acc $\uparrow$	F1 $\uparrow$	Macro-F1 $\uparrow$	Token-F1 $\uparrow$	IOU-F1 $\uparrow$	Comp $\uparrow$	Suff $\downarrow$
XLMR-LIME	0.8223	0.8609	0.8075	0.3886	0.2279	0.6500	0.1244
XLMR-XMuTeST	0.8223	0.8609	0.8075	0.4087	0.2195	0.6842	0.0945
XLMR-Rationale-LIME	0.8349	0.8673	0.8245	0.4665	0.2389	0.7449	0.0559
XLMR-Rationale-XMuTeST	0.8349	0.8673	0.8245	0.4752	0.2357	0.7682	0.0546
Muril-LIME	0.8307	0.8694	0.8144	0.3908	0.2002	0.7365	0.0898
Muril-XMuTeST	0.8307	0.8694	0.8144	0.3928	0.2184	0.7527	0.0871
Muril-Rationale-LIME	0.8433	0.8744	0.8385	0.4553	0.2121	0.7992	0.0856
Muril-Rationale-XMuTeST	0.8433	0.8744	0.8385	0.4743	0.2195	0.8372	0.0795
GPT-4o	0.7802	0.8266	0.7633	0.4613	0.3097	-	-
Llama-3.1	0.7560	0.8086	0.7362	0.3995	0.2819	-	-
Mistral-24.07	0.7918	0.8448	0.7643	0.3934	0.3176	-	-
BERT-HateXplain-LIME	0.8339	0.8647	0.8247	0.4454	0.2382	0.8555	0.2019
BERT-ILP	0.8145	0.7967	0.8028	0.4387	0.2291	0.7824	0.2008
X-MuTeST with LLM	0.8604	0.8827	0.8513	0.5125	0.3086	0.8433	0.0507

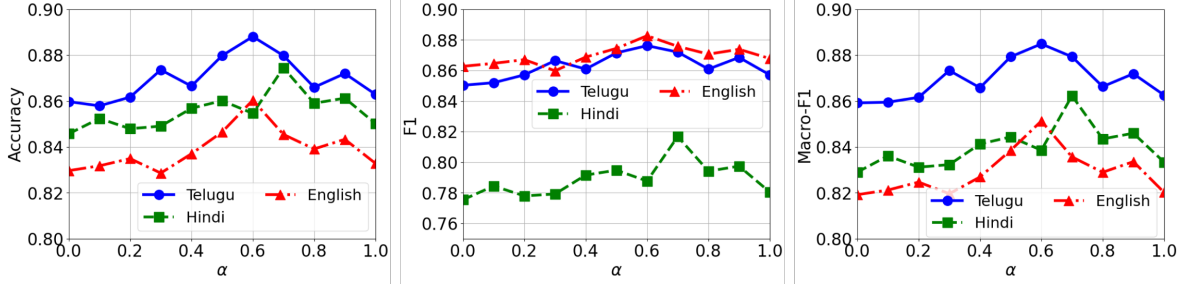


Figure 4.4: Influence of α Parameter on Classification Performance in Stage-2 Training

the X-MuTeST’s attention mask aligns well with that of the model. Lower performance is noted on the boundary conditions when either the entire weight is given to attention loss or the classification loss, denoting that integration of the X-MuTeST explainability method strikes a balance, resulting in enhanced performance. The best performance for Telugu is noted when α is set to 0.7. Additionally, Figure 4.5 shows the ablation study results on two stages of training. Performance degradation is observed on the removal of any of the two stages of training as depicted in the graph. This further demonstrates that the proposed training method of X-MuTeST enhances the classification performance while improving the explainability.

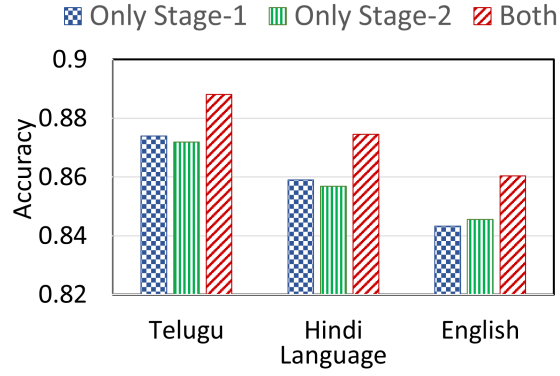


Figure 4.5: Ablation on Two Stages of Training of X-MuTeST

4.4.8 Generalizability

To assess the explanation generalizability of the proposed framework, we conducted additional experiments on two benchmark datasets: HateXplain [10] and HateBRXplain [143]. On the HateXplain dataset, X-MuTeST achieved an IOU-F1 of 0.314 and a Token-F1 of 0.587, surpassing the best reported results in this work by 9.2% and 8.1%, respectively. It also yielded a 5.6% improvement in comprehensiveness. Similarly, on the HateBRXplain dataset, X-MuTeST with LLM recorded a 6.9% gain in Token-F1 and a 2.3% improvement in sufficiency over the strongest baseline. These results demonstrate that the proposed framework generalizes effectively across diverse datasets and settings.

4.5 Conclusion

In this study, we present rationale resources and the LLM-consulted X-MuTeST framework, which demonstrates robust classification and explainability performance across three languages, Hindi, Telugu, and English, for the task of hate speech detection. By incorporating a combination of traditional classification metrics and explainability measures, the framework not only improved classification accuracy but also provided deeper insights into model behaviour. The results suggest that two-stage training and integration of explainability into multilingual text classification enhances both model trustworthiness and user understanding. Further investigations could involve expanding the framework to handle more languages.

Chapter 5

Multimodal Hostile Content Detection

5.1 Introduction

Building upon the discussion of multilingual hostile content detection in the previous chapters, this chapter shifts focus to the multimodal dimension of online hostility against women. Social media has become an indispensable aspect of our daily existence, offering a platform for open communication and idea-sharing. Despite its positive uses, the prevalence of offensive content is a notable concern [144, 145]. Content is considered offensive if it includes any form of unacceptable language, such as insults, threats, or foul language. Such offensive content is frequently directed at individuals or particular societal groups [146], and there is a noticeable pattern where a substantial portion is aimed at women [2]. Social media content that displays hatred, contempt, or prejudice against women is referred to as misogynistic content, and content that discriminates against women based on their gender is identified as sexist [147]. Such content can manifest in various forms, including texts, images, and videos. Since women form a large social media user base, it is crucial to identify misogynistic content on social media as it helps in addressing online gender-based harassment and fostering a safer and more inclusive digital environment.

5.1.1 Women on Social Media: A Statistical View

The increasing number of users on social media platforms signifies the growing influence and widespread adoption of digital communication channels in today's interconnected world

5.1 Introduction

[148]. As of October 2023, social media has infiltrated nearly 61.4% of the global population with nearly 4.95 billion users. Out of all internet users worldwide, 93.5% use social media. There has been a 4.5% increase in social media users within a single year, accounting for almost 215 million new users [149]. The gender distribution on popular social media platforms like Snapchat, Instagram, and Facebook varies slightly, with Snapchat having the highest percentage of female users at 51.00%, Instagram has around 48.20% female users, and Facebook has 43.70% [150]. These statistics indicate that women actively participate in online communication on social media; therefore, it is imperative to find solutions to safeguard women from online harassment by filtering out misogynistic and sexist content.

5.1.2 Existing Approaches and Challenges

Examining and tackling the growing prevalence of online misogyny and sexism is crucial. Multiple potential solutions exist for identifying misogyny and sexism in the literature. Early approaches to misogyny detection relied on unimodal feature engineering such as N-grams [151], Bag of Words [152], Term Frequency-Inverse Document Frequency [153], and the utilization of word embeddings [154] for text analysis. Existing works have also explored linguistic and syntactic handcrafted features such as part of speech, sentence length, sentiment, and hate lexicons to flag misogyny [151, 154]. Although these methods were foundational, recent advancements have shifted towards more sophisticated techniques, such as transformer-based methods [155], multimodal analysis [156], and larger and diverse datasets [157], allowing for more refined and context-aware misogyny detection across a broader spectrum of content.

A significant volume of misogynistic content is disseminated through multimodal memes that combine images and text. This presents a challenge as images that might appear misogynistic in one context may not be interpreted as such in another. Figure 5.1 illustrates an instance where the interpretation of an image varies when combined with different contexts. Figure 5.1(a) and Figure 5.1(b) have the same image but different texts, giving them different class labels. Likewise, identical text can have different labels when paired with different images. Figure 5.1(c) and Figure 5.1(d) have identical text but with different images, which changes the class label in both memes. Misogynistic content can also be disguised in sarcasm or irony, making detection a complex task. Therefore, for the aforementioned reasons,



Figure 5.1: Example of Misogynistic and Non-misogynistic Content

comprehending both modalities through context-aware cross-modal interaction while also concentrating on discriminative information within individual modalities is crucial in misogyny detection. Additionally, it can adapt to new slang or lexicons, further complicating identification. Our proposed approach revolves around tackling these challenges.

5.1.3 Research Objectives

As mentioned in the previous section, numerous challenges are present in the domain of multimodal content analysis, underscoring the complex task of effective integration of diverse modalities such as text and images. Leveraging our prior discourse on these challenges, the primary objectives of this research are as follows:

- To explore a novel multimodal context-aware cross-attention mechanism for offensive content detection against women.
- Exploring the influence of feature space test-time augmentation on the generalizability of predictions for multimodal content.

**Offensive content warning: The presented image is just for illustration. We do not promote the views presented in the image.

- Modeling a unified framework that learns different features through dedicated modules and exploring its effectiveness for multimodal misogyny detection.

5.1.4 A Glimpse into the Proposed Solution Framework

Reflecting on the aforementioned objectives and challenges, we propose a multimodal context-aware attention-based approach for misogyny and sexism detection. The proposed approach employs a multi-faceted strategy where a graph neural network is used for unimodal feature reconstruction to focus on discriminative information in each modality, and a context-aware attention module provides multimodal features. Our approach also leverages a comprehensive set of additional features that include content-specific text and image features. Text features include misogyny-specific lexicon score (MSL) and toxicity features, whereas image features include image caption embeddings and NSFW features. We employ test time augmentation (TTA) to help the model generalize efficiently to variations in the test data that may not have been present in the training data. To our knowledge, none of the existing studies harness the collective advantages of these techniques for detecting offensive content targeting women. In essence, our approach capitalizes on the synergy of these features that help improve the model’s performance by deriving valuable insights from multimodal content.

5.1.5 Key Contributions

The primary contributions of this research study are as follows:

1. Our proposed multimodal misogyny detection approach explores the fusion of features through dedicated modules where multimodal features are learned through a novel multimodal context-aware cross-attention module, unimodal features are refined through a graph-based module, and another module learns the content-specific features.
2. The proposed context-aware cross-attention effectively emphasizes the interaction between crucial regions of an image and words of the text through the integration of the global context of the image.

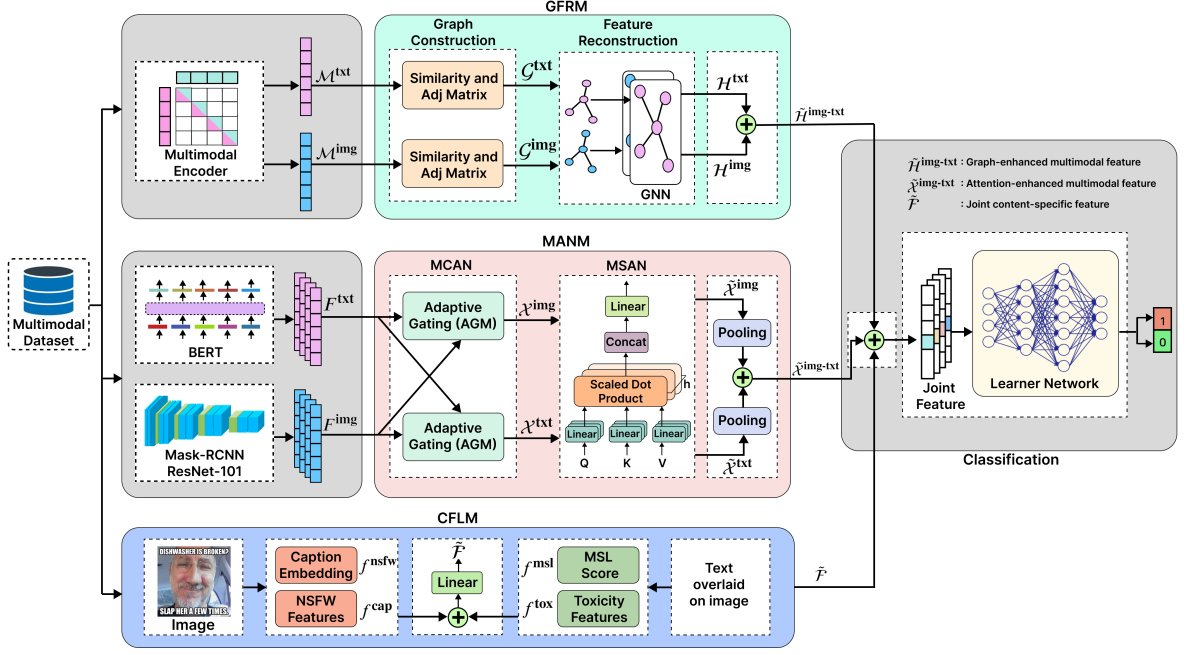


Figure 5.2: System Architecture of Multimodal Misogyny Detection Method

3. We employ an additive random perturbations-based test time augmentation method to help enhance its ability to generalize to variations in the test data. It is achieved through the application of diverse augmentations to the test samples, the model is subjected to a broader range of inputs, enhancing its robustness.
4. We perform extensive analysis of the proposed framework on two diverse multimodal datasets. Experimental evaluations demonstrate and affirm the efficacy of our approach.

5.2 Methodology

This section presents a detailed description of the proposed approach for detecting misogynistic instances on social media. The architecture explaining the proposed method is shown in Figure 5.2. We first apply data cleaning and extract features for Graph and Attention modules. These features are passed through corresponding networks, MANM and GFRM. Content-specific features such as toxicity features from text and captions from images are also extracted to gain insights into the content. Content-specific features are passed through CFLM. Next, the refined features are then combined to form a unified representation for final

classification. Subsequent sections elaborate on the proposed methodology.

5.2.1 Data Preprocessing

Preprocessing of data involves the cleaning of both images and text. Image cleaning entails the removal of text from images through processes like image masking and inpainting. Text cleaning is conducted to improve the quality and standardize textual data. The preprocessed content aids in accurate feature extraction.

5.2.1.1 Image Cleaning

The multimodal datasets we use have text superimposed on the images. We employ image cleaning through the image inpainting technique proposed by Yu et al. [158] to remove the text from images. The cleaning process includes masking and inpainting. In the masking step, text regions are identified in the image and a binary image mask is created that isolates text from the inherent visual content. In the inpainting step, the removed areas are reconstructed to ensure that inpainted regions blend seamlessly with nearby regions, resulting in visually appealing and coherent images. This facilitates the generation of more accurate and relevant feature representations as the emphasis is directed towards the intrinsic visual content of the image. Two examples of cleaned images are shown in Figure 5.3.



Figure 5.3: Example of Cleaned Images

5.2.1.2 Text Cleaning

The text cleaning process involves several steps to enhance the quality and consistency of textual data [159]. It includes converting text to lowercase and removing unwanted characters, symbols, and excessive white spaces. The process extends to the removal of URLs or

usernames.

5.2.2 Feature Extraction

To enhance the capabilities of our framework, we incorporate diverse feature extraction techniques for images and text alongside content-specific features, providing a more comprehensive understanding of the data.

5.2.2.1 Feature Extraction for Graph-based Module

For the proposed GFRM, we extract image and text features through the Contrastive Language-Image Pre-training (CLIP) model [160]. CLIP is a versatile model that learns representations by aligning images and their associated textual descriptions. CLIP encoder employs a transformer architecture similar to Vision Transformers (ViT) to extract visual features and utilizes a causal language model to capture text features. The encoder processes the image-text pair and subsequently maps image-text to a shared latent space with equivalent dimensions, i.e., image embeddings $E^{\text{img}} \in \mathbb{R}^{512}$ and text embeddings $E^{\text{txt}} \in \mathbb{R}^{512}$.

5.2.2.2 Feature Extraction for Attention Module

The proposed MANM requires region embeddings from images and word embeddings from the text. We utilize pretrained BERT [100], a transformer-based model, to generate word embeddings. BERT is designed to capture the bidirectional context for each word. It takes tokens (words) as inputs and generates word embeddings of 768 dimensions.

$$F^{\text{txt}} = \overrightarrow{\text{BERT}}(w) \quad (5.1)$$

Here, F^{txt} denotes the token feature matrix for a sentence, $F^{\text{txt}} = \{q_k\}_{k=1}^{100}$, where $q_k \in \mathbb{R}^{768}$ represents the token embedding of the k -th token, and w denotes the tokens. We consider a maximum of 100 tokens.

For visual feature extraction, MASK RCNN-based Detectron2 is employed that provides region-based features [161]. This model utilizes ResNet-101 as its backbone [162]. ResNet-101-based MASK RCNN is a popular architecture for image feature extraction and instance

5.2 Methodology

segmentation. To extract visual embeddings, we need features from various regions in the image. In this work, we refer to this model as MASK RCNN. The region proposal network detects object regions (bounding boxes). The region bounding-boxes are used to extract features. MASK RCNN ensures that the features are consistently sized and aligned with the regions.

$$F^{\text{img}} = \text{MASK_RCNN}(B) \quad (5.2)$$

The F^{img} represents the features extracted from the region proposals B . These features capture the information about the object contained within the regions. We thus get a feature matrix $F^{\text{img}} = \{r_k\}_{k=1}^{100}$ for each image, where $r_k \in \mathbb{R}^{1024}$ denotes the k th region's embedding of a given image.

5.2.2.3 Feature Extraction for CFLM

We incorporate four additional features, namely, toxicity features, image caption features, NSFW features, and misogyny-specific lexicon scores, to enhance the text analysis capabilities. These features are learned in a content-specific feature learning module.

Toxicity features: We have used toxic-bert to generate the toxicity features and their score for the text¹. Developed at Unitary Labs, the toxic-bert model uses the bert-base-uncased [100] transformer method. BERT consists of multiple encoder layers stacked on top of each other. Each layer refines the understanding of the sentence by passing information between words through the self-attention mechanism. The bert-base-uncased model has a total of 12 encoder layers. The toxic-bert is trained on the Toxic Comment Classification Challenge dataset². The dataset defines six toxicity levels which include threat, insult, toxic, identity hate, obscene, and severe toxic. We compute these six toxicity feature scores that are represented by f^{tox} where $f^{\text{tox}} \in \mathbb{R}^6$.

NSFW features: NSFW model is used to identify indecent content³. The NSFW model we utilize is based on Mobilenetv2 [163]. Unlike traditional CNNs, where residual connections are between expanded layers, Mobilenetv2 uses inverted residual blocks. This means

¹Github. <https://github.com/unitaryai/detoxify>

²Github. <https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge>

³Github. https://github.com/GantMan/nsfw_model

the connections are between thin bottleneck layers, making the model more efficient. Mobilenetv2 boasts a significantly lower parameter count compared to traditional CNNs. It excels in tasks like object recognition, scene classification, and image segmentation. NSFW model, based on Mobilenetv2, is trained on a dataset collected using NSFW Data Scraper, which uses more than 1.5 million source URLs to collect images. The trained NSFW model uses an image and gives probabilities of five classes which are drawing, hentai, neutral, porn, and sexy. We denote NSFW features with f^{nsfw} where $f^{\text{nsfw}} \in \mathbb{R}^5$.

Misogyny-specific Lexicon score (MSL): Misogynous text might contain a few words that are explicitly offensive towards women. Through the MSL score, we aim to compute the number of such words present in a text. To achieve this, we collect a set of misogynous lexicons [164, 165]. This set is further extended by adding synonyms, resulting in a large set of misogynous lexicons. To compute the msl score, we tokenize a given sentence and match each token to this list. After calculating the number of misogynous lexicons in each sentence, this number is normalized using min-max normalization. We denote the msl score with f^{msl} .

Image Caption Features: To leverage additional information from the image, we use image caption features. Image captions are generated using pretrained Bootstrapping Language-Image Pre-training (BLIP) method [166]. It is trained on the COCO dataset for image-captioning, and it uses ViT-large as the backbone. The features for generated captions are extracted using the CLIP text feature extractor. Caption features are denoted with f^{cap} where $f^{\text{cap}} \in \mathbb{R}^{512}$.

5.2.3 Multimodal Attention Module (MANM)

Attention mechanisms play a crucial role in Deep Learning, especially in Computer Vision and Natural Language Processing (NLP). We first apply a novel context-aware cross-attention to get cross-modal representations of both modalities, and then self-attention is applied to focus on pertinent features within cross-modal features. Next, mean pooling is applied to represent the aggregate sequence embedding.

5.2.3.1 Multimodal Context-aware Attention (MCAN)

Multimodal Context-aware Attention (MCAN) enhances the representations of images and texts by exploiting the inter-modal context information. Figure 5.4 illustrates the MCAN mechanism. MCAN utilizes context-aware region representations of images by leveraging semantic and spatial relations among regions [167]. Internally, it uses cross-attention that aids in propagating the contextual clues between text and images [168]. Before applying MCAN, the dimensions of image and text embeddings are aligned by projecting each image’s region embeddings F^{img} into the same dimension as text embeddings, i.e., 768. The projection is done using the fully-connected dense layer as follows:

$$F_{\text{proj}}^{\text{img}} = F^{\text{img}} \cdot W_i + b_i \quad (5.3)$$

where, $W_i \in \mathbb{R}^{1024 \times 768}$ is a learnable projection matrix and b_i is a bias vector. Hence, we have image-embeddings $F_{\text{proj}}^{\text{img}} = \{r_k\}_{k=1}^{100}$ and text embeddings $F^{\text{txt}} = \{q_k\}_{k=1}^{100}$, where $r_k, q_k \in \mathbb{R}^{768}$.

Next, the query, key, and value are computed for the image as well as the text. Let us assume for an image-text pair, the query, key, and value vectors are defined as follows:

$$\begin{cases} Q_i = F_{\text{proj}}^{\text{img}} \cdot (W_{Q_i}) \\ K_i = F_{\text{proj}}^{\text{img}} \cdot (W_{K_i}) \\ V_i = F_{\text{proj}}^{\text{img}} \cdot (W_{V_i}) \end{cases} \quad (5.4)$$

$$\begin{cases} Q_t = F^{\text{txt}} \cdot (W_{Q_t}) \\ K_t = F^{\text{txt}} \cdot (W_{K_t}) \\ V_t = F^{\text{txt}} \cdot (W_{V_t}) \end{cases} \quad (5.5)$$

where, (Q_i, K_i, V_i) are the query, key, and value of the image modality, (Q_t, K_t, V_t) represent the same for the text, $(W_{Q_i}, W_{K_i}, W_{V_i})$ denote the weight matrices associated with the query, key, and value projections of the image, $(W_{Q_t}, W_{K_t}, W_{V_t})$ are weight matrices associated with the query, key, and value projections of text, and $(F^{\text{img}}, F^{\text{txt}})$ denote the

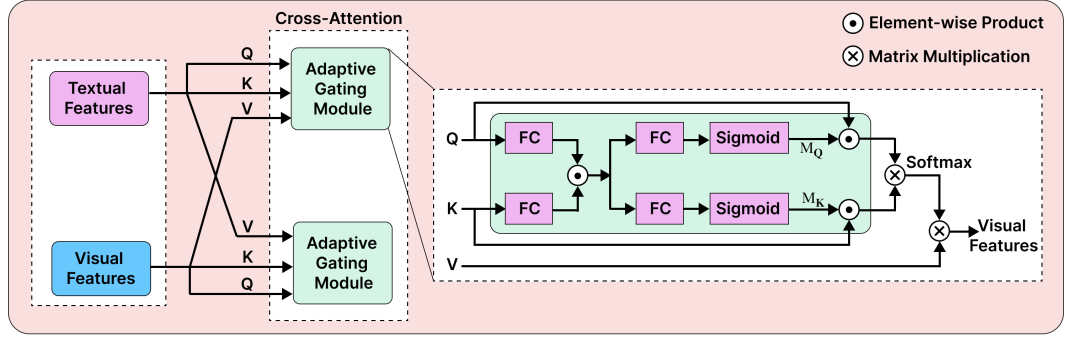


Figure 5.4: Adaptive Gating-based Cross Attention for MCAN

context-aware region embeddings of image and word embeddings for text, respectively.

3.3.1.1. Cross-Attention : We employ cross-attention between the two modalities by taking the value from the one while using the other for the query and key [168]. The two outputs are provided by the cross-attention, $\mathcal{X}^{\text{img}}$ and $\mathcal{X}^{\text{txt}}$ that denote the cross-attended image features and cross-attended text features, respectively. The cross-attention operations to generate $\mathcal{X}^{\text{img}}$ and $\mathcal{X}^{\text{txt}}$ are defined as follows:

$$\text{Cross_Att}(Q_t, K_t, V_i) = \text{softmax} \left(\frac{Q_t \cdot K_t^T}{\sqrt{d_k}} \right) \cdot (V_i) \quad (5.6)$$

$$\text{Cross_Att}(Q_i, K_i, V_t) = \text{softmax} \left(\frac{Q_i \cdot K_i^T}{\sqrt{d_k}} \right) \cdot (V_t) \quad (5.7)$$

where T is the transpose, $\text{Cross_Att}(\cdot)$ denotes cross-attention operation and d_k denotes the key vector's dimension.

3.3.1.2. Gating Mechanism : Instead of applying dot-product between a query Q and key K as shown in Equation 5.6 and Equation 5.7, we use Adaptive Gating Module (AGM) with fusion strategy [167]. AGM aims to dynamically regulate the information exchange between queries (Q) and keys (K) with the goal of enhancing meaningful interactions and restraining noise or irrelevant information. Within AGM, queries and keys undergo initial projection

into a shared space, followed by fusion. The fusion operation is expressed as follows:

$$G = (Q \cdot W_{Q_G} + b_{Q_G}) \odot (K \cdot W_{K_G} + b_{K_G}) \quad (5.8)$$

where, G is the fusion result, (W_{Q_G}, W_{K_G}) are learnable projection matrices, and (b_{Q_G}, b_{K_G}) are bias vectors, and $\odot$ is the element-wise product.

Gating masks, M_Q and M_K , corresponding to Q and K , are generated by applying fully-connected layers followed by the sigmoid function as shown in Equation 5.9 and Equation 5.10.

$$M_Q = \sigma(G \cdot W_{Q_M} + b_{Q_M}) \quad (5.9)$$

$$M_K = \sigma(G \cdot W_{K_M} + b_{K_M}) \quad (5.10)$$

Here, $\sigma(\cdot)$ denotes the sigmoid function, (W_{Q_M}, W_{K_M}) are learnable weights, and (b_{Q_M}, b_{K_M}) bias vectors. The obtained gating masks, M_Q and M_K , are used to control the information flow of the original queries and keys in the attention mechanism. The dot-product attention is applied with this controlled flow as shown in Equation 5.11, hence providing context-enhanced cross-modal representation.

$$\mathcal{X} = \text{Cross_Att}(M_Q \odot Q, M_K \odot K, V) \quad (5.11)$$

Here, $\mathcal{X}$ is the result of the gated cross-attention.

This mechanism allows the model to selectively focus on meaningful interactions, suppress irrelevant or noisy information, and provide a context-enhanced cross-attention mechanism for various tasks.

3.3.1.3 Context-aware Image Region Features : Context-aware image region embeddings are calculated by performing the element-wise dot product between region embeddings and

position embeddings [167]. We encapsulate semantic embeddings with $F_{\text{proj}}^{\text{img}} \in \mathbb{R}^{R \times D}$, and position embeddings with $\hat{I}^{\text{pos}} \in \mathbb{R}^{R \times D}$. $\hat{I}^{\text{pos}}$ is computed by taking absolute position encodings [169] of image regions, aspect ratio and area of image [167]. These values are normalized, and we get a vector $I \in \mathbb{R}^6$. To obtain position embeddings $\hat{I}^{\text{pos}}$, I is projected to the same dimension as region embeddings, i.e., 768 using a fully-connected dense layer as shown in Equation 5.12.

$$\hat{I}^{\text{pos}} = \sigma(W_I \cdot I + b_I) \quad (5.12)$$

Here, $W_I \in \mathbb{R}^{768 \times 6}$ is weight matrix and b_I is bias. Next, fusion is performed between image region embeddings and positional embeddings to obtain context-aware region features, $\tilde{F}^{\text{img}}$, as shown in Equation 5.13.

$$\tilde{F}^{\text{img}} = F_{\text{proj}}^{\text{img}} \odot \hat{I}^{\text{pos}} \quad (5.13)$$

Hence, in Equation 5.4 we pass context-aware image features $\tilde{F}^{\text{img}}$ in place of $F_{\text{proj}}^{\text{img}}$ which return cross-modal image features $\mathcal{X}^{\text{img}}$. $\tilde{F}^{\text{img}}$ leverages the global spatial features as well as semantic region features, which help in getting context-enhanced cross-modal features.

5.2.3.2 Multihead Self-Attention (MSAN)

Multihead Self-Attention (MSAN) enhances the self-attention mechanism by employing multiple sets of queries, keys, and values, which are referred to as attention heads [170]. Each attention head processes the input independently, capturing different aspects and relationships within the sequence. The outputs of the attention heads are then concatenated to create a comprehensive representation that integrates information from different attention subspaces. We apply MSAN on cross-modal features obtained through MCAN to focus on pertinent information within cross-modal features. In MSAN, each attention head performs the self-attention operation as shown in Equation 5.14.

$$Attention(Q_k, K_k, V_k) = \text{softmax} \left(\frac{Q_k K_k^T}{\sqrt{d}} \right) V_k \quad (5.14)$$

Here, Q_k , K_k , and V_k denote the query, key, and value of k th head. Outputs from all the heads are concatenated and subjected to linearly transformation to get the final representation.

$$\text{concat_output} = \text{concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_H) \quad (5.15)$$

Here, head_k denote the output from k -th head. In the MSAN, we pass cross-modal image and text features, $\mathcal{X}^{\text{img}}$ and $\mathcal{X}^{\text{txt}}$ which returns self-attended features, $\tilde{\mathcal{X}}^{\text{img}}$ and $\tilde{\mathcal{X}}^{\text{img}}$ as shown in Equation 5.16:

$$\tilde{\mathcal{X}}^{\text{img}}, \tilde{\mathcal{X}}^{\text{txt}} = \text{MSAN}(\mathcal{X}^{\text{img}}, \mathcal{X}^{\text{txt}}) \quad (5.16)$$

where, $\tilde{\mathcal{X}}^{\text{img}}, \tilde{\mathcal{X}}^{\text{txt}} \in \mathbb{R}^{128 \times 256}$.

5.2.3.3 Pooling and Concatenation

Next, we apply max pooling on $\tilde{\mathcal{X}}^{\text{img}}$ and $\tilde{\mathcal{X}}^{\text{txt}}$ to get sequence vectors such that both vectors are of size $\mathbb{R}^{256}$. Subsequently, resultant vectors are concatenated to get the final multimodal attended feature vector of the image-text.

$$\tilde{\mathcal{X}}^{\text{img-txt}} = \text{concat}(\text{max_pool}(\tilde{\mathcal{X}}^{\text{img}}, \tilde{\mathcal{X}}^{\text{txt}})) \quad (5.17)$$

Here, $\tilde{\mathcal{X}}^{\text{img-txt}} \in \mathbb{R}^{512}$ denotes the final joint representation of multimodal attended image-text features.

5.2.4 Graph-based Feature Reconstruction Module (GFRM)

The Graph-Based Feature Reconstruction Module enhances unimodal features through a graph neural network. Feature reconstruction occurs within the context of node neighborhoods, which are determined by the similarity of each node to others. To achieve this, we introduce unimodal graphs where features serve as nodes, and edges are established by computing cosine similarity between nodes. A pictorial representation of graph construction and feature reconstruction method is shown in Figure 5.5. Subsequent sections elaborate on graph construction and feature refinement.

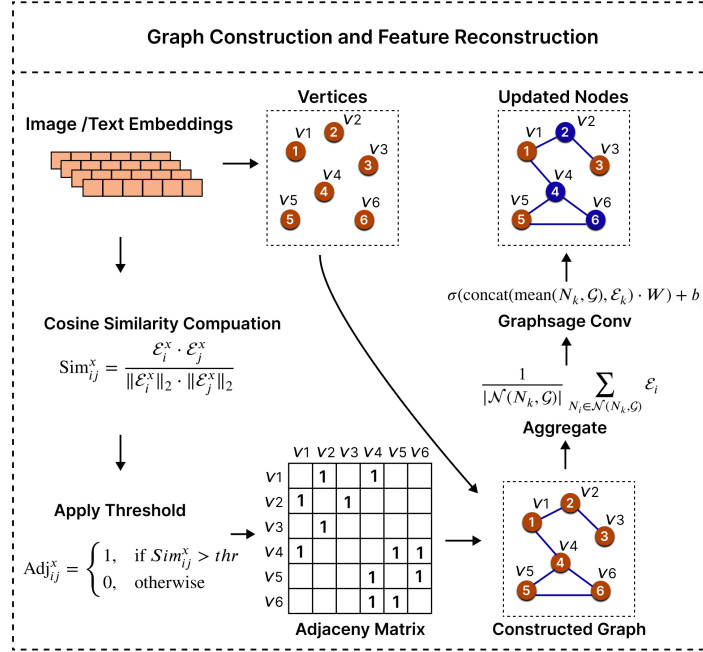


Figure 5.5: Graph Construction and Node Update

5.2.4.1 Graph Construction

As shown in Algorithm 5.1, the graph is constructed by calculating cosine similarity between nodes and forming an adjacency matrix.

Let us assume $\mathcal{M}^x = \{E_i^x\}_{i=1}^R$, where $\mathcal{M}^x \in \mathbb{R}^{R \times D}$ is the embedding matrix, $x \in \{\text{txt}, \text{img}\}$ denotes one of the modalities, R denotes the total number of embeddings, and $D = 512$ represents the dimension. We first normalize $\mathcal{M}^x$ with L2 normalization, $\mathcal{M}_{\text{norm}}^x = \{\mathcal{E}_i^x\}_{i=1}^R$.

5.2 Methodology

We compute the cosine similarity between each pair of embeddings to obtain the similarity matrix, sim^x as shown in Equation 5.18:

$$\text{sim}_{ij}^x = \text{cos_sim}(\mathcal{E}_i^x, \mathcal{E}_j^x) = \frac{\mathcal{E}_i^x \cdot \mathcal{E}_j^x}{\|\mathcal{E}_i^x\|_2 \cdot \|\mathcal{E}_j^x\|_2} \quad (5.18)$$

where, sim_{ij}^x is the similarity score of two embeddings. Upon computing similarities between every pair, we get a square matrix of size $R \times R$.

Algorithm 5.1 Graph Construction

Input: $\mathcal{M}$: Embeddings matrix
thr: Similarity threshold
Output: Graph $\mathcal{G}$

```

1: function COS_SIM( $\tilde{\mathbf{V}}_1, \tilde{\mathbf{V}}_2$ )
2:   return  $\frac{\tilde{\mathbf{V}}_1 \cdot \tilde{\mathbf{V}}_2}{\|\tilde{\mathbf{V}}_1\| \cdot \|\tilde{\mathbf{V}}_2\|}$ 
3: end function
4: function GRAPH_CONSTRUCTION( $\mathcal{M}, thr$ )
5:   Adj  $\leftarrow$  [ ] [ ]
6:   for  $i \leftarrow 1$  to  $\text{len}(\mathcal{M})$  do
7:     for  $j \leftarrow 1$  to  $\text{len}(\mathcal{M})$  do
8:       sim  $\leftarrow$  cos_sim( $\mathcal{M}_i, \mathcal{M}_j$ )
9:       if sim > threshold then
10:        Adj[ $i$ ][ $j$ ]  $\leftarrow$  1
11:       end if
12:     end for
13:   end for
14:   return Graph  $\mathcal{G}$ (Adj)
15: end function
```

Next, we compute an adjacency matrix Adj^x by putting an edge between two nodes based on the similarity threshold, thr .

$$\text{Adj}_{ij}^x = \begin{cases} 1, & \text{if } \text{sim}_{ij}^x > thr \\ 0, & \text{otherwise} \end{cases} \quad (5.19)$$

$\text{Adj}_{ij}^x = 1$ signifies the existence of an edge, whereas $\text{Adj}_{ij}^x = 0$ signifies that there is no edge between any two arbitrary nodes i and j .

5.2.4.2 Feature Reconstruction

Unimodal features are reconstructed to enhance their representation. We employ GraphSAGE, an inductive learning-based graph neural network for feature reconstruction [171]. GraphSAGE can generalize its learning to unseen nodes during training (inductive). GraphSAGE acquires node embeddings by aggregating information from the neighborhood of each node. Different aggregation functions can be used in GraphSAGE, such as mean, max, sum, and min aggregation. We adopt mean aggregation as shown in Equation 5.20:

$$\text{mean}(N_k, \mathcal{G}) = \frac{1}{|\mathcal{N}(N_k, \mathcal{G})|} \sum_{N_i \in \mathcal{N}(N_k, \mathcal{G})} \mathcal{E}_i \quad (5.20)$$

where, $\mathcal{G}$ represents the graph, $\mathcal{N}(N_k, \mathcal{G})$ is the set of neighbors for the node N_k , and $\mathcal{E}$ is the embedding such that $\mathcal{E} \in \mathcal{M}$. Aggregating neighboring feature vectors, the current representation of node $\mathcal{E}_k$ corresponding to N_k is subsequently combined through concatenation with the aggregated neighborhood vector. The concatenated features are passed through a fully connected layer utilizing learnable weight W , an activation function σ , and bias b .

$$\mathcal{H} = \sigma(\text{concat}(\text{mean}(N_k, \mathcal{G}), \mathcal{E}_k) \cdot W) + b \quad (5.21)$$

Finally, we concatenate image and text features, $\mathcal{H}^{\text{img}}$ and $\mathcal{H}^{\text{txt}}$, to form a multimodal relation feature vector denoted as $\tilde{\mathcal{H}}^{\text{img-txt}} \in \mathbb{R}^{512}$, where, $\mathcal{H}^{\text{img}} \in \mathbb{R}^{256}$ and $\mathcal{H}^{\text{txt}} \in \mathbb{R}^{256}$ represent graph-reconstructed image and text features, respectively.

5.2.5 Content-specific Feature Learning Module (CFLM)

In this module, all the content-specific features described in Section 5.2.2.3 are learned. These features include image-based features and text-based features. Image-based features include NSFW features $f^{\text{nsfw}} \in \mathbb{R}^5$ and caption features $f^{\text{cap}} \in \mathbb{R}^{512}$, whereas text-based features include toxicity features $f^{\text{tox}} \in \mathbb{R}^6$ and MSL score $f^{\text{msl}} \in \mathbb{R}^1$. These features are first concatenated to get a joint representation $\tilde{f} \in \mathbb{R}^{524}$. Next, the vector $\tilde{f}$ is passed through

a fully connected layer to get the learned joint vector.

$$\tilde{\mathcal{F}} = \sigma(W_f \cdot \tilde{f} + b_f) \quad (5.22)$$

Here, $\tilde{\mathcal{F}} \in \mathbb{R}^{256}$ is learned joint vector for content-specific features, $W_f \in \mathbb{R}^{256 \times 512}$ is weight matrix, σ denotes activation function and b_f is bias. Joint vector $\tilde{\mathcal{F}}$ is then concatenated with output vectors of MANM and GFRM modules for joint learning and classification.

5.2.6 Classification Module

The classification module generates final predictions of given multimodal content. During testing, it applies test-time augmentation, while training is performed only on original samples. The classification module first concatenates the features obtained through MANM, GFRM, and CFLM as shown in Equation 5.23:

$$\tilde{\mathcal{Y}}^{\text{joint}} = \text{concat}(\tilde{\mathcal{X}}^{\text{img-txt}}, \tilde{\mathcal{H}}^{\text{img-txt}}, \tilde{\mathcal{F}}) \quad (5.23)$$

where, $\tilde{\mathcal{Y}}^{\text{joint}} \in \mathbb{R}^{1280}$, represents the final joint multimodal vector. The integrated representation of these features is passed through a series of three fully connected layers, which have output dimensions of 1024, 512, and 256 respectively. Each fully connected layer is followed by a dropout layer of 0.3. A sigmoid-activated fully connected layer is used to generate the final probability. A pictorial representation of the classification module is shown in Figure 5.6.

During training, we use Adam optimizer, a batch size of 128, a learning rate of 1e-4, and 100 epochs. Experiments are performed on a system with an Intel(R) Xeon(R) Silver 4215R 3.20GHz processor, 384 GB RAM, and Tesla T4 GPU. We used the 4.12.0 version of conda on the Ubuntu-20.04.2 operating system. During testing, we employ TTA, which is discussed in the next subsection.

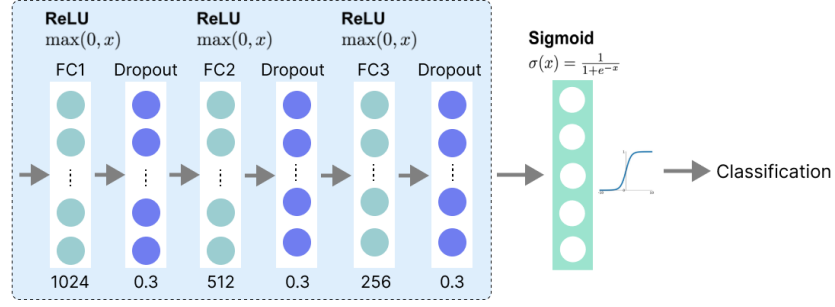


Figure 5.6: Classification Module

5.2.6.1 Test-Time Augmentation (TTA)

Test-Time augmentation (TTA) is performed during testing by augmenting multiple versions of a test sample and aggregating the predictions on these samples to get the final prediction. TTA has been explored in Computer Vision [172, 173] and text classification tasks [144, 174]. Figure 5.7 illustrates augmentation and final prediction generation approach at test-time.

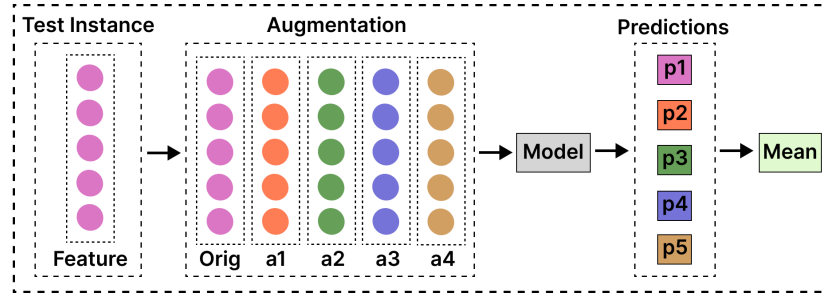


Figure 5.7: Test-Time Augmentation

We employ feature space data augmentation using additive random perturbations [175]. Algorithm 5.2 represents the steps for the additive random perturbation method. To apply additive random perturbations to feature vectors, we generate a vector with random values between $-\alpha$ and $+\alpha$. Subsequently, an element-wise sum is performed between the feature vector and random vector that provides a new synthetic feature vector. The same approach is used to generate synthetic feature vectors for both modalities. To make sure that synthetic vectors are not very similar or very dissimilar to the original feature vector, we take only those synthetic vectors that have cosine similarity between 0.6 and 0.7, including the boundary conditions. For cosine-similarity computation between two 2D vectors, we take a mean of 2D vectors to get 1D sequence vectors.

We generate four additional synthetic feature embeddings for each test sample in order

5.3 Experimental Evaluations

Algorithm 5.2 Additive Random Perturbation

Input: H^{txt} : Text features of test set
 H^{img} : Image features of test set
Output: $S_{\text{aug}}^{\text{txt}}, S_{\text{aug}}^{\text{img}}$: Augmented test sets
function: $\text{TTA}(H^{\text{txt}}, H^{\text{img}})$

- 1: $S_{\text{aug}}^{\text{txt}}, S_{\text{aug}}^{\text{img}} \leftarrow \{ \}$
- 2: **for each** $h^{\text{txt}} \in H^{\text{txt}}$ **and** $h^{\text{img}} \in H^{\text{img}}$ **do**
- 3: **for** $i > n$ **do**
- 4: $\mathcal{R}_t \leftarrow \text{rand_vector}(\text{len}(h^{\text{txt}}), p)$
- 5: $\mathcal{R}_v \leftarrow \text{rand_vector}(\text{len}(h^{\text{img}}), p)$
- 6: $S_{\text{aug}}^{\text{txt}} \leftarrow h^{\text{txt}} + \mathcal{R}_t$
- 7: $S_{\text{aug}}^{\text{img}} \leftarrow h^{\text{img}} + \mathcal{R}_v$
- 8: **end for**
- 9: **end for**
- 10: **return** $S_{\text{aug}}^{\text{txt}}, S_{\text{aug}}^{\text{img}}$

to apply TTA. During test-time augmentation, we are able to generate the synthetic embeddings; however, for content-specific features, we use the same values as extracted for the original sample.

5.3 Experimental Evaluations

The experimental setup is initially explained in this section, followed by the experimental results to illustrate the effectiveness of the proposed method.

5.3.1 Experimental Setup

In this section, we provide an overview of the datasets utilized for experimental evaluations. Subsequently, we delve into the discussion of evaluation metrics and distinct methods employed for comparative analysis.

5.3.1.1 Summarization of Datasets

We perform misogyny detection and sexism detection tasks on two different datasets, namely, Multimedia Automatic Misogyny Identification (MAMI) and MMHS150K. The misogyny detection task is performed on the MAMI dataset that is provided in Task-5 of SemEval-2022 competition [157]. For sexism detection, we utilize the MMHS150K dataset [51].

- a) *MAMI*: The MAMI dataset has 10,000 annotated memes. The dataset has 5,000 misogynous and 5,000 non-misogynous memes in the training set. The testing set has a total of 1,000 samples that are equally distributed in the misogynous and non-misogynous classes.
- b) *MMHS150K*: The MMHS150K dataset comprises 1,50,000 tweets with text and images collected from Twitter. It focuses on tweets containing 51 Hatebase terms associated with hate speech. Among different hate speech classes, we utilize samples only from the sexism class, which has 3,494 samples. For the non-hate class, we take a total of 10,000 samples. The total samples, 13,494, are then split into a ratio of 80:20 for training and testing sets.

5.3.1.2 Comparison Methods

We conduct a comparative analysis with the following approaches for misogyny detection:

- a) *Unimodal Methods*: We employ a state-of-the-art transformer model, specifically BERT [100] for textual data and MASK RCNN RESNET-101 [176] for images, to independently generate embeddings or feature representations for each modality. BERT is applied to capture contextual information within the text, while MASK RCNN RESNET-101 excels in extracting meaningful region-based features from images.
- b) *Multimodal Methods*: We use a combination of images and text for misogyny detection. We compare the performance of the proposed method on both datasets with multimodal methods that include the Late Fusion method with BERT and MASK RCNN RESNET-101 features [177], VisualBERT [178], CLIP [160], and ViLT [179]. We also compare the performance of the proposed approach on MAMI with the following existing state-of-the-art methods:

BERT+ViT [180]: The method combines the BERT model, trained on hateful memes from Facebook AI, with ViT. The text is analyzed by the BERT model, while the image is processed by the ViT. These two outputs are then integrated using an attention mechanism.

RIT Boston [181]: This method employs the CLIP to obtain consistent visual and

language representations. These features are then used in a simple logistic regression model to make binary predictions.

DD-TIG [182]: DD-TIG method employs ERNIE-ViL-Large as the foundational model. To address the issue of model overfitting, it implements strategies to reduce the impact of biased words and template memes. In addition, it utilizes image captions.

SRCB [4]: The method employs an XGBoost classifier, utilizing image features extracted by the CLIP model. Building upon this, a system is proposed with an ensemble incorporating pretraining models, a boosting method, and rule-based adjustments. Text information is integrated into the system through a late sequential fusion approach.

For comparison on the MMHS150K dataset, along with BERT+ViT [180], we use the following state-of-the-art methods:

FCM [51]: Feature Concatenation Model (FCM) utilizes Inception v3 for visual modality representation and Glove word embeddings for text modality. FCM concatenates these representations to obtain a joint vector that passes through a sequence of three fully connected layers for final classification.

MSKAV+KDAC [13]: The presented framework includes a module known as the ‘multi-scale kernel attentive visual’ (MSKAV), employing an efficient multi-branch structure to extract distinctive visual features. The model also introduces a new component, the ‘knowledge distillation-based attentional caption’ (KDAC) module.

ARC-NLP [183]: This approach uses a multimodal method that leverages both textual and visual features. A syntactical feature vector and Bag-of-Words (BoW) method-based n-grams are incorporated as additional features. Multiple classifiers are used for ensemble learning in this method.

Additionally, we compare the performance of the proposed method with three recent state-of-the-art methods, KOSMOS-2 [184], OFA [185] and GPT-4o [186]. KOSMOS-2 is employed by extracting its multimodal embeddings, whereas OFA and GPT-4o are employed using a prompt-based zero-shot classification approach.

5.3.2 Results

This section presents experimental results and comparisons. First, a comparison of the proposed approach with different unimodal and multimodal methods is presented. Subsequently, an ablation analysis is conducted, exploring the influence of various components and features.

Table 5.1: Performance comparison over MAMI dataset

	Method	Acc	P	R	F1
Unimodal	BERT [100]	68.16	67.84	69.20	68.16
	Mask RCNN [176]	63.16	65.13	56.80	63.01
Multimodal	Late Fusion (MASK RCNN + BERT) [177]	69.26	68.03	72.76	69.23
	VisualBert [178]	73.85	69.48	78.80	73.94
	CLIP [160]	78.10	76.77	80.06	78.08
	ViLT [179]	73.20	66.66	92.80	73.12
	BERT + ViT [180]	86.20	-	88.10	87.40
	RIT Boston [181]	-	-	-	77.80
	DD-TIG [182]	79.40	-	-	79.30
	SRCB [4]	-	-	-	83.40
	KOSMOS-2 [184]	74.07	74.22	74.07	74.03
Prompt-based	OFA [185]	57.61	57.62	57.61	57.58
	GPT-4o [186]	83.18	84.94	83.17	82.96
	Proposed	87.42	83.81	89.90	87.95

5.3.2.1 Performance Comparison over MAMI Dataset

In this section, we undertake a thorough comparison between our proposed method and existing approaches on the MAMI dataset. This comparative analysis seeks to highlight the distinctive strengths and advantages of our methodology in relation to the prevailing unimodal and multimodal techniques. The results are shown in Table 5.1.

The proposed model outperforms the unimodal approaches as it learns from both texts as well images. These features allow our proposed model to achieve an improvement of 19.79% in terms of macro-F1 over the unimodal BERT approach and 24.94% in terms of macro-F1 over the unimodal MASK RCNN RESNET-101 approach. In comparison to multimodal approaches, the proposed method shows significant performance gains on different evaluation metrics. The proposed model improves the macro-F1 score by 18.72%, 14.01%,

5.3 Experimental Evaluations

9.87%, and 14.83% for the late-fusion method, VisualBERT, CLIP, and VILT, respectively. Our proposed model outperforms the existing state-of-the-art works. In terms of macro-F1, it achieves an absolute improvement of 4.55% over the SRCB, 8.55% over the DD-TIG, 0.55% over the BERT +ViT and 10.15% over the RIT Boston. In comparison to the recent state-of-the-art techniques KOSMOS-2, OFA, and GPT-4o, the proposed method achieves a gain of 4.28%, 28.83%, and 14.77%, respectively, in terms of macro-F1. The huge performance gain over OFA and GPT-4o can attributed to the fact that these two models are used in a prompt-based zero-shot approach. The comparison results on the MAMI dataset denote the efficacy of the proposed framework in misogyny detection. The performance gain on other methods is due to the fact that the proposed method employs a dedicated graph-based module for unimodal feature refinement and a dedicated attention module for cross-modal features. It also utilizes the synergy of other features, such as toxicity features, misogyny lexicons, NSFW model features, and caption embeddings. Additionally, to generalize the predictions, we employ TTA.

5.3.2.2 Performance Comparison over MMHS150K Dataset

The MMHS dataset contains multimodal content that has two classes: sexism and non-sexism. The proposed approach outperforms the existing methods by a significant margin in terms of accuracy, precision, recall, and F1 score. The results are shown in Table 5.2.

The proposed model outperforms the unimodal approaches by achieving an improvement of 17.76% and 16.39% in terms of macro-F1 score over the unimodal BERT and the unimodal MASK RCNN RESNET-101 approach, respectively. The results outline the importance of using both modalities. The proposed model outperforms the multimodal approaches as it learns from content-specific features, test time augmentation, MCAN, and graph-based features. This allows it a deeper understanding of misogyny-related content. The proposed model improves macro-F1 score by 13.15%, 11.66%, 2.71%, and 12.78% for late-fusion, VisualBERT, CLIP, and VILT, respectively. Our proposed model outperforms the existing state-of-the-art methods. It achieves an absolute improvement of 7.19%, 9.93%, 15.27%, and 12.20% in terms of accuracy, precision, recall, and macro-F1 score over BERT +ViT. It also gains an improvement of 6.82%, 10.79%, 15.19%, and 12.99% in terms of accuracy, precision, recall and macro-F1 score over the FCM, 3.89%, 1.94%, 4.19% and 3.17% in

Table 5.2: Performance comparison over MMHS150K dataset

	Method	Acc	P	R	F1
Unimodal	BERT [100]	78.84	73.97	67.01	68.86
	Mask RCNN [176]	76.84	70.37	70.09	70.23
Multimodal	Late Fusion (MASK RCNN + BERT) [177]	80.69	75.88	72.02	73.47
	VisualBert [178]	81.58	77.26	73.85	74.96
	CLIP [160]	86.63	82.58	85.83	83.91
	ViLT [179]	81.13	76.56	72.62	73.84
	BERT + ViT [180]	80.75	75.33	73.18	74.42
	FCM [51]	81.12	74.47	73.26	73.63
	MSKAV+KDAC [13]	84.05	83.32	84.26	83.45
	ARC-NLP [183]	87.22	83.46	86.17	84.26
	KOSMOS-2 [184]	85.17	80.83	84.95	82.34
Prompt-based	OFA [185]	61.35	58.78	61.07	57.79
	GPT-4o [186]	74.70	71.22	76.28	71.77
	Proposed	87.94	85.26	88.45	86.62

terms of accuracy, precision, recall and macro-F1 over MSKAV+KDAC and 0.72%, 1.80%, 2.28% and 2.36% over the ARC-NLP. The proposed method outperforms KOSMOS-2 by a margin of 4.28% in terms of macro-F1, whereas it outperforms prompt-based methods, OFA and GPT-4o, by 28.83% and 14.85%, respectively, in terms of macro-F1. It achieves significant performance gain over OFA and GPT-4o as they are utilized in zero-shot classification setting through prompts. Overall, the proposed approach outperforms the existing methods across multiple evaluation metrics.

5.3.2.3 Ablation Analysis

In the ablation analysis, we analyze the impact of important components and features on misogyny detection over the MAMI dataset.

a) ***Influence of Different Modules:*** To evaluate the impact of attention and graph-based modules, we first remove one of these components while keeping the other components intact, and then we evaluate its performance on all four evaluation metrics. We then compare the results with the originally proposed model. The results are presented in Table 5.3.

Without the attention-based module MANM, accuracy, precision, recall, and macro-F1 reduce by 4.17%, 2.25%, 4.49%, and 4.66%, respectively. The graph-based module GFRM also has a significant impact on the model’s performance. Without graph-based module ac-

Table 5.3: Ablation analysis on different modules

	Acc	P	R	F1
w/o MANM	83.25	81.56	85.41	83.29
w/o GFRM	84.40	82.11	87.25	84.48
w/o CFLM	84.91	82.26	87.38	84.73
Proposed	87.42	83.81	89.90	87.95

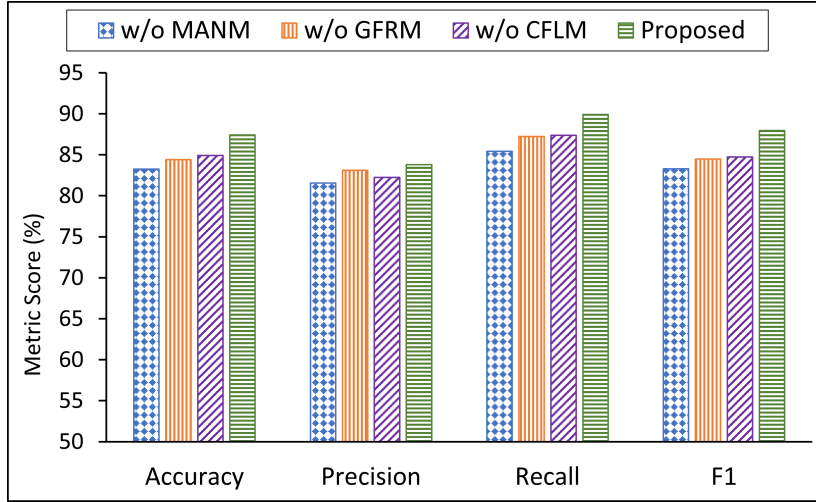


Figure 5.8: Influence of Different Modules

curacy, precision, recall, and macro-F1 score decrease by 3.02%, 1.70%, 2.65%, and 3.47%, respectively. Removal of all the content-specific features reduces the model’s performance by 2.51%, 1.55%, 2.52%, and 3.22% in terms of accuracy, precision, recall, and macro-F1. Figure 5.8 shows a pictorial representation of module influence results. Using attention allows our model to capture context-aware relationships between images and text, enabling it to understand dependencies between relevant regions and words. This empowers the model to concentrate on particular segments of the input by assigning weights determined by their pertinence. The graph-based module is able to provide modality-specific refined features, where content-specific features provide useful insights into the given multimodal that help train the model.

b) ***Influence of Different Attention Mechanisms:*** The proposed approach utilizes two attention mechanisms, MSAN and MCAN. We analyze the influence of MSAN and MCAN for misogynous content identification. To evaluate the performance of an attention mechanism, we remove that attention and analyze the reduction in performance. Next, we evaluate the

combined influence of both attention mechanisms by removing both. When we remove both attention mechanisms, we still use BERT and MASK RCNN features using max-pooling. We evaluate the performance in terms of precision, recall, macro-F1, and accuracy. The results are shown in Table 5.4. Without self-attention, accuracy, precision, recall, and macro-F1 reduce by 2.08%, 0.69%, 2.74%, and 2.57%, respectively. Whereas without MCAN, accuracy, precision, recall, and macro-F1 score decrease by 2.70%, 1.07%, 3.64%, and 3.24%, respectively. When both attention mechanisms are removed, we observe a significant decrease in accuracy, precision, recall, and macro-F1, i.e., 3.79%, 1.23%, 4.34%, and 4.30%, respectively. The results demonstrate that MSA and MCAN attentions allow the model to dynamically adjust its focus based on the surrounding context, enabling more informed and contextually relevant processing of information. Figure 5.9 presents a graphical visualization of the attention-based ablation results.

Table 5.4: Ablation analysis on attention mechanisms

	Acc	P	R	F1
w/o MCAN	84.72	82.74	86.26	84.71
w/o MSA	85.34	83.12	87.16	85.38
w/o Both	83.63	82.58	85.56	83.65
Proposed	87.42	83.81	89.90	87.95

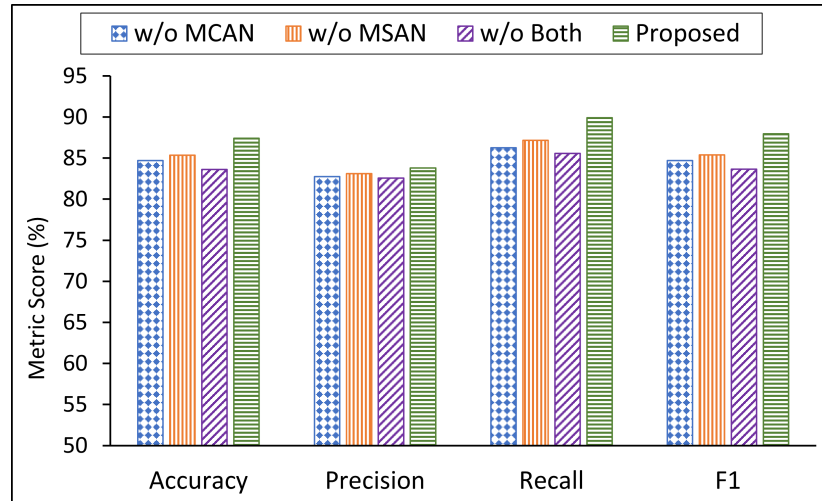


Figure 5.9: Influence of Different Attention Mechanisms

c) *Influence of Modalities*: The influence of modalities on the model’s performance refers to using either only text or only images for misogyny detection while keeping other

5.3 Experimental Evaluations

components the same as before. The results are shown in Table 5.5. For text-only and image-only methods, the MANM module can not be used because it requires cross-attention. Content-specific features are used only for the modality that we are comparing. Removal of MANM, embeddings of other modality, and content-specific features of other modality affect the performance. While using only text modality, the accuracy reduces by 14.94%, precision reduces by 18.76%, recall decreases by 5.70%, and macro-F1 score decreases by 15.10%. These results demonstrate that images help significantly in misogyny detection. Next, we measure the performance using only images. While using only images, the accuracy reduces by 18.32%, precision reduces by 19.39%, recall reduces by 4.39% and macro-F1 score decreases by 18.72%.

As shown in Figure 5.10, the proposed model outperforms the text-only and image-only approaches as it learns from both texts as well as images for classification.

Table 5.5: Ablation analysis on modalities

	Acc	P	R	F1
Text only	72.48	65.05	84.20	72.31
Image only	69.10	64.42	85.51	69.23
Proposed	87.42	83.81	89.90	87.95

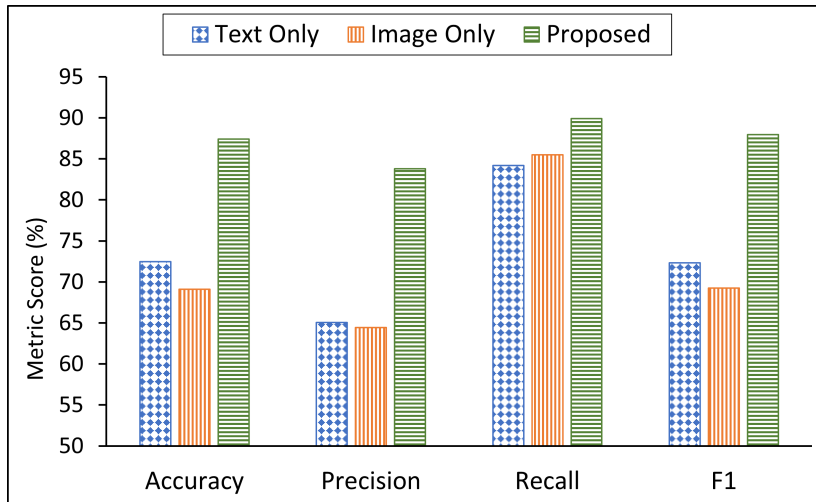


Figure 5.10: Influence of Different Modalities

d) *Influence of Additional Features*: We examine the influence of supplementary features on the performance of our method. In order to assess the impact of a feature, we remove

that feature and get the results. These supplementary features include NSFW features, toxicity features, caption embeddings, and MSL score. Additionally, we also show the influence of TTA. The results of this ablation are shown in Table 5.6. The improvements in terms of accuracy, precision, recall, and macro-F1 score by using these features are as follows: 0.80%, 0.25%, 0.75%, 0.97% (NSFW features), 1.19%, 0.36%, 1.32%, and 1.71% (toxicity features), 1.9%, 0.69%, 2.00%, and 2.34% (caption embeddings), 1.00%, 0.11%, 1.34%, and 1.14% (MSL score), and 2.04%, 1.54%, 2.42%, and 2.60% (TTA). A graphical representation of these results is given Figure 5.11. These features allow the proposed model to understand the context and useful insights of the content by utilizing both modalities, hence improving the overall performance of the model.

Table 5.6: Ablation analysis on additional features

	Acc	P	R	F1
w/o NSFW	86.62	83.56	89.15	86.98
w/o Toxicity	86.23	83.45	88.58	86.24
w/o Caption	85.52	83.12	87.90	85.61
w/o MSL Score	86.43	83.70	88.56	86.81
w/o TTA	85.38	82.27	87.48	85.35
Proposed	87.42	83.81	89.90	87.95

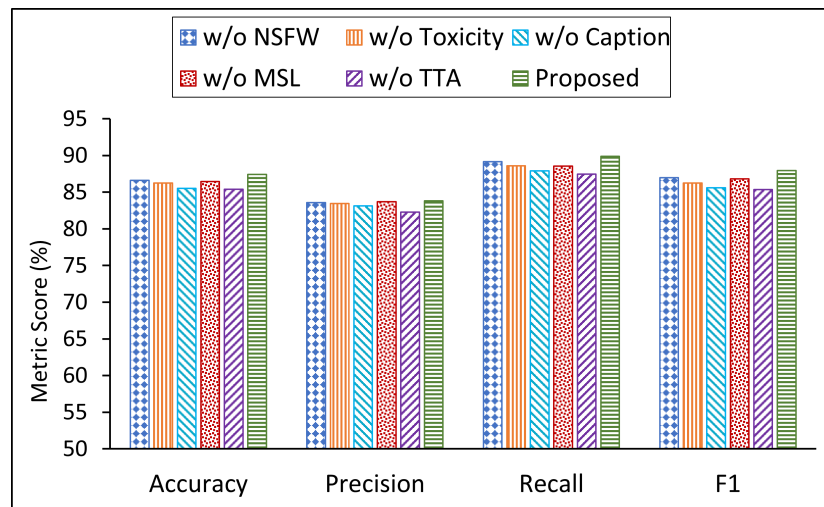


Figure 5.11: Influence of Additional Features

5.3.3 Qualitative Analysis

This section presents a qualitative analysis of the proposed framework for different memes. The analysis is based on its comparison with predictions of other multimodal methods by examining various aspects of the model's performance in relation to its counterparts. A pictorial representation of qualitative analysis is given in Figure 5.12.

Figure 5.12(a) presents multimodal content that has strong language against women. The content reads, "wife said she wanted space, sent her there with one punch" with a smiling man's image. The overall content depicts violence against women; hence, it is labeled as misogynous. Because of its strong language, all the methods predict it correctly.

In Figure 5.12(b), the image portrays a laughing lady, which conveys a positive emotion. The text in the image reads, "Marriage is just texting each other memes from different rooms," which contains humor and irony. Hence, the memes present humor without any offensive element against women. ViLT, for some reason, predicts it as misogynous; this is due to the fact that in a few training samples, the word marriage is associated with misogyny class. Therefore, ViLT is not able to detect the overall positive sentiment of the image. Most of the methods, including the proposed one, predict it correctly as non-misogynous.

Figure 5.12(c) shows a meme that depicts violence against women. The visual content shows a woman with a bruised face, giving a very negative sentiment, and the text conveys that she is hit due to something she does not agree to do. VisualBert and ViLT do not predict it correctly, while the others are able to capture the relationship between the text and image.

The content presented in Figure 5.12(d) reads, "a large group of karens is called, a homeowner' association" with visual content showing a lady with neutral expressions. Although the visual content does not depict any offense, however in the text, the word Karen is used as an offensive term against women. While most of the methods misclassify the content, the proposed method is able to predict it correctly as misogynous due to the incorporation of the MSL score that is computed based on slangs and misogynous words.

In conclusion, the qualitative analysis underscores the effectiveness of the proposed method in correctly predicting the presence of misogyny in memes. Through rigorous comparison with other model predictions, it becomes evident that the proposed framework demonstrates its strength by consistently and accurately identifying misogynistic content

Visual Modality	(a) 	(b) 	(c) 	(d) 
Textual Modality	Wife said she wanted space, sent her there with one punch	Marriage is just texting each other memes from different rooms	She didn't make a sandwich she deserved it	A large group of karen is called, a homeowner' association
Ground Truth	misogynous	Non-misogynous	misogynous	misogynous
Predictions	Late Fusion : misogynous VisualBert : misogynous CLIP : misogynous ViLT : misogynous BERT + ViT : misogynous Proposed : misogynous	Late Fusion : non-misogynous VisualBert : non-misogynous CLIP : non-misogynous ViLT : misogynous BERT + ViT : non-misogynous Proposed : non-misogynous	Late Fusion : misogynous VisualBert : non-misogynous CLIP : misogynous ViLT : non-misogynous BERT + ViT : misogynous Proposed : misogynous	Late Fusion : non-misogynous VisualBert : non-misogynous CLIP : non-misogynous ViLT : misogynous BERT + ViT : non-misogynous Proposed : misogynous

Figure 5.12: Qualitative Analysis of Misogyny Detection Method

across diverse content.

5.4 Conclusion

The proliferation of misogynistic and sexist content on social media has emerged as a noteworthy concern in recent times. In this article, we propose a novel multimodal framework for the detection of misogynistic and sexist content. Our proposed methodology involves feature learning via specialized modules, a novel context-aware attention module, a graph-based module for unimodal feature refinement, and a content-specific feature module. The final classification is achieved by integrating these features through a process of feature fusion, resulting in a unified representation. This joint feature vector is then processed through the classification network. Our findings indicate that context-aware attention helps get pertinent region-word features, resulting in a deeper understanding of the content. The graph-based method is able to comprehend and refine unimodal features. Moreover, we prepare a set of misogynistic lexicons to calculate the MSL score from the text that provides an initial

5.4 Conclusion

insight into the content. During testing, we apply Test-Time Augmentation with additive random perturbations, contributing to enhanced generalization and more robust predictions across diverse inputs. We have performed extensive experiments on two multimodal datasets, MAMI and MMHS150K. Results demonstrate our proposed method achieves competitive results by outperforming existing state-of-the-art methods.

The limitation of our approach is that it learns from labeled data. Supervised learning algorithms excel at specific tasks; however, a major challenge with supervised learning is the requirement for a consistent and large amount of annotated training data. Recent advancements in pretrained models offer a solution to the limitations of supervised learning. Pretrained models are trained on massive datasets with millions of labeled examples, often encompassing a wide range of tasks and concepts. This allows them to learn generalizable features and representations that can be transferred to new tasks with minimal additional training data. By using a pretrained model as a starting point, you only need to annotate a smaller portion of your specific data. This significantly reduces the time and cost associated with data labeling.

In conclusion, the proposed approach could be used in filtering misogynistic and sexist content. In the future, the approach can be extended for audio and video-based content and fine-grained hate speech classes can be explored instead of binary classification. Additionally, in the future, the approach can be extended for general hate speech irrespective of gender targetted.

Chapter 6

Multimodal-Multitask Method for Hostile Content Detection

6.1 Introduction

This chapter focuses on a multitask method for detecting hostility in multimodal content. Hostile intent in online discourse is rarely expressed through a single communicative signal and often emerges from the interplay of multiple affective and pragmatic cues. As a result, modeling hostile content as a *multitask* problem. Hence, jointly capturing sentiment, humor, sarcasm, offensiveness, and motivational intent, provides a more holistic understanding of user-generated content. Such a multimodal-multitask perspective is particularly effective in social media settings, where meaning is distributed across text, visuals, and latent emotional signals.

With the continuous evolution of online social media platforms, the practice of sharing multimodal content has become widespread [187, 188]. Multimodal content refers to content that is presented using multiple forms of media, such as text, images, audio, and video. Combinations of different modalities, for instance, memes that contain text superimposed on images, enable the users to articulate different emotions from various perspectives, ultimately enhancing the overall communication experience. Memes are frequently used as a means to communicate humor, sarcasm, motivation, or even expressions of hate through a combination of text and visual elements. These visual and textual modalities provide dis-

tinct perspectives on a subject, thereby complementing each other and playing a vital role in the overall comprehension [14, 15]. Due to its ability to convey emotions and opinions in a concise and easily shareable format, multimodal content is widely shared on social media.

The presence of an extensive collection of multimodal social media content has garnered attention toward the computational analysis of such content. Existing approaches have explored multimodal content within the field of sentiment analysis [189], as well as the realms of hateful content identification [13, 190], sarcasm detection [11] and humor recognition [12]. Multimodal content can convey a range of emotions; hence, there is a pressing requirement to address multiple cognitions, giving rise to the advancement of multimodal-multitask learning.

6.1.1 Existing Approaches and Challenges

Prior research has often focused on creating fusion techniques for acquiring multimodal features from unimodal features [191]. These fusion techniques encompass various approaches, including embedding fusion [192], transformer-based fusion [193], and attention-based fusion [194]. However, these techniques do not take into account the noise in monomodal features. Such noise, often arising from shared dominant components, anisotropy, or semantic redundancy, has been shown to obscure discriminative signals in embedding spaces [195]. We define this embedding-level noise as the presence of irrelevant, low-variance, or misaligned components that hinder effective multimodal fusion. This is distinct from raw input noise, which typically originates from modality-specific distortions such as background clutter in images or typographical errors in text; while raw noise affects the input data directly, embedding-level noise emerges during representation learning and interferes with cross-modal alignment. This inherent noise in monomodal representations influences cross-modal interactions and the overall multimodal representation [17], leading to inconsistent results. For instance, the meme displayed in the Figure 6.1 is very offensive, has a negative sentiment, and is not motivational. However, it is misclassified as not offensive, neutral, and motivational. This is particularly critical in multimodal architectures involving cross-modal interaction mechanisms, such as cross-attention or co-attention, where representations from one modality are used to condition or modulate representations from the other modality through operations such as dot-product similarity. In such cases, noise in one modality's em-

6.1 Introduction

bedding space can propagate and directly contaminate the other modality’s features due to the multiplicative nature of these interactions. Such instances can be dealt with by cross-modal learning without explicit interaction between the two modalities. Hence, it is imperative to explore multimodal feature representation methods that do not require feature interaction explicitly, reducing the influence of noise while enhancing the representations in a cross-modal fashion.

Earlier investigations have also established that one modality takes precedence in multimodal learning, with other modalities serving more as supplementary components [196]. The existing fusion methods used to generate multimodal features lead to the oversight of discriminative information in monomodal features. Moreover, it is worth noting that memes have text superimposed on images, and this text sometimes lies on the important regions of the image, which hampers the extraction of pertinent features and creates noisy features. This necessitates removing text from the images in order to get refined and less noisy features. In the existing works, data augmentation has proven highly successful in enhancing model performance across various computer vision and natural language processing tasks [8]. However, the recent state-of-the-art generative AI-based large language models (LLMs) have not yet been explored for multimodal-multitask learning.

6.1.2 Navigating Our Approach: Proposed Solution at a Glance

In light of the aforementioned challenges, we propose a hierarchical interactive attention-aware multimodal-multitask framework for semantic comprehension with cross-modal relation learning. The proposed method learns latent multimodal features in a cross-modal manner without explicit interaction between different modality representations. This is achieved by constructing cross-modal relation graphs where nodes are created based on the features of one modality, and the edges between nodes are established based on the features of another modality. This method effectively reduces the impact of monomodal feature noise during cross-modal feature updates. The superimposed text region of a meme undergoes inpainting before feature extraction, effectively reducing the introduction of noise. We propose a **Hierarchical Interactive Monomodal Attention (HIMA)** mechanism. At the first level, HIMA generates region-based and word-level attention vectors; subsequently, at the next level, it creates a unified representation for the batch of each modality. These representa-

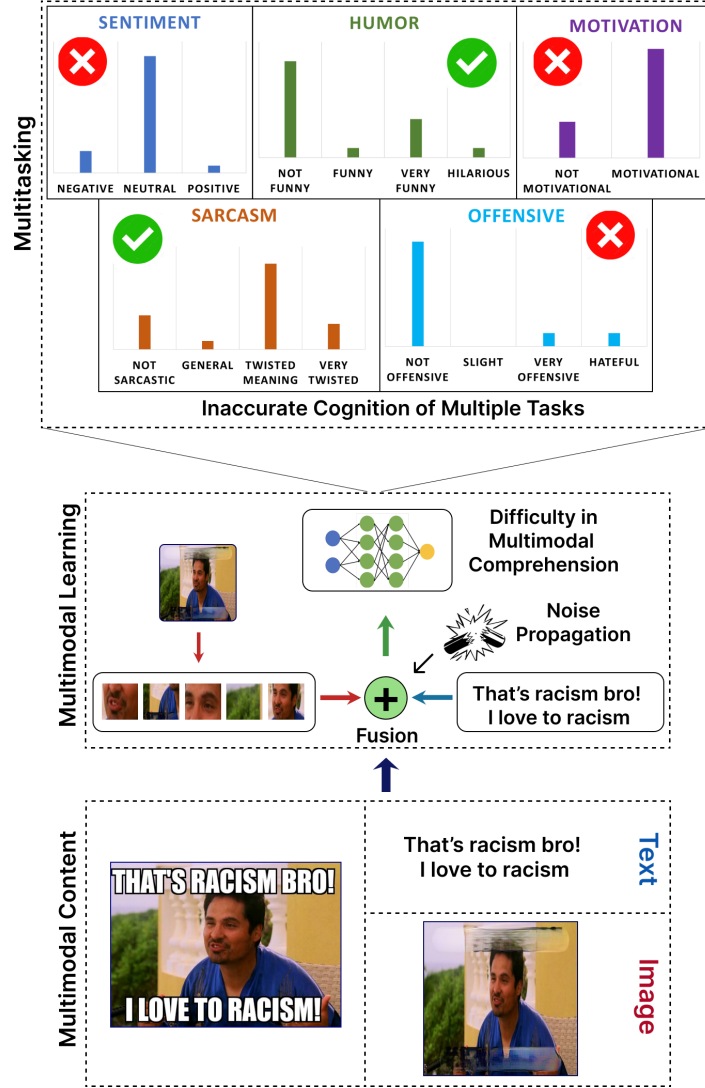


Figure 6.1: Multimodal Machine Cognition and Noise Propagation

tions are late-fused to generate a joint representation. Cross-modal relation graphs help in multimodal comprehension, whereas the proposed attention mechanism benefits multitask learning. Additionally, we employ generative AI-based LLM for text augmentation. Lastly, task-specific text features are added in the final feature representation. The framework is primarily designed and evaluated on image-text multimodal content. However, its modular approach may be used to extend it for other modalities too.

6.1.3 Key Contributions

To sum up, our key contributions can be summarized as follows:

1. We propose a novel framework for semantic comprehension of multimodal content for multiple tasks, such as sarcasm detection, humor recognition, offensive content detection, motivation detection, and sentiment classification.
2. We design a novel feature reconstruction approach for semantic multimodal comprehension via cross-modal relation graphs. This approach does not necessitate a direct interaction between distinct modality representations, leading to a reduction in noise propagation in the latent multimodal features.
3. We introduce a Hierarchical Interactive Monomodal Attention (HIMA) to benefit multitasking. HIMA is applied on monomodal features in two steps: first, on the individual images and text, and later, on the batch of multimodal content. Unlike traditional Hierarchical Attention Network (HAN) [197], which is applied on the document level and where the second stage vector (sentence vector) serves as the classification vector, HIMA utilizes both levels of vectors and finally integrates two modalities to generate a joint representation.
4. We evaluate the performance of the proposed method on three multimodal datasets. Quantitative and qualitative experiments conducted for five tasks provide confirmation of the efficacy of the proposed approach.

6.2 Problem Definition and Objectives

The primary challenges arise from the complex interplay between text and images. Furthermore, the multitask nature of the problem introduces additional complexity, as each meme may convey multiple emotions simultaneously.

6.2.1 Main Problem

Given a dataset with N samples, $D = \{(X_i, Y_i)\}_{i=1}^N$, where (X_i, Y_i) represents a multimodal sample with aligned text and image modalities $X = (x_i^{\text{txt}}, x_i^{\text{img}})$ and associated labels $Y_i \in \{y_1, y_2, \dots, y_n\}$ where each y_i denotes a single task. Our goal is to build a unified framework F that effectively comprehends the multimodal content X_i and outputs the multitask labels Y_i as follows:

$$F(X_i) = Y_i \quad (6.1)$$

$$\text{s.t. } F(x_i^{\text{txt}}, x_i^{\text{img}}) = (y_1, y_2, \dots, y_n) \quad (6.2)$$

We define two major subproblems as our objectives, each addressing a key aspect of our framework.

6.2.2 Objective 1: Cross-modal Feature Interaction

Given $E_i^{m_1}$ and $E_i^{m_2}$, denoting the features extracted from modalities m_1 and m_2 respectively, where $m_1, m_2 \in \{\text{txt}, \text{img}\}$, our objective is to generate refined feature representations for a given task using cross-modal relation graphs $\mathcal{G}_{\text{cross}}(m_1, m_2)$ wherein $m_1 \neq m_2$, nodes correspond to features of the m_1 modality, while edges are derived from the features of the m_2 modality. This approach is designed to mitigate the impact of monomodal feature noise, thereby enhancing the representation of multimodal data. Mathematically, this can be expressed as:

$$\mathcal{R}_i^{m_1} = \begin{cases} f(E_i^{\text{txt}}, \mathcal{G}_{\text{cross}}(\text{txt}, \text{img})), m_1 = \text{txt} \\ f(E_i^{\text{img}}, \mathcal{G}_{\text{cross}}(\text{img}, \text{txt})), m_1 = \text{img} \end{cases} \quad (6.3)$$

where $\mathcal{R}_i^{m_1}$ represents the refined feature representation of the i -th sample in m_1 modality, f is the function that generates refined feature representations for $E_i^{m_1}$ based on $\mathcal{G}_{\text{cross}}(m_1, m_2)$.

6.2.3 Objective 2: Hierarchical Attention-based Feature Interaction

For a given modality $m \in \{\text{txt}, \text{img}\}$, our objective is to develop a function $\mathcal{H}^m(Q_i^m, \gamma^m)$ that effectively harnesses the monomodal features at different levels of granularity to produce an enhanced joint feature representation for a given sample. Here, $m \in \{\text{txt}, \text{img}\}$ represents the modality. Mathematically, this can be formulated as:

$$\mathcal{H}^m = \begin{cases} \mathcal{H}^{\text{txt}}(f_{\text{word}}(Q_i^{\text{txt}}), f_{\text{sentence}}(\gamma^{\text{txt}})), m = \text{txt} \\ \mathcal{H}^{\text{img}}(f_{\text{region}}(Q_i^{\text{img}}), f_{\text{image}}(\gamma^{\text{img}})), m = \text{img} \end{cases} \quad (6.4)$$

where $Q_i^m \in \gamma^m$ represents the feature matrix of m modality associated with the i -th sample, and γ^m is a batch of feature matrices corresponding to m modality. The functions $\mathcal{H}^{\text{txt}}$ and $\mathcal{H}^{\text{img}}$ process the outputs of the attention mechanisms. $f_{\text{word}}(Q_i^{\text{txt}})$, $f_{\text{sentence}}(\gamma^{\text{txt}})$, $f_{\text{region}}(Q_i^{\text{img}})$, and $f_{\text{image}}(\gamma^{\text{img}})$ are attention mechanisms that operate at different levels of granularity within each modality, with their corresponding levels mentioned in the subscripts.

6.3 Methodology

In this section, we describe the methodology of our proposed multimodal-multitask framework, MM-ORIENT, for semantic comprehension. As depicted by the Figure 6.2, the proposed framework first preprocesses the images and text and then extracts the features for the HIMA and Cross-modal Relation Learning (CMRL) modules. Finally, a joint representation of the features is passed through the learner network for emotion detection. Subsequent subsections elaborate on the proposed approach.

6.3.1 Data Preprocessing

Data preprocessing entails two components: Image Masking & Inpainting and Text Cleaning. Image masking & inpainting ensures the removal of overlaid text to emphasize visual elements, while text cleaning standardizes text format for accurate extraction.

6.3.1.1 Image Masking & Inpainting

Memes often contain humorous or contextual text overlaid on the image. This text can introduce noise and interfere with the feature extraction process. To eliminate the text from image, we utilize masking and inpainting techniques as shown in Figure 6.2. This ensures that the primary focus is on the underlying visual content of the image, allowing for more precise and pertinent feature representations.

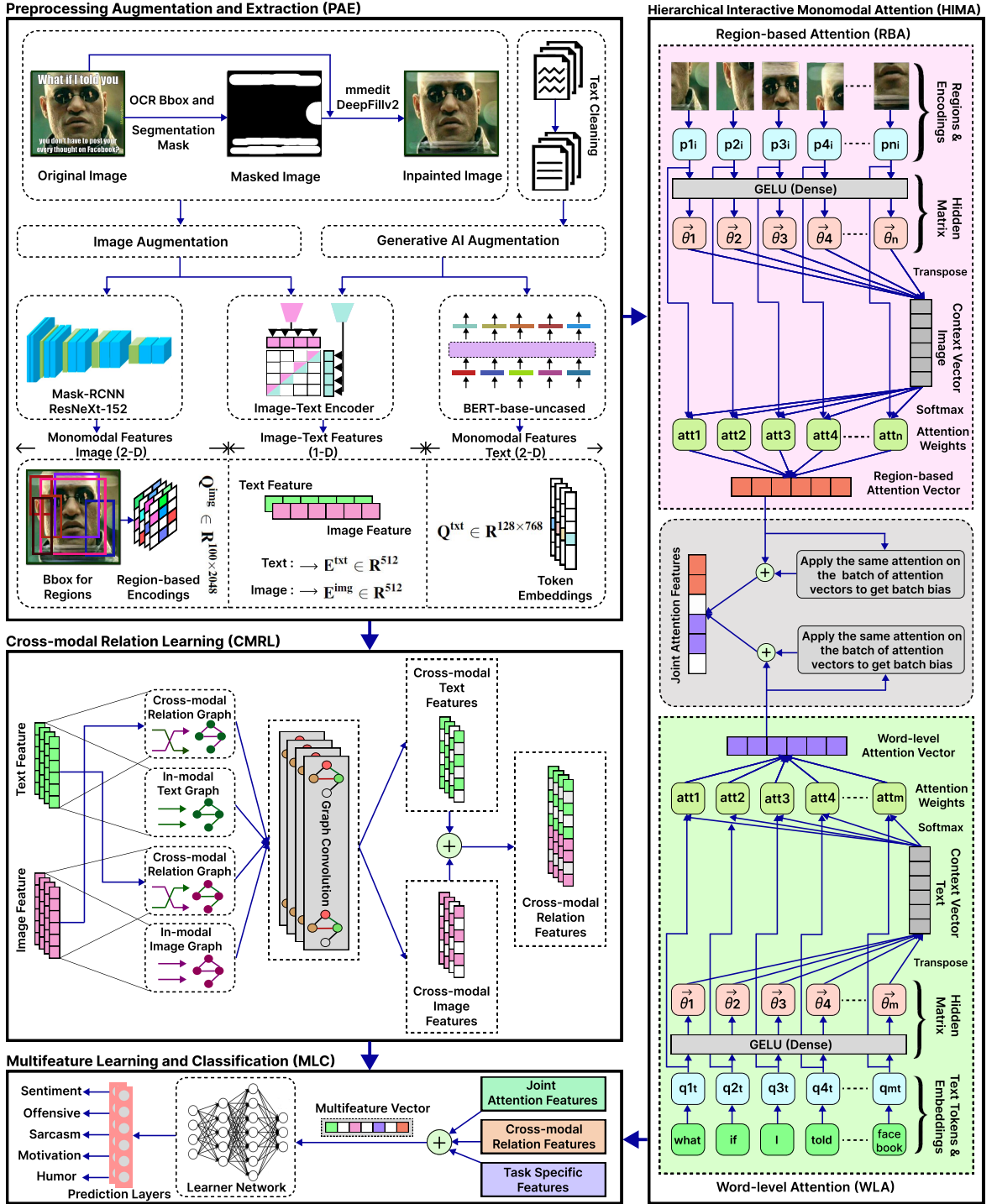


Figure 6.2: Overall Architecture of MM-ORIENT

Masking involves the creation of a binary image mask, separating text from the underlying visual content, and enabling the removal of text-containing regions in the image. This allows for subsequent inpainting of the text-removed regions that involves generating substitute elements in areas that are absent, ensuring that the alteration appears both visually

convincing and semantically accurate. We employ a generative image inpainting technique based on gated convolutions proposed by Yu et al. [158] to produce the absent sections within masked images, thereby achieving a natural appearance. By using masking in conjunction with inpainting techniques, we are able to remove the text from the image entirely.

6.3.1.2 Text Cleaning

Text cleaning encompasses tasks like purging undesirable characters, symbols, and excessive white spaces, as well as standardizing its case. If URLs or usernames are present in the image, they are removed to ensure only relevant textual content is retained. This step is essential to ensure that the extracted text is in a clean and standardized format, which in turn improves the quality of extracted features.

While these preprocessing steps help improve input quality, the core contribution of our work lies in mitigating structural noise in the learned feature space, which impacts multi-modal representation learning.

6.3.2 Data Augmentation

Data augmentation enriches our dataset with diverse variations and increases the minority class samples, thus enhancing the adaptability of our models for finer interpretation of multi-modal content. For each image and text in the training data, we generate a few more similar images and text. These new images and text are added to the training data before the training of the model.

6.3.2.1 Image Augmentation

For image augmentation, we strategically manipulate image attributes by subjecting each input image to a randomized transformation process. This involves horizontal flipping, which mirrors the image along the vertical axis, and adjustments to brightness, contrast, hue, and saturation. The range of these adjustments is carefully selected to introduce meaningful variations while preserving the underlying content. Hence, each image is transformed using one of these transformations that are selected randomly. The resultant image is added to the training set and is given the same label as the original. The text of the given image is also

transformed using text augmentation.

6.3.2.2 Text Augmentation

For text augmentation, we harness the capabilities of OpenAI’s state-of-the-art language model, GPT-3.5-turbo, using a technique known as few-shot prompting. This method allows us to provide specific examples that guide the model’s responses in generating alternative versions of sentences. By framing the task as a conversational exchange, we prompt the system to produce several distinct rephrased renditions for each original sentence. We prepare the following prompt for the GPT-3.5; *{“role”: “system”, “content”: “You are a helpful, pattern-following assistant that removes the URLs and usernames from sentence, and then rephrases the sentence in 5 different forms”}*. Subsequently, sentences are provided to the GPT-3.5 which returns sentences in different words without losing the meaning of the text. This approach enables us to diversify our dataset with a curated set of linguistically diverse alternatives, ultimately enhancing the richness and variability of our training corpus.

6.3.3 Feature Extraction

Within our framework, we incorporate various feature extraction techniques for image and text modalities. Additionally, we introduce supplementary features like emotion, sentiment, and toxicity-related attributes derived from the text.

6.3.3.1 Feature Extraction for CMRL

We utilize the pre-trained CLIP [160] encoder for extracting features from datasets having both image and text modalities in a unified manner. Unlike traditional methods that focus on either images or text separately, it is trained to learn representations that can bridge the gap between the two modalities. It learns to associate images with their corresponding text, allowing it to generate meaningful representations for both modalities altogether. The encoder processes the image-text pair X_i and produces image embeddings $E^{\text{img}} \in \mathbb{R}^{512}$ and text embeddings $E^{\text{txt}} \in \mathbb{R}^{512}$.

$$E^{\text{img}}, E^{\text{txt}} = \overrightarrow{\text{CLIP}}(X_i) \quad (6.5)$$

Compared to other multimodal models, CLIP produces embeddings with lower dimensionality. The efficiency of CLIP in generating embeddings quickly and at scale aligns well with the requirements of GraphSAGE, which benefits from rapid and scalable embedding computations making it an ideal choice.

6.3.3.2 Feature Extraction for HIMA

We employ BERT [100] for textual feature extraction. The `bert-base-uncased` model is used for tokenization, breaking down raw text into tokens. For each sample in the training data, the tokenized input is passed through BERT. The model is configured with a maximum sequence length of 128 tokens, enabling truncation and padding for uniform sizes. The BERT model processes the set of tokens $\{w_i\}_{i=1}^{128}$, producing the feature matrix $Q^{\text{txt}} = \{q_i\}_{i=1}^{128}$, where $q_i \in \mathbb{R}^d$ represents the token embedding of the i -th token and d denotes the dimension of token embedding. For $m = \text{txt}$, $d = 768$.

$$q_i = \overrightarrow{\text{BERT}}(w_i) \quad (6.6)$$

For the visual modality, we employ a region-based image feature extraction approach `detectron2` [161] utilizing a Mask RCNN [176] model with ResNeXt-152 as the backbone network. We refer to this model as Mask RCNN X152. Each image is divided into 100 regions $\{r_i\}_{i=1}^{100}$, from each of which a feature vector $p_i \in \mathbb{R}^d$ is extracted where $d = 2048$. These features are derived from the `frc6` layer of the ResNeXt-152 backbone network. The resulting visual feature matrix is denoted as $Q^{\text{img}} = \{p_i\}_{i=1}^{100}$.

BERT has achieved state-of-the-art results on a wide range of NLP benchmarks, demonstrating its effectiveness in capturing linguistic insights and improving the performance of downstream tasks such as text classification. `Detectron2` generates precise region proposals, identifying multiple objects within an image and delineating their boundaries accurately. This is particularly useful for extracting detailed region-specific features. `Detectron2` extracts rich, multi-scale features from images, capturing fine details and high-level semantics.

$$p_i = \overrightarrow{\text{MRCNN-X152}}(r_i) \quad (6.7)$$

6.3.3.3 Task Specific Features

We incorporate emotion features, sentiment features, and toxicity-related features to enhance the text analysis capabilities of MM-ORIENT. These features provide a comprehensive understanding of the underlying sentiment and emotional insights within the text samples. To extract emotion features, each word in the text is mapped to its underlying emotion category {fear, anger, anticipation, trust, surprise, positive, negative, sadness, disgust, joy}. We then obtain a feature vector for the given text sample, where each element corresponds to the frequency of a specific emotion category¹.

Furthermore, we retrieve sentiment values for each text sample. The sentiment categories span from 0 to 4, encompassing Very Negative, Negative, Neutral, Positive, and Very Positive sentiment categories [198]. To extract toxicity-related features, we leverage the RoBERTa Toxicity Classifier², which is trained to classify the text in toxic and not-toxic classes. A 768-dimensional feature vector is extracted for toxicity elements.

6.3.4 Hierarchical Interactive Monomodal Attention

Hierarchical attention-based feature interaction has proved to be very effective in monomodal as well as multimodal tasks [187, 197]. Understanding the intricate and subtle interplay of words within text modality and regions in image modality is paramount for accurately discerning the contextual meaning of a given meme. To this end, we employ word-level attention (WLA) and region-based attention (RBA) mechanisms designed to emphasize pivotal words and regions within the text and image modality, respectively, enabling the model to synthesize a comprehensive understanding of memes, capturing both linguistic subtleties and visual cues for a more accurate interpretation of their intended message. Algorithm 6.1 shows the implementation of HIMA.

Suppose we have $\gamma^m = \{Q_y^m\}_{y=1}^B$ which denotes a batch of 2-dimensional features ex-

¹<https://pypi.org/project/NRCLex/>

²https://huggingface.co/s-nlp/roberta_toxicity_classifier

Algorithm 6.1 Hierarchical Interactive Monomodal Attention

Input: k : Index of a sample in the batch
 γ^m : Feature batch containing k -th sample
Output: z_k^m : Multi-level contextual vector for k -th sample

```
1: function HIERARCHICAL_ATTENTION( $k, \gamma^m$ )
2:   for all  $Q_y^m \in \gamma^m$  do
3:     for all  $v_x^m \in Q_y^m$  do
4:        $\theta_x^m \leftarrow \text{GELU}(v_x^m W_1^m + b_1^m)$ 
5:     end for
6:      $\text{hid\_rep1}_y^m \leftarrow \{\theta_x^m\}_{x=1}^L$ 
7:     for all  $\theta_x^m \in \text{hid\_rep1}_y^m$  do
8:        $\text{att}_x^m \leftarrow \text{softmax}((\theta_x^m)^T u_1)$ 
9:     end for
10:     $t_y^m \leftarrow \sum_{x=1}^L \text{att}_x^m v_x^m$ 
11:  end for
12:   $T^m \leftarrow \{t_y^m\}_{y=1}^B$ 
13:  for all  $t_y^m \in T^m$  do
14:     $l_y^m \leftarrow \text{GELU}(t_y^m W_2^m + b_2^m)$ 
15:  end for
16:   $\text{hid\_rep2}_y^m \leftarrow \{l_y^m\}_{y=1}^B$ 
17:  for all  $l_y^m \in \text{hid\_rep2}_y^m$  do
18:     $p_y^m \leftarrow \text{softmax}((l_y^m)^T u_2)$ 
19:  end for
20:   $s^m \leftarrow \sum_{y=1}^B p_y^m t_y^m$ 
21:   $z_k^m \leftarrow \text{concat}(t_k^m, s^m)$ 
22:  return  $z_k^m$ 
23: end function
```

tracted from $m \in \{\text{txt}, \text{img}\}$ modality, where $Q_y^m = \{v_x^m\}_{x=1}^L$ represents the y -th feature matrix of m modality that corresponds to a single sample in the batch, and $B = 128$ denotes the batch size. Here, $v_x^m \in \mathbb{R}^d$ represents the word/region embedding vector. For $m = \text{txt}$, v_x^m represents the word embedding of the x -th word in the sequence, and L denotes the sequence length. For $m = \text{img}$, v_x^m represents the embedding vector of the x -th bounding box or region in the image, and L denotes the total number of regions.

6.3.4.1 Word-level and Region-based Attention (stage 1)

Both WLA and RBA follow the same procedure. First, we compute the hidden representation θ_x^m of v_x^m through the transformation using the Gaussian Error Linear Unit (GELU), which provides a smooth approximation to the Rectified Linear Unit (ReLU) while maintaining smooth gradients:

$$\theta_x^m = \frac{1}{2} (v_x^m W_1^m + b_1^m) \left(1 + \text{erf} \left(\frac{v_x^m W_1^m + b_1^m}{\sqrt{2}} \right) \right) \quad (6.8)$$

where erf is the error function, the W is the weight parameter and b is the bias. $W_1^m \in \mathbb{R}^{768 \times \beta}$ and $b_1^m \in \mathbb{R}^\beta$ are fine-tuned during training. Here, β denotes the output dimension of the linear transformation. We evaluate the significance of the x -th word/region by calculating the similarity between θ_x^m and a word/region-level context vector u_1 . Subsequently, we obtain the attention weights att_x^m using a softmax function:

$$\text{att}_x^m = \frac{\exp((\theta_x^m)^T u_1)}{\sum_{x=1}^L \exp((\theta_x^m)^T u_1)} \quad (6.9)$$

These attention weights serve as dynamic coefficients, emphasizing words for $m = \text{txt}$ and regions for $m = \text{img}$ that carry heightened contextual significance, enabling the model to allocate greater importance to semantically relevant elements during the aggregation process. Finally, the WLA vector t_y^{txt} is computed, aggregating the contributions of individual words, and the RBA vector t_y^{img} is computed, aggregating the contributions of individual regions, based on their respective attention scores:

$$t_y^m = \sum_{x=1}^L att_x^m v_x^m \quad (6.10)$$

6.3.4.2 Sentence-level and Image-level Attention (stage 2)

Similarly, as we calculated t_y^m to capture significant cues at the word/region level, we extend our attention mechanism to a higher level to capture batch-wise contextual subtleties at the sentence/image level by introducing a sentence/image-level context vector u_2 . Given a batch of WLA/RBA vectors $T^m = \{t_y^m\}_{y=1}^B$, where t_y^m represents the WLA/RBA vector of the y -th sample in the batch, we calculate the attention vector s^m in a similar manner as shown in Equations 6.8-6.10 with the only difference being that we use $t_y^m \in T^m$ instead of $v_x^m \in Q_y^m$.

$$l_y^m = \frac{1}{2} (t_y^m W_2^m + b_2^m) \left(1 + \operatorname{erf} \left(\frac{t_y^m W_2^m + b_2^m}{\sqrt{2}} \right) \right) \quad (6.11)$$

$$p_y^m = \frac{\exp((l_y^m)^T u_2)}{\sum_{y=1}^B \exp((l_y^m)^T u_2)} \quad (6.12)$$

$$s^m = \sum_{y=1}^B p_y^m t_y^m \quad (6.13)$$

We refer to this attention vector s^m as the batch bias that summarizes the information of all the sentences in the batch (for $m = \text{txt}$) or all the images in the batch (for $m = \text{img}$). In order to obtain a combined feature representation z_k^m for the k -th sample in the batch, that encapsulates a multi-level or hierarchical contextual understanding of m modality, we concatenate t_k^m and s^m :

$$z_k^m = \operatorname{concat}(t_k^m, s^m) \quad (6.14)$$

Finally, a joint attention feature vector Z_k for the k -th meme is obtained by late-fusing z_k^m obtained from both modalities:

$$Z_k = \text{concat}(z_k^{\text{txt}}, z_k^{\text{img}}) \quad (6.15)$$

The *for* loops in lines 2-3 of Algorithm 6.1 will run for $B \times L$ iterations in total. In line 4, there is a matrix multiplication operation between two matrices with shapes $1 \times d$ and $d \times \beta$ contributing $O(d \times \beta)$. The remaining steps do not significantly impact the overall time complexity. Thus, the overall time complexity of Algorithm 6.1 is $O(B \times L \times d \times \beta)$.

6.3.5 Cross-modal Relation Learning

The idea behind CMRL is to learn latent multimodal features through a cross-modal approach, without the need for direct interaction between representations of different modalities at the early stage. In order to accomplish this, we propose cross-modal relation graphs, which reconstruct monomodal features to obtain multimodal representations. The reconstruction of features relies on the node neighborhood, with the neighborhood determined by the features of a distinct modality. Algorithm 6.2 shows the implementation of CMRL. The subsequent sections elaborate on the graph construction and feature reconstruction process.

6.3.5.1 Graph Construction

The construction of the graph involves cosine similarity computation and adjacency matrix formation.

Given an embedding matrix $X^m = \{E_i^m\}_{i=1}^\lambda$ such that $X^m \in \mathbb{R}^{\lambda \times D}$, where $m \in \{\text{txt}, \text{img}\}$ is the modality, λ is the total number of embeddings, and $D = 512$ is the embedding dimension, the embeddings are first normalized using L2 normalization to obtain a matrix $\text{norm}_X^m = \{\mathcal{E}_i^m\}_{i=1}^\lambda$:

$$\text{norm}_X^m = \frac{X^m}{\|X^m\|_2} \quad (6.16)$$

Algorithm 6.2 Cross-modal Relation Learning

Input: k : Index of a sample in the batch
 X^{txt} : Text embedding matrix
 X^{img} : Image embedding matrix
 thr : Similarity threshold

Output: H_k : Cross-modal relation feature vector

```

1: function GRAPH_LEARNING( $k, X^{\text{txt}}, X^{\text{img}}, threshold$ )
2:   for all  $m \in \{\text{txt}, \text{img}\}$  do
3:      $norm\_X^m = X^m / \|X^m\|_2$ 
4:      $S^m \leftarrow \{\text{cs}(\mathcal{E}_i^m, \mathcal{E}_j^m) \mid \forall \mathcal{E}_i^m, \mathcal{E}_j^m \in norm\_X^m\}$ 
5:      $A^m \leftarrow \{A_{ij}^m \mid A_{ij}^m = 1 \text{ if } S_{ij}^m \geq threshold \text{ else } 0\}$ 
6:   end for
7:    $G \leftarrow \{\}$ 
8:   for all  $m_1 \in \{\text{txt}, \text{img}\}$  do
9:     for all  $m_2 \in \{\text{txt}, \text{img}\}$  do
10:       $G.add(\mathcal{G}(m_1, m_2))$ 
11:    end for
12:  end for
13:  for all  $\mathcal{G} \in G$  do
14:     $N \leftarrow \{V_i \mid V_i \text{ is a neighbour of the } k\text{-th node in } \mathcal{G}\}$ 
15:     $sum \leftarrow \sum_{V_i \in N} E_i^{m_1}$ 
16:     $M \leftarrow sum / |N|$ 
17:     $H_k^{m_1-m_2} \leftarrow \text{concat}(M, E_k^{m_1}) \cdot W + b$ 
18:  end for
19:   $H_k \leftarrow \text{concat}(H_k^{\text{img-img}}, H_k^{\text{txt-txt}}, H_k^{\text{img-txt}}, H_k^{\text{txt-img}})$ 
20:  return  $H_k$ 
21: end function

```

Here, $\|X^m\|_2$ represents the L2 norm of the embeddings, computed along axis 1, ensuring that each embedding vector has a unit length, $norm_X^m$ denotes the normalized embeddings matrix, and $\mathcal{E}_i^m$ represents a single normalized embedding.

Next, the similarity matrix S^m is computed using cosine similarity:

$$S^m = \frac{norm_X^m \cdot norm_X^m}{\|norm_X^m\|_2 \cdot \|norm_X^m\|_2} \quad (6.17)$$

Here, S^m represents the cosine similarity between normalized embeddings. In the algorithm, “cs” denotes the cosine similarity function. The resulting similarity matrix S^m is a square matrix of size $\lambda \times \lambda$. Finally, a binary adjacency matrix A^m is derived by thresholding S^m at a specified *threshold*, indicating the presence or absence of edges between nodes:

$$A_{ij}^m = \begin{cases} 1, & \text{if } S_{ij}^m > threshold \\ 0, & \text{otherwise} \end{cases} \quad (6.18)$$

The value of A_{ij}^m indicates whether there is an edge between the i -th and j -th nodes, with 1 indicating the presence of an edge and 0 denoting its absence.

We employ in-modal and cross-modal relations by constructing a set of four graphs $G = \{\mathcal{G}(m_1, m_2)\}$ using all possible combinations of m_1 and m_2 , where $m_1, m_2 \in \{\text{txt}, \text{img}\}$. $\mathcal{G}(m_1, m_2)$ represents a graph in which the edges are determined using the adjacency matrix A^{m_2} and the nodes represent the embedding vectors of m_1 modality. The graph $\mathcal{G}(m_1, m_2)$ is constructed such that the nodes are derived from modality m_1 , while the edges are determined using the cosine similarity between features of modality m_2 . For instance, in $\mathcal{G}(\text{img}, \text{txt})$, the image features are treated as nodes, but the edge connections between them are computed based on the similarity between corresponding text features. The core motivation is to leverage one modality to provide a relational prior for the other. For example, two visually dissimilar images might be semantically linked if their corresponding texts convey similar meanings (e.g., “a dog playing” and “a pup running”). The proposed cross-modal graphs encode such indirect semantic relationships by using one modality (e.g., text) to de-

fine edge weights among nodes in another modality (e.g., image), thereby enhancing the semantic expressiveness of learned features. We get two graphs representing in-modal relations ($m_1 = m_2$), and two graphs representing cross-modal relations ($m_1 \neq m_2$) as shown in Figure 6.3 and Figure 6.4, respectively. In the in-modal graphs, edges are computed based on the same modality as the nodes. For instance, if we are constructing an in-modal graph of text, then nodes are the text features, and edges are based on the similarity of the text features. Based on our experiments, we set the threshold values as 0.7 and 0.8 for constructing $\mathcal{G}(\text{txt}, \text{txt})$ and $\mathcal{G}(\text{img}, \text{img})$, respectively. Similarly, we set a threshold of 0.85 for $\mathcal{G}(\text{txt}, \text{img})$ and 0.75 for $\mathcal{G}(\text{img}, \text{txt})$.

The mapping in cross-modal learning is meaningful in the context of memes, where strong structural patterns exist across samples. In particular, it is common for a single or similar image template to be reused with different textual captions, and for a single or similar text phrase to appear alongside different images

Notably, if a particular image template is used to convey hate in one meme, it is often reused with similar intent in other memes. The same holds true for motivational and humorous content. Likewise, text phrases associated with hate tend to cluster together in the semantic embedding space, as do motivational or humorous expressions. This behavior results in a latent semantic structure across modalities, where memes sharing thematic or emotional intent are embedded close to each other, even across different modalities.

Our model leverages this phenomenon by defining edge connections in one modality based on similarity in the other. For example, if two text embeddings are semantically similar, their associated image nodes are connected in the image graph. This allows the model to exploit natural multimodal clusters during graph-based message passing. This approach enhances the robustness of multimodal representations.

6.3.5.2 Feature Reconstruction

Reconstruction of visual and textual features involves updating nodes by leveraging local neighborhood information via inductive representation learning. Inductive learning methods offer the benefit of generating embeddings for nodes or subgraphs that have not been previously seen. To employ inductive learning, we utilize GraphSAGE [171] that operates on node features and utilizes a sparse adjacency matrix to aggregate information from a node's

neighborhood, enhancing the model's ability to capture complex relationships within the graph. In the model architecture, the adjacency matrices A^{img} and A^{txt} are utilized as sparse

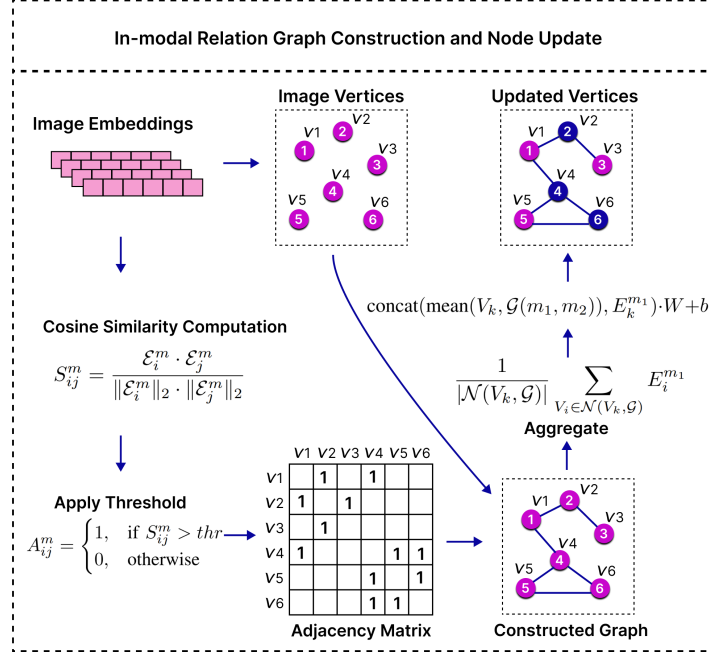


Figure 6.3: In-modal Relation Graph construction and Node Update

inputs. The aggregation process is performed using the mean function. This function calculates the average of the neighboring nodes' features for each node, effectively creating a summarization of the local neighborhood information. The mean value for the k -th node V_k of a graph $\mathcal{G}(m_1, m_2)$ is calculated as follows:

$$\text{mean}(V_k, \mathcal{G}) = \frac{1}{|\mathcal{N}(V_k, \mathcal{G})|} \sum_{V_i \in \mathcal{N}(V_k, \mathcal{G})} E_i^{m_1} \quad (6.19)$$

where $\mathcal{N}(V_k, \mathcal{G})$ denotes the set of neighboring nodes for V_k , V_i denotes the i -th node in $\mathcal{G}(m_1, m_2)$ corresponding to the embedding vector $E_i^{m_1} \in X^{m_1}$.

This aggregate neighborhood information is then combined with the original features $E_k^{m_1}$ corresponding to V_k in a concatenation operation, forming a comprehensive feature representation for further processing.

Following the aggregation step, the combined features undergo a linear transformation utilizing learned weights $W \in \mathbb{R}^{1024 \times C}$ and biases b . Here, $C = 512$ denotes the number of

6.3 Methodology

output channels in the GraphSageConv operation. This transformation enables the model to learn intricate relationships within the graph structure. Furthermore, to maintain consistency and control the scale of the output, the features are subjected to L2 normalization. Mathematically, the GraphSageConv operation $\mathcal{Y}$ applied on the k -th sample can be summarized as follows:

$$\mathcal{X}(k, X^{m_1}, A^{m_2}) = \text{concat}(\text{mean}(V_k, \mathcal{G}(m_1, m_2)), E_k^{m_1}) \cdot W + b \quad (6.20)$$

$$\mathcal{Y}(k, X^{m_1}, A^{m_2}) = \frac{\mathcal{X}(k, X^{m_1}, A^{m_2})}{\|\mathcal{X}(k, X^{m_1}, A^{m_2})\|_2} \quad (6.21)$$

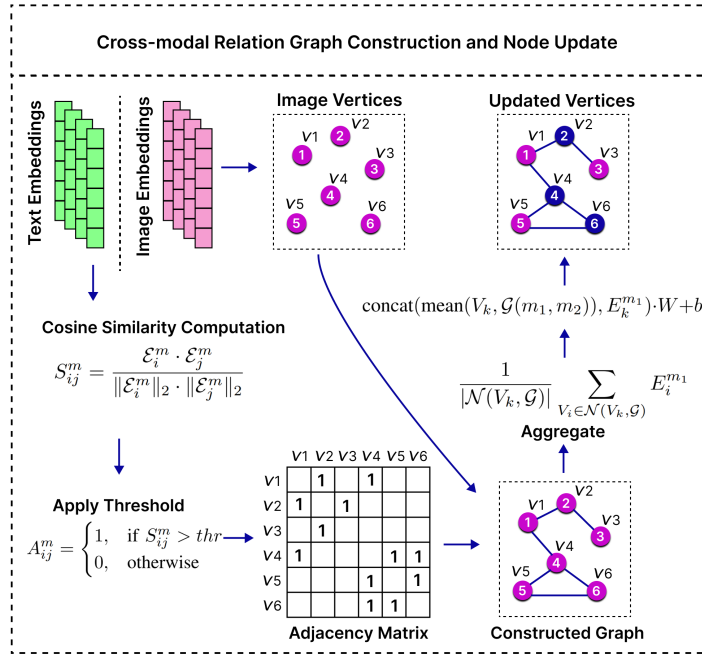


Figure 6.4: Cross-modal Graph Construction and Node Update

We apply this operation to both the in-modal relation graphs, as represented in Equations 6.22 and 6.23, and both the cross-modal relation graphs, as represented in Equations 6.24 and 6.25, constructed for a given meme. This allows us to obtain embedding vectors $H_k^{\text{img-img}}$, $H_k^{\text{txt-txt}}$, $H_k^{\text{img-txt}}$, $H_k^{\text{txt-img}}$. Finally, we concatenate all these vectors to form a cross-modal relation feature vector denoted as H_k , as shown in Equation 6.26.

$$H_k^{\text{img-img}} = \mathcal{Y}(k, X^{\text{img}}, A^{\text{img}}) \quad (6.22)$$

$$H_k^{\text{txt-txt}} = \mathcal{Y}(k, X^{\text{txt}}, A^{\text{txt}}) \quad (6.23)$$

$$H_k^{\text{img-txt}} = \mathcal{Y}(k, X^{\text{img}}, A^{\text{txt}}) \quad (6.24)$$

$$H_k^{\text{txt-img}} = \mathcal{Y}(k, X^{\text{txt}}, A^{\text{img}}) \quad (6.25)$$

$$H_k = \text{concat} \left(H_k^{\text{img-img}}, H_k^{\text{txt-txt}}, H_k^{\text{img-txt}}, H_k^{\text{txt-img}} \right) \quad (6.26)$$

Figure 6.3 illustrates the construction and updating of the in-modal relation graphs, while Figure 6.4 depicts the construction and updating of the cross-modal relation graphs.

The proposed cross-modal relation learning utilizes sparse graphs, where edges are based on the cosine similarity threshold of other modality. In contrast, the traditional cross-attention-based fusion mechanism can be considered as a fully connected graph in a way that involves computing attention weights between every pair of nodes across different modalities. This means that each node attends to every other node, forming connections that can be represented as edges in a graph. The proposed CMRL method further uses the GraphSAGE algorithm to enhance feature representation, which adds to its run time complexity. However, the objective of CMRL is to reduce the impact of noise during the computation of multimodal features, trading off runtime complexity to enhance overall performance. The traditional cross-attention mechanism uses queries and keys of different modalities to calculate the similarity matrix using dot-product. This results in the noise of a modality influencing the multimodal representation. On the other hand, the proposed CMRL uses other modalities to build the edges between nodes. These edges, while representing the similarity between two modalities, do not directly interact with other modality's features. This

way, the impact of noise is removed during the multimodal feature enhancement. The other simpler approach to reduce the impact of one modality's noise could be the application of self-attention on individual modality features before concatenating them. However, in that way, the features of different modalities become independent of each other. We have shown in ablation analysis that the proposed approach outperforms both methods.

For Algorithm 6.2, graph construction (lines 2-6) has the most significant impact on the overall time complexity. In line 4, computing the similarity matrix S^m involves calculating the cosine similarity between pairs of D -dimensional embedding vectors for a total of N^2 pairs. The remaining steps can be disregarded as they do not significantly affect the overall time complexity. Therefore, the time complexity of Algorithm 6.2 is $O(N^2 \times D)$.

6.3.6 Multifeature Learning and Classification

This section describes a multifeature learner network that derives its inputs from the application of HIMA and CMRL, concatenated with task-specific features discussed in Section 6.3.3.3. In the fusion process, we first concatenate the emotion feature vector, sentiment feature vector, and toxicity feature vector for the k -th sample to form a unified representation of task-specific features denoted by T_k . Then, we concatenate the embeddings obtained from the application of HIMA (Z_k), the embeddings derived from the application of CMRL (H_k), and T to form a multifeature vector M given by:

$$M = \text{concat}(Z_k, H_k, T_k) \quad (6.27)$$

Subsequently, M is passed through a series of fully connected layers to obtain a hidden representation H , which serves as the input to multiple output layers, each corresponding to one of the tasks. In our case, since we are addressing five distinct tasks, we have five separate output layers. Each output layer is equipped with a softmax activation function, which normalizes the output scores across the different classes for its respective task. The output $\hat{y}_i$ of a task-specific output layer for task i is computed as follows:

$$\hat{y}_i = \text{softmax}(H\mathbf{W}_i + \mathbf{b}_i) \quad (6.28)$$

where $\mathbf{W}_i$ and $\mathbf{b}_i$ denote the trainable parameters corresponding to task i .

The loss function is defined using categorical cross-entropy for each task and summing it up. Let T represent the number of tasks, C_i denote the number of classes for task i , and y_{ij} be the true label for class j in task i . The loss function $\mathcal{L}$ is computed as:

$$\mathcal{L} = - \sum_{i=1}^T \sum_{j=1}^{C_i} y_{ij} \log(\hat{y}_{ij}) \quad (6.29)$$

where $\hat{y}_{ij}$ represents the predicted probability of class j in task i . Minimizing this total loss during training ensures that the model learns to make accurate predictions across all tasks simultaneously.

MM-ORIENT is trained for 200 epochs with a batch size of 128 and a learning rate of 5e-6. We conducted all experiments on a computing system equipped with an Intel(R) Xeon(R) Silver 4316 CPU @ 2.30GHz, 512 GB RAM, and four NVIDIA A16 GPUs. The training time of MM-ORIENT is 1.3 seconds per sample, whereas the testing time is 0.2 seconds per sample.

6.4 Experimental Evaluations

6.4.1 Datasets

6.4.1.1 Memotion [199]

The Memotion dataset, introduced as part of the SemEval-2020 Memotion Analysis task, is a comprehensive collection of 9,871 publicly available memes, each encompassing two key modalities: text and images. The dataset includes labels for sentiment, humor, sarcasm, offense, and motivation, making it particularly well-suited for tasks related to sentiment analysis, humor detection, sarcasm identification, offense detection, and motivational content

6.4 Experimental Evaluations

recognition. Table 6.1 represents the dataset statistics, including the number of original and augmented training samples, as well as the number of test samples for each category.

Table 6.1: Memotion Dataset Statistics

Task	Label	#Train_orig	#Train + Aug	#Test
Sentiment	Negative	629	4,948	173
	Neutral	2,193	3,332	594
	Positive	4,138	6,923	1,111
Humor	Not Funny	1,646	4,297	445
	Funny	2,442	3,782	654
	Very Funny	2,224	3,757	605
	Hilarious	648	3,367	174
Sarcasm	Not Sarcastic	1,540	3,057	421
	General	3,490	5,688	937
	Twisted Meaning	1,540	3,410	424
	Very Twisted	390	3,048	96
Offensiveness	Not Offensive	2,699	4,337	707
	Slight	2,583	4,662	709
	Very Offensive	1,459	3,770	387
	Hateful Offensive	219	2,434	75
Motivational	Not Motivational	4,505	9,877	1,188
	Motivation	2,455	5,326	690

6.4.1.2 MMHS150K [200]

The MMHS150K dataset comprises 1,50,000 tweets with text and images collected from Twitter. It focuses on tweets containing 51 Hatebase terms associated with hate speech. Notably, the dataset includes both the raw, unfiltered content and associated images, providing a snapshot before content filters are applied. We use hate and not-hate classification of tweets. A total of 1,34,823 samples are utilized for training, 5,000 samples for validation, and 10,000 samples for testing.

6.4.1.3 HarMeme Dataset [201]

The HarMeme dataset contains a total of 3544 memes. The dataset is labeled into three categories: very harmful, partially harmful, and harmless. For binary classification, we combine very harmful and partially harmful classes. In the final dataset, we get 1249 harmful and

2295 harmless memes. The dataset is split into training, validation, and testing with 3013, 177, and 354 samples, respectively.

6.4.2 Baselines for Comparison

In this section, we provide an overview of the baseline methods utilized in our experiments, categorized into two groups: monomodal and multimodal techniques. Monomodal methods focus solely on either text or image information, whereas multimodal methods integrate both text and image data for analysis.

BERT [100] and Mask RCNN-based detectron2 [161] monomodal methods are taken text and images, respectively. We utilize the ResNeXt-152-based (X152) Mask RCNN [176] model for image region features.

In multimodal methods we take Decision-level Fusion [177], Feature-level Fusion [177], MMCA [202], MMSA [170], CLIP [160], VisualBERT [178], ViLT [179], FCM [200], Dis-MultiHate [203], MSKAV+KDAC [13], Momenta [204], MR.HA- RM [205], MIMN [206], MIIM [207], MIMN-MTL [206], MIIM-MTL [207], HIMM [14], Ensemble [208], Aye-tiran [209], MMFHS [210], ALFRED [211], and GPT-4o [186]. GPT-4o, the latest multimodal model from OpenAI, is employed in the prompt-based zero-shot classification approach.

6.4.3 Results

In this section, we compare the performance of our proposed method with the baseline methods mentioned in the previous section across three different datasets: Memotion, MMHS150K, and HarMeme. The comparison results are shown in Table 6.2, Table 6.3, and Table 6.4, respectively. It should be noted that all the scores are presented in percentages(%).

6.4.3.1 Performance Comparison on Memotion Dataset

Table 6.2 presents a comprehensive performance comparison of various methods on the Memotion dataset. To facilitate a meaningful comparison, we utilize the same number of samples as adopted by HIMM [14] for generating the results. Instances containing fewer

6.4 Experimental Evaluations

Table 6.2: Performance Comparison on Memotion Dataset

	Method	Sentiment	Humor	Sarcasm	Offensive	Motivation
Monomodal	BERT [100]	39.24	32.15	45.42	39.51	60.15
	Mask RCNN X152 [176]	45.96	34.65	41.05	34.00	62.23
Multimodal	Decision-level Fusion [177]	46.19	36.81	45.65	40.77	62.66
	Feature-level Fusion [177]	45.86	36.43	46.06	40.82	62.51
	MMCA [202]	44.76	35.62	45.18	39.52	61.33
	MMSA [170]	46.30	36.97	46.11	41.34	62.89
	CLIP [160]	47.28	37.52	46.96	41.24	64.58
	VisualBert [178]	44.26	34.47	43.74	39.85	62.44
	ViLT [179]	45.36	34.18	46.27	37.04	63.28
	MIMN [206]	35.34	29.90	47.18	35.34	59.61
	MIIM [207]	36.12	31.26	45.05	34.37	50.29
	MIMN-MTL [206]	42.91	34.56	48.54	35.15	63.11
	MIIM-MTL [207]	40.97	35.15	48.35	36.31	63.30
	HIMM [14]	46.80	36.70	48.74	41.55	65.05
	Ensemble [208]	45.42	37.25	45.89	41.33	62.87
	GPT-4o [186]	29.25	34.12	29.49	35.42	63.88
	MM-ORIENT	49.94	38.09	49.38	43.61	66.41

than 10 words in the text are omitted. Following this step, the dataset is further partitioned into training, validation, and testing sets using an 8:1:1 ratio. However, for the ablation analysis, we utilize samples in the ratio depicted in Table 6.1. To ensure consistency among the results, we use micro-F1 as adopted in MIMN, MIIM, MIMN-MTL, MIIM-MTL, and HIMM.

Notably, our proposed method stands out across all tasks. It achieves the highest micro-F1 scores in all the tasks. Monomodal methods, BERT (text) and Mask RCNN X152 (image) do not consider both modalities; hence, these models understandably provide lower performance than the proposed method. MM-ORIENT outperforms BERT by the highest margin of 10.70% in the sentiment task and Mask RCNN X152 by 9.61% in the offensive task. Feature-level fusion integrates both modalities before further processing, whereas decision-level fusion joins them after processing through the network. These methods do not apply any attention mechanism. MM-ORIENT outperforms these two methods in all the tasks, but the highest gain is achieved in the sentiment task, where the proposed method achieves 3.75% and 4.08% higher micro-F1 than decision-level fusion and feature-level fusion, respectively. The MMSA method, based on self-attention, outperforms the aforementioned fusion techniques, demonstrating the ability of monomodal attention to concentrate on discriminative information. Similarly, our proposed method employs monomodal attention, specifically HIMA, leading to enhancements in multitasking performance. The lower performance of MMCA denotes that cross-attention is influencing multimodal representation

because of noise. Hence, the proposed approach learns multimodal representations without direct interaction between both modality features that reduces the influence of noise. The highest gain against MMCA and MMSA is achieved in the sentiment task, where the proposed method achieves 5.18% and 3.64% higher micro-F1, respectively. Among the VisualBERT, ViLT, and CLIP methods, CLIP achieves competitive scores, indicating the contrastive learning-based approach can better understand and interpret multimodal content than the counterpart transformer-based models. Against CLIP, MM-ORIENT achieves the highest gain of 2.66% in the sentiment task. Nevertheless, the intricate high-order intramodality and intertask relationships have not been thoroughly addressed in all these methods. GPT-4o, employed in zero-shot settings using the prompt-based approach, performs relatively less effectively than all the other methods. In the sentiment task, MM-ORIENT outperforms it by 20.69% in terms of micro-F1. This indicates that large language models (LLMs) must be finetuned, as they may not perform well specific downstream tasks in zero-shot settings.

MIMN and MIIM methods, which have already been tested on the Memotion dataset, are analyzed. MIMN employs two memory networks for interaction between text and image representations, whereas MIIM employs a BERT-based approach and a co-attention module. Results indicate that these two methods could not comprehend the content effectively for multitasking. Multimodal-multitask methods, MIMN-MTL, MIIM-MTL, and HIMM, show significant performance improvement and generalization ability, with HIMM performing relatively better. Against HIMM, MM-ORIENT achieves a performance gain of 3.14%, 1.61%, 2.64%, 1.06%, and 1.36% in sentiment, humor, sarcasm, offensive, and motivation tasks, respectively. These methods fall short of accurately capturing potential high-order relationships within and between modalities and overlook the facilitation among multiple tasks. In comparison, the proposed method uses cross-modal relation learning using graphs to comprehend multimodal content and hierarchical interactive attention to facilitate capturing discriminative information to help in multitasking. The quantitative results demonstrate the efficacy of the proposed method in capturing high-order multimodal relationships and the generalizing ability for multitasking.

6.4 Experimental Evaluations

Table 6.3: Performance Comparison on MMHS150K Dataset

	Method	Acc	P	R	F1
Monomodal	BERT [100]	77.19	68.29	56.78	62.01
	Mask RCNN X152 [176]	70.16	56.35	55.30	55.82
Multimodal	Decision-level Fusion [177]	78.94	67.22	62.26	64.64
	Feature-level Fusion [177]	78.73	66.74	62.45	64.52
	MMCA [202]	77.61	65.24	60.85	62.97
	MMSA [170]	79.17	67.58	63.05	65.24
	CLIP [160]	79.26	68.83	63.57	66.09
	VisualBert [178]	76.51	63.53	61.74	62.62
	ViLT [179]	75.40	62.65	59.75	61.17
	FCM [200]	68.40	-	-	70.40
	DisMultiHate [203]	76.54	67.51	71.42	69.40
	MSKAV+KDAC [13]	80.78	68.98	72.04	70.49
	Ayetiran [209]	68.70	65.90	76.10	70.60
	MMFHS [210]	70.26	69.00	70.00	70.00
	GPT-4o [186]	53.02	51.68	93.27	66.51
	MM-ORIENT	83.45	76.45	72.60	74.47

6.4.3.2 Performance Comparison on MMHS150K Dataset

Table 6.3 provides a comprehensive performance comparison of different methods on the MMHS150K dataset across different metrics, including accuracy (Acc), precision (P), recall (R), and F1-score (F1). This dataset focuses on hate speech detection, a critical task in online content moderation.

Among the monomodal approaches, both BERT and Mask RCNN X152 are evaluated. BERT achieves an F1-score of 62.01% and an accuracy of 77.19%, demonstrating a solid understanding of textual content in identifying hate speech. However, Mask RCNN X152 lags slightly behind in accuracy at 70.16%. This suggests that while image-based methods can contribute to hate speech detection, they may not be as effective as their text-based counterparts in isolation. MM-ORIENT outperforms BERT by a margin of 11.46% and Mask RCNN X152 by a margin of 18.65% in terms of F1-score.

Results on the MMHS150K dataset denote that focusing on pertinent information before modality fusion helps in different tasks. This is affirmed by examining various fusion techniques, revealing that decision-level and MMSA fusion exhibit comparatively superior performance when compared to feature-level fusion and MMCA methods. This is attributed to the fact that both decision-level and MMSA methods prioritize individual modalities before combining them. The proposed MM-ORIENT achieves a performance gain of 11.5%

and 9.23% against MMCA and MMSA, respectively, in terms of F1-score. CLIP outperforms transformer-based multimodal methods such as VisualBERT and ViLT, suggesting that the multimodal contrastive learning approach has a superior ability to interpret multimodal content. These aforementioned techniques achieve satisfactory results but fall short in modeling complex intermodal and intramodal relationships. GPT-4o again achieves relatively lower scores than the other methods due to zero-shot settings. Next, we analyze methods that have been proposed for the tasks specific to the MMHS150K dataset. FCM, Dis-MultiHate, MSKAV, Ayetiran and MMFHS methods also achieve competitive accuracy and F1-scores, with MSKAV performing relatively better than the others. MM-ORIENT outperforms MSKAV by a margin of 2.67% and 3.98% in terms of accuracy and F1-score, respectively. The aforementioned three techniques generate high-order multimodal relationships via direct feature interaction at the latent stage, which influences pertinent monomodal information due to inherent noise. In contrast, our proposed approach generates cross-modally trained multimodal representation and avoids a direct interaction between monomodal features at the early stage. This is achieved by the cross-modal relation graphs. Additionally, proposed HIMA attention is effective in bringing out discriminative information from text and image features.

6.4.3.3 Performance Comparison on HarMeme Dataset

Table 6.4: Performance Comparison on HarMeme Dataset

	Method	Acc	P	R	Macro-F1
Monomodal	BERT [100]	72.41	70.18	66.87	67.61
	Mask RCNN X152 [176]	66.95	64.01	64.06	64.04
Multimodal	Decision-level Fusion [177]	73.85	71.51	71.58	71.54
	Feature-level Fusion [177]	72.41	70.31	71.19	70.61
	MMCA [202]	71.27	70.98	68.62	69.31
	MMSA [170]	75.00	72.73	72.11	72.38
	CLIP [160]	83.07	81.63	81.31	81.46
	VisualBert [178]	79.27	78.93	77.80	78.46
	ViLT [179]	74.85	72.90	74.33	73.32
	V-BERT COCO [178]	81.36	79.55	81.19	80.13
	Momenta [204]	83.82	-	-	82.80
	MR.HARM [205]	86.16	-	-	85.43
	ALFRED [211]	-	-	-	85.88
	GPT-4o [186]	67.51	63.55	58.27	57.62
	MM-ORIENT	88.32	87.74	86.37	86.84

6.4 Experimental Evaluations

Table 6.4 illustrates the performance of various methods on the Harmful Meme dataset, evaluated using metrics such as accuracy (Acc), precision (P), recall (R), and macro-F1 (F1). This dataset focuses on detecting harmful content in memes, an important aspect of content moderation.

For monomodal approaches, MM-ORIENT achieves a 19.23% gain in macro-F1 compared to BERT, and a 22.80% improvement over Mask RCNN X152. These results indicate that BERT performs slightly better with text-based content analysis compared to Mask RCNN X152 with image content. This indicates the challenges of using visual data alone for this task. Among the multimodal approaches, Decision-level Fusion and MMSA methods show competitive performance. CLIP, which employs a multimodal contrastive learning approach, significantly outperforms other methods but lags behind MM-ORIENT by a margin of 5.38% in terms of macro-F1. VisualBert and ViLT also perform well, but not as effectively as CLIP.

Specialized methods for this dataset, such as Momenta, MR.HARM, and ALFRED achieve even higher performance, with MR.HARM reaching 86.16% accuracy and 85.43% F1-score. However, MM-ORIENT outperforms MR. HARM by 2.16% and 1.41% in terms of accuracy and macro-F1. This demonstrates the effectiveness of MM-ORIENT in leveraging both textual and visual data to accurately detect harmful memes.

6.4.4 Ablation Analysis

In this section, we present a detailed ablation analysis to gain a deeper understanding of the factors contributing to the performance of our proposed method. The analysis in this section focuses exclusively on the Memotion dataset to ensure a consistent and comprehensive examination of the model’s performance. By isolating and modifying specific aspects of the model, we aim to shed light on the critical elements that enable the models to excel in multiple tasks.

Through a series of experiments, we systematically investigate the impact of different components and architectural choices on the model’s performance on multiclass classification tasks (shown in Tables 6.5-6.9) as well as binary classification tasks (shown in Figures 6.5-6.9). Four tasks, humor recognition, sarcasm detection, offensive content identification, and motivation detection, are analyzed for binary classification, where the presence of that

emotion is classified as one class, and the absence of that emotion is classified as another class. For instance, different humor levels, such as funny, very funny, and hilarious, are classified using a single label as humorous as they all depict the presence of humor, whereas not funny is classified with a different label as not humorous. The sentiment classification task is performed only in a multiclass setting. Micro-F1 is used as an evaluation metric for ablation analysis.

6.4.4.1 Effect of Modalities

In order to assess the individual impact of each modality, we conduct experiments utilizing features exclusively from each modality. The results are shown in Table 6.5 and Figure 6.5 for multiclass and binary classification, respectively.

Table 6.5: Ablation Analysis on Different Modalities in MM-ORIENT

	Sentiment	Humour	Sarcasm	Offensive	Motivation
Image Only	54.52	31.81	36.59	35.16	61.34
Text Only	53.01	29.13	45.47	37.21	60.79
MM-ORIENT	58.20	35.49	49.50	39.45	63.78

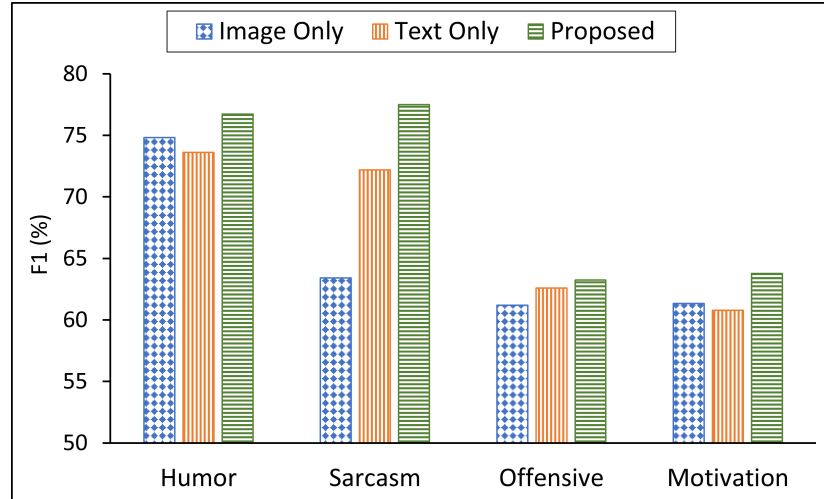


Figure 6.5: Influence of Different Modalities on Binary classification in MM-ORIENT

For the text modality, we pass the CLIP text features to the graph component, and BERT features to the attention component. The task-specific features are concatenated. Subsequently, the features from all these three components are concatenated and passed to fully

6.4 Experimental Evaluations

connected dense layers. Similarly, for the image modality, we follow identical steps, using CLIP image features and Mask RCNN X152 features instead of the CLIP text features and BERT features. These are then directed to the image graph and region-based attention component, respectively. Given the absence of text in this analysis, toxicity BERT features are omitted from the additional features. The subsequent steps remain the same. When excluding the text modality and exclusively utilizing the image modality, we observe a reduction in the micro-F1 score by 3.68% for sentiment classification and by an average of 5.83% and 5.12% for multiclass and binary classification, respectively. It should be noted that for both multiclass and binary classification, the average is performed across all tasks mentioned in Table 6.1 except sentiment. Similarly, by removing the image modality and solely relying on the text modality, we see a decrease in the micro-F1 score by 5.19% for sentiment classification and by an average of 3.91% and 3% for multiclass and binary classification, respectively. This notable drop in performance across all three tasks when the image modality is removed suggests the substantial significance of image modality compared to the text modality.

6.4.4.2 Effect of Attention Components

Investigating the attention components, we systematically examine the effects of both word-level attention and region-based attention. To do this, we conduct experiments where each attention component is removed while keeping the remainder of the architecture consistent. The results are summarized in Table 6.6 and Figure 6.6 for multiclass and binary classification, respectively.

Table 6.6: Ablation Analysis on Attention Components in MM-ORIENT

	Sentiment	Humour	Sarcasm	Offensive	Motivation
w/o RBA	56.76	31.58	48.27	37.21	62.23
w/o WLA	55.88	33.15	48.02	36.46	62.34
w/o RBA & WLA	54.72	30.13	47.30	36.04	61.28
MM-ORIENT	58.20	35.49	49.50	39.45	63.78

Firstly, when excluding region-based attention, we observe a notable reduction in the micro-F1 score across all tasks. Specifically, there is a decrease of 1.44% for sentiment classification, 2.23% for multiclass classification, and 1.34% for binary classification. Similarly, omitting word-level attention leads to a decrease in performance, with a reduction of 2.32%

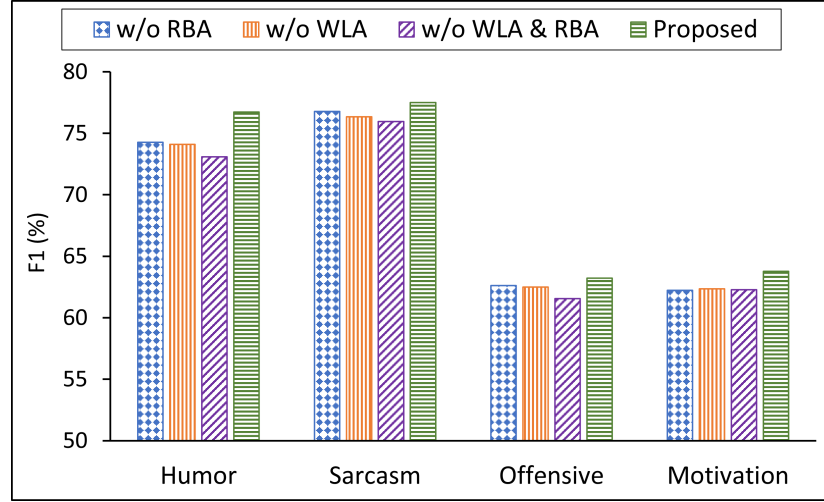


Figure 6.6: Influence of Different Attention Mechanisms on Binary Classification in MM-ORIENT

for sentiment classification, 2.06% for multiclass classification, and 1.5% for binary classification. Additionally, when both region-based and word-level attention are removed, we observe a substantial drop in performance, with a decrease of 3.48% for sentiment classification, 2.82% for multiclass classification, and 2.1% for binary classification. These results underscore the critical role that both word-level and region-based attention play in enhancing the model’s performance across a range of tasks.

6.4.4.3 Effect of Different Attention-based Fusion Mechanisms

To evaluate the impact of the proposed attention mechanism HIMA as compared to the different attention-based fusion mechanisms, we perform experiments using MMCA [202] and MMSA [170] while maintaining a consistent architecture. The outcomes are detailed in Table 6.7 and Figure 6.7 for multiclass and binary classification tasks, respectively. Using MMSA, we observe reasonable performance across all tasks in multiclass classification, with micro-F1 scores of 56.96% for sentiment classification, 33.61% for humor detection, 47.84% for sarcasm detection, 37.43% for offensive content detection, and 62.55% for motivation classification.

MMCA mechanism results in slightly lower performance, particularly for sarcasm and motivation detection. In contrast, the MM-ORIENT mechanism consistently outperforms both MMSA and MMCA, achieving the highest scores across all tasks: 58.20% for senti-

6.4 Experimental Evaluations

Table 6.7: Effect of Different Attention-based Fusion Mechanisms in MM-ORIENT

	Sentiment	Humour	Sarcasm	Offensive	Motivation
with MMSA	56.96	33.61	47.84	37.43	62.55
with MMCA	56.17	32.71	46.68	37.26	61.35
MM-ORIENT	58.20	35.49	49.50	39.45	63.78

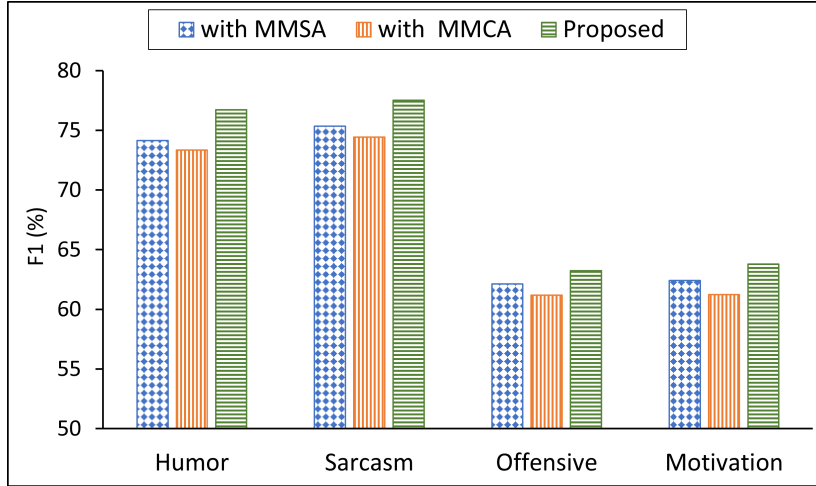


Figure 6.7: Influence of Different Attention-based Fusions on Binary Classification in MM-ORIENT

ment, 35.49% for humor, 49.50% for sarcasm, 39.45% for offensive content, and 63.78% for motivation. As shown in Figure 6.7, in binary classification too, HIMA-based MM-ORIENT achieves higher scores across all tasks as compared to MMSA and MMCA. These findings underscore the efficacy of the HIMA method in improving MM-ORIENT’s performance.

6.4.4.4 Effect of Graph Components

Examining the graph components, we systematically investigate the effects of both cross-modal and in-modal graphs. In order to do this, we conduct experiments where each graph component is removed while keeping the rest of the architecture unchanged. The results are summarized in Table 6.8. When excluding in-modal graphs, we observe a decrease of 1.93% in micro-F1 for sentiment and an average of 1.62% decrease for other tasks in a multiclass setting. Moreover, as shown in Figure 6.8, we observe a decrease of 1% in the average micro-F1 score over four tasks in a binary classification setting.

Results suggest that in-modal graphs help focus on intramodal feature refinement. However, without cross-modal relation graphs, there is a more pronounced impact on perfor-

Table 6.8: Ablation Analysis on Graph Components in MM-ORIENT

	Sentiment	Humour	Sarcasm	Offensive	Motivational
w/o In-modal Graphs	56.27	34.37	47.39	36.72	63.25
w/o Cross-modal Graphs	55.72	32.15	46.30	35.04	62.28
MM-ORIENT	58.20	35.49	49.50	39.45	63.78

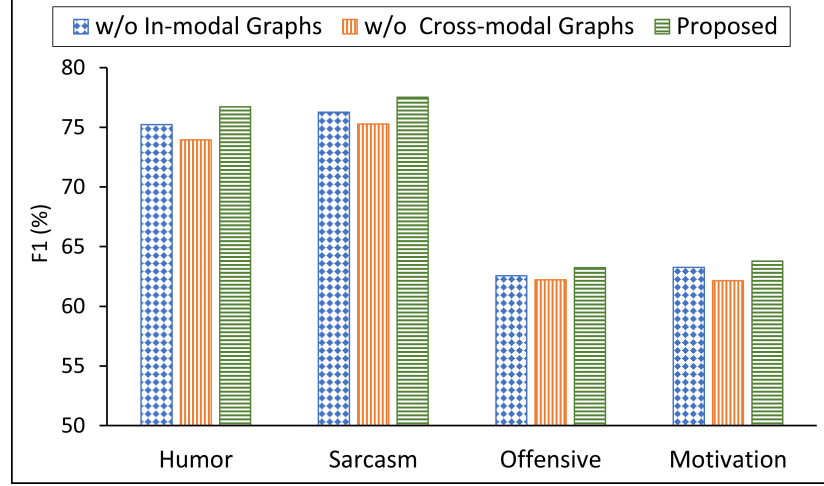


Figure 6.8: Influence of Different Graph Components on Binary Classification in MM-ORIENT

mance, suggesting that cross-modal graphs generate rich multimodal features. The micro-F1 score sees a substantial decrease of 2.48% for sentiment and an average of 2.84% decrease for other tasks in a multiclass setting. For binary classification, a decrease of 1.92% is observed in the average micro-F1 score over four tasks as shown in Figure 6.8. These findings emphasize that cross-modal relation graphs provide efficient multimodal representations which in turn improve the model’s performance. This confirms the ability of cross-modal relation graphs for effective semantic comprehension of image and text-based multimodal content.

6.4.4.5 Effect of Additional features

We systematically remove each of the features individually while keeping the rest of the architecture unchanged in order to evaluate the impact of each of these features as shown in Table 6.9 and Figure 6.9 for multiclass and binary classification respectively.

Firstly, without conducting image cleaning, we observe a decrease of 1.05% for sentiment, an average of 0.86% for multiclass classification, and an average of 0.75% for binary classification in the micro-F1. Without the augmentation, we witness a substantial decrease

Table 6.9: Ablation Analysis on Additional Features in MM-ORIENT

	Sentiment	Humour	Sarcasm	Offensive	Motivation
w/o Image Cleaning	57.15	34.75	48.48	38.26	63.28
w/o Augmentation	54.39	33.64	47.34	36.60	61.06
w/o Emotion	56.27	34.37	47.75	37.72	63.07
w/o Sentiment	57.89	35.12	48.88	39.26	63.34
w/o Toxicity	57.48	34.24	48.24	38.33	62.79
MM-ORIENT	58.20	35.49	49.50	39.45	63.78

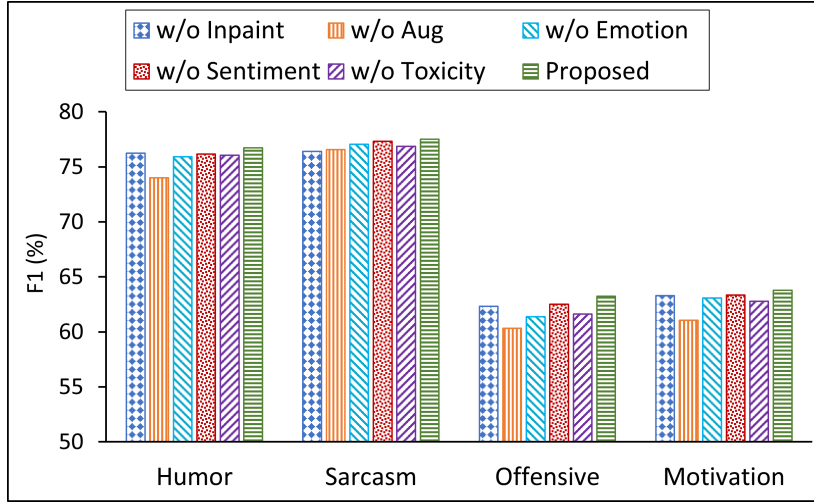


Figure 6.9: Influence of Features on Binary Classification in MM-ORIENT

in performance. The micro-F1 score drops by 3.81% for sentiment, an average of 2.4% for multiclass classification, and an average of 2.32% for binary classification. Next, by excluding emotion features, observe a notable decrease in performance. The micro-F1 score diminishes by 1.93% for sentiment, an average of 1.32% for multiclass classification, and an average of 0.95% for binary classification. Subsequently, removing the sentiment features, results in a marginal reduction of 0.31% for sentiment, an average of 0.4% for multiclass, and an average of 0.47% for binary classification. Similarly, the removal of toxicity-related features results in a decrement of 0.72% for sentiment, an average of 1.15% for multiclass, and an average of 0.98% for binary classification. These findings highlight the importance of each component in enhancing the model’s performance across various tasks. The results denote that augmentation contributes the most to improve the model’s performance as compared to other features as shown in Table 6.9 and Figure 6.9.

6.4.5 Qualitative Analysis

Visual Modality	(a) 	(b) 	(c) 	(d) 
Textual Modality	Gave \$100 to beggar dies of cocaine overdose	Science improves your neural network	Kill Count Hitler: 11 million Mao: 20 million +	On facebook you can say anything I Like
Ground Truth	Neutral, Not Funny Twisted Meaning, Very Offensive, Not Motivational	Positive, Funny General, Not Offensive, Motivational	Neutral, Hilarious Very Twisted, Hateful, Not Motivational	Neutral, Funny General, Slight Offensive, Motivational
Predictions	Neutral, Not Funny Not Sarcastic, Very Offensive, Not Motivational	Positive, Funny General, Not Offensive, Motivational	Negative, Not Funny Very Twisted, Hateful, Not Motivational	Neutral, Funny General, Not Offensive, Motivational
Visual Modality	(e) 	(f) 	(g) 	(h) 
Textual Modality	On facetime with your best friend be like	So SNL, your unfunny weekend update sketch about misogyny had no impact on the industry may be it wasn't long enough	When my friends ask if I want to get drunk	Why socialize when there's perfectly good wifi and food at home
Ground Truth	Positive, Very Funny General, Slight Offensive, Not Motivational	Neutral, Funny Twisted Meaning, Very Offensive, Not Motivational	Neutral, Funny Very Twisted, Slight Offensive, Not Motivational	Positive, Very Funny Very Twisted, Very Offensive, Motivational
Predictions	Positive, Very Funny General, Slight Offensive, Not Motivational	Negative, Funny Twisted Meaning, Slight Offensive, Not Motivational	Neutral, Funny Very Twisted, Slight Offensive, Not Motivational	Positive, Very Funny Very Twisted, Very Offensive, Not Motivational

Figure 6.10: Qualitative Analysis of MM-ORIENT

The qualitative analysis of our meme comprehension framework involves a detailed examination of how the model interprets and categorizes different memes based on sentiment, humor, sarcasm, offensiveness, and motivation.

In Figure 6.10(a), the visual component portrays a child with a determined facial expression, which, on its own, might convey a positive sentiment. However, a closer examination of the accompanying text, “Gave \$100 to beggar dies of cocaine overdose,” reveals a stark contrast, describing a negative outcome following what initially seems like a charitable act. This contradiction results in an overall neutral sentiment for the meme. Despite the ostensibly positive visual element, the meme, with its morbid twist, lacks humor and motivation. The model successfully predicts a neutral sentiment, appropriately recognizing the offen-

siveness of the content. However, despite the ironic interplay between the visual and textual components, the model falls short of capturing the sarcasm present in this particular instance.

Moving to Figure 6.10(b), we encounter an image featuring a smiling woman wearing spectacles with the accompanying text, "science improves your neural network." This combination delivers an overall positive and motivational message. There is an absence of any offensive content in both the visual and textual components, rendering the meme categorically "Not Offensive." MM-ORIENT correctly predicts all the tasks for this meme.

In Figure 6.10(c) features an image of Mao Zedong. The accompanying text provides a "Kill Count" comparison, stating that Hitler is attributed to 11 million deaths, while Mao's count is indicated to be 20 million or more. Although the given label for sentiment is neutral, the correct sentiment of this meme is likely negative, as predicted by our model, due to the serious and grim nature of the content. Additionally, despite the actual label for humor being "Hilarious," the meme doesn't appear to be designed for humor. The subject matter is sensitive, which is accurately captured by our model by classifying the meme as "Not Funny." The seemingly straightforward presentation of death tolls creates a sarcastic contrast with the gravity of the subject matter. Consequently, the meme is appropriately categorized as "Very Twisted" for sarcasm. The meme could be considered offensive, particularly to those with strong opinions about historical events and figures, warranting the highest degree of offensiveness classification - "Hateful."

The meme in Figure 6.10(d) is interpreted as negative due to its implication of restricted freedom of expression, despite being labeled as "funny" for its ironic humor. The meme sarcastically suggests that Facebook users can only express opinions that align with Mark Zuckerberg's preferences, creating a satirical tone. While it addresses content moderation on social media, the model incorrectly predicts it as "Not Offensive." Both our model and the actual label classify the meme as "Motivational," but a more accurate label would be "Not Motivational" due to its emphasis on restriction rather than inspiration. Mislabeling like this, though rare, can still occur in the dataset.

Similarly, the memes depicted in Figures 6.10(e) through 6.10(h) can be analyzed. In Figures 6.10(e) and 6.10(g), if we look at the text modality alone, it seems that the label for offensiveness should be "Not Offensive," but then if we combine it with the visual modality, we can say that the label should be "Slight Offensive" as given in the labels and our model's

predictions. For each task, our model predicted the labels correctly in both of these figures. On the contrary, in Figure 6.10(f), the text modality alone is sufficient to determine that the meme should be classified in the “Very Offensive” category on the basis of offensiveness. It can be observed that our model falls slightly behind in this task and classifies it as “Slight Offensive” for this sample. Interestingly, for the sentiment classification, despite the actual label being “Neutral,” our model predicts the label to be “Negative,” which seems more appropriate for this sample. Similarly, for Figure 6.10(h), the model predicts the class as “Not Motivational” for motivation detection, which seems to be the case here despite the actual label being “Motivational.” Hence, for figures from 6.10(e) to 6.10(h), our model predicts the relevant and appropriate labels across all tasks.

6.5 Limitations

While MM-ORIENT is primarily designed for multimodal inputs comprising image and text, the proposed architecture is inherently modular. Each component, including the Hierarchical Interactive Monomodal Attention (HIMA), Cross-modal Relation Learning (CMRL), and multifeature learner, is built to operate on high-level feature representations. As such, the framework can be extended to other modality pairs such as video–text, audio–text, or even tri-modal scenarios, video, audio, and text, by substituting the current feature extractors with suitable alternatives such as TimeSformer for video, VGGish or Wav2Vec for audio. This opens avenues for applying MM-ORIENT to a broader class of semantic comprehension tasks beyond image–text settings. However, this work focuses on image–text pairs due to their prevalence in web media and leaves exploration of additional modalities as future work. Additionally, the overall performance of MM-ORIENT is heavily reliant on the quality and domain alignment of the pre-trained feature extractors. Suboptimal or mismatched encoders may impact its performance, which is also true for other methods. Furthermore, while the architecture generalizes well across typical web-based data, its performance on domain-specific content such as medical or legal images and texts remains untested and may require significant adaptation or fine-tuning.

6.6 Conclusion

In this chapter, we propose a novel multimodal-multitask framework, MM-ORIENT, to comprehend visual and textual element-based multimodal content for multiple tasks such as sentiment, humor, offensiveness, and sarcasm detection. Our findings suggest that the inherent noise in monomodal features hampers the effectiveness of fusion techniques in creating efficient multimodal representations due to the direct interaction of features at the latent stage. To address this, we propose a cross-modal relation graph-based approach for learning multimodal representations. This approach provides multimodal features without requiring direct interaction between features of different modalities in the initial stages. Our findings show that MM-ORIENT effectively reduces noise and generates high-level multimodal features in a cross-modal manner. The proposed attention mechanism, HIMA, employs a two-stage learning process to extract discriminative information from individual modalities. Additionally, our results indicate that generative AI-based augmentation enhances MM-ORIENT's performance as augmented samples help generalize the results. We have performed extensive experiments on three benchmark multimodal datasets to analyze the performance of the proposed method. In conclusion, the findings indicate that the proposed framework effectively cognizes multiple tasks in image and text-based multimodal content. By employing both quantitative and qualitative analyses on the Memotion dataset, we have shown that our method consistently surpasses the performance of existing advanced approaches.

Chapter 7

Multimodal Hostile Content Detection via LLM Integration

7.1 Introduction

Multimodal hostile content detection has gained increasing attention with the emergence of LLMs. Integrating LLMs into multimodal frameworks is particularly crucial for detecting the forms of hostility that often evade traditional classifiers. One such challenging manifestation is dark humor, which frequently embeds offensive intent beneath layers of subtlety and ambiguity. Dark humor represents a challenging domain for data mining, as it uses subtle elements to convey complex and often controversial forms of humor. The difference between conventional “clean” humor and dark humor can be quite subtle. While there can never be an objective way to distinguish them, there are some key identifiers for dark humor. Clean humor generally uses topics that everyone can relate to and collectively laugh at. On the other hand, dark humor exploits topics that are typically considered insensitive, as shown in Figure 7.1. According to Ruch et al. [74], dark humor typically incorporates one of four distinct elements, sarcasm, cynicism, satire, or irony, each characterized by unique linguistic and cognitive features that shape its expression.

However, in today’s digital landscape, social media platforms frequently convey dark humor not just through linguistic expressions but also by image-text combinations [212, 213] such as memes, which add a multimodal layer to their interpretation. While memes have



Figure 7.1: Comparison of Clean and Dark Humor

become a dominant form of online communication, detecting dark humor within them remains an underexplored problem due to the interplay between imagery, text, and cultural context. Effectively mining and detecting dark humor memes requires multimodal approaches capable of integrating textual and visual cues to accurately capture these subtle humor styles [214,215]. Advancing automated detection in this space can significantly enhance content moderation, sentiment analysis, and the broader understanding of how humor operates in complex, multimodal social media data [16]. However, to the best of our knowledge, there exists no multimodal dark humor dataset, which limits progress in this underexplored yet critical area.

To address the scarcity of resources for studying dark humor in memes, we introduce a novel dataset comprising a large collection of memes sourced from popular social media platforms, each containing both visual and textual components for comprehensive multimodal analysis. Annotators annotated the dataset not only to distinguish between dark humor and non-dark humor memes in a binary classification framework but also to identify target groups commonly referenced or affected by dark humor, including gender, mental health, violence, race, disability, and other sensitive categories. This granular labeling supports a deeper exploration of the social and ethical dimensions embedded within dark humor, enabling both the technical challenge of humor detection and the analysis of its underlying themes and

potential impacts in online meme culture. Since humor plays an important role in mental health [216,217], our dataset could be an important resource.

In addition to the dataset, we propose a novel multimodal approach for dark humor detection in memes that harnesses the complementary strengths of large language models (LLMs) [218] and visual-textual feature extraction. Specifically, we first employ the Qwen-2.5-7B Vision–Language Model to generate an initial structured reasoning for each meme, capturing high-level contextual and inferential cues. Next, we apply an Iterative Reasoning Refinement via Role-Reversal Self-Loop. Qwen is prompted to “become the original author” of the meme and critically review its own explanation, verifying that each field (summary, implied joke, narrative structure, emotional effect, dark attributes, and target) is both faithful and complete. This role-reversal step mimics reflective human reasoning, encouraging the model to correct omissions or misalignments before yielding a final, refined explanation. Textual features are then extracted from both the meme’s transcript using BERT and the self-refined reasoning using S-BERT, while visual features are obtained via ViT for multimodal representation learning. To effectively model the interplay among text, image, and reasoning, we employ a Tri-stream Cross-Reasoning Network (TCRNet) that applies pairwise scaled dot-product attention across all three modalities. The resulting attended representations are concatenated into a unified feature vector, which is passed to a final classification head. This pipeline, combining self-refined reasoning with multimodal fusion, demonstrates the power of hybrid, alignment-aware architectures in capturing the nuanced affective and figurative characteristics of dark humor in social media memes.

Key Contributions of this work are as follows:

- **Dark Humor Meme Dataset:** We introduce a first multimodal dark humor dataset consisting of memes collected from Reddit. Each entry includes both image and OCR-extracted text, annotated not only for binary *dark humor* versus *non-dark humor*, but also for *target category* (gender, mental health, violence, race, disability, and other) and a three-level *intensity* rating (mild, moderate, severe). This granular annotation enables detailed analysis of social and ethical dimensions in dark humor in memes.
- **Iterative Self-Refined Reasoning:** We use Qwen-2.5-7B VLM to generate an initial structured explanation (summary, implied joke, narrative, emotional effect, dark

attributes, target). Then, via a *Role-Reversal Self-Loop*, Qwen assumes the meme author’s perspective to critique and refine its own reasoning, ensuring higher alignment and completeness before feature extraction.

- **Reasoning-Enhanced Multimodal Framework:** We propose a multimodal architecture that fuses image embeddings, text embeddings, and embeddings of the self-refined Qwen reasoning through a Tri-stream Cross-Reasoning Network (TCRNet). Pairwise attention across the three streams captures implicit and semantic cues, yielding state-of-the-art performance on dark humor detection, target identification, and intensity prediction.

7.2 Dataset Description and Annotation Process

In this section, we describe the process of dataset collection and the annotation methodology employed. We detail the sources from which the data is gathered, the criteria used for selecting samples, and the steps taken to ensure annotation quality and consistency. This comprehensive overview provides the foundation for understanding the dataset’s structure and reliability for subsequent analysis.

7.2.1 Dataset Collection

All memes in our dataset were sourced from Reddit. We used the Reddit API to retrieve posts containing both an image and accompanying text, and applied OCR to extract embedded text. A subsequent manual screening eliminated the duplicates, resulting in a high-quality corpus of 4,379 multimodal memes for annotation and analysis.

7.2.2 Annotator Training

The annotation process was overseen by a faculty member and a PhD researcher with expertise in computational humor and harmful content analysis. The actual annotations were carried out by three undergraduate computer science students who voluntarily participated with informed consent. As a token of appreciation, they were granted access to A100 GPU resources for their own research.

7.2 Dataset Description and Annotation Process

Me after completing 1 of the 20 things I'm supposed to do



(a) Dark Humor: No
Target: N/A
Intensity: N/A

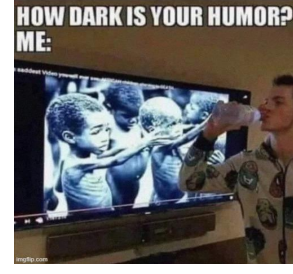


(b) Dark Humor: Yes
Target: Disability
Intensity: 2

9 hours into NoNutNovember and you bump shoulders with a female in the hallway



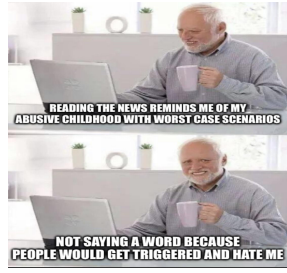
(c) Dark Humor: Yes
Target: Gender/Sex
Intensity: 3



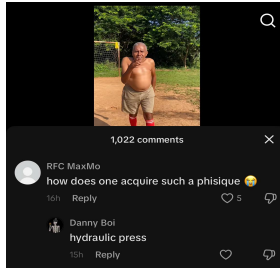
(d) Dark Humor: Yes
Target: Race/Ethnicity
Intensity: 3



(e) Dark Humor: Yes
Target: Gender/Sex
Intensity: 2



(f) Dark Humor: Yes
Target: Mental Health
Intensity: 2



(g) Dark Humor: Yes
Target: Disability
Intensity: 2



(h) Dark Humor: Yes
Target: Violence/Death
Intensity: 3

Figure 7.2: Example of Dark Humor, Target, and Intensity

A gold-standard seed set of 100 memes, balanced across dark and non-dark humor, was first annotated by the expert team. This dataset, along with a detailed codebook, was used to train the student annotators. The codebook defined annotation criteria with examples for three subtasks: (i) dark humor identification, (ii) target of the humor, and (iii) intensity level. After the initial labeling round, annotation discrepancies were discussed collectively to refine interpretation and improve consistency.

7.2.2.1 Annotation Dimensions

Each meme was annotated across three distinct dimensions:

- **Dark Humor (DH):** Binary label indicating whether the meme contains dark humor.
- **Target:** The specific group, topic, or subject that is the focus of the meme's humor (e.g., gender, mental health, disability).

- **Intensity:** The perceived intensity of the dark humor, annotated on a three-level ordinal scale:
 - *Mild (1)*: Humor that is subtly dark or lightly suggestive.
 - *Moderate (2)*: Humor that is more direct and clearly involves a sensitive theme.
 - *Severe (3)*: Highly explicit or extreme dark humor involving graphic or taboo content.

7.2.3 Annotation in Batch Mode

Following the training phase, the remaining memes were annotated in weekly batches of 200 memes. To mitigate potential cognitive fatigue from exposure to sensitive content, annotators were encouraged to take breaks of 10–15 minutes after each set of 10 memes, aligning with recommendations in prior work on annotator well-being [219]. We also enforced a soft cap of 50 memes per day to prevent overload. Regular check-in meetings were held to discuss any difficult cases, resolve annotation conflicts, and monitor the psychological impact of labeling dark content [22]. All annotations were finally reviewed by the supervising researchers to create a high-quality, adjudicated gold-standard dataset.

7.2.4 Inter-Annotator Agreement

To assess the consistency of annotations across annotators, we computed inter-annotator agreement scores using both pairwise Cohen’s Kappa and Fleiss’ Kappa. Table 7.1 summarizes the results for the three annotation tasks: *Dark Humor* (DH), *Target*, and *Intensity*.

For the dark humor task, unweighted Cohen’s Kappa scores ranged from 58.92 to 85.88 across annotator pairs, with a Fleiss’ Kappa of 70.29, indicating substantial agreement. The target task yielded similarly strong results, with pairwise Cohen’s Kappa scores as high as 83.55 and a Fleiss’ Kappa of 72.13, reflecting high agreement on identifying the meme’s intended subject.

The intensity task, which involves ordinal labels (mild, moderate, severe), naturally presents more subjectivity. We computed both unweighted and **weighted** Cohen’s Kappa for this task only, since weighted Kappa accounts for the degree of disagreement in ordinal annotations. While unweighted scores ranged from 45.89 to 76.59, the weighted Kappa

7.2 Dataset Description and Annotation Process

Table 7.1: Inter-annotator Agreement Scores of D-Humor Dataset. A: Annotator, DH: Dark Humor, Unwt: Unweighted, Wt: Weighted, C Kappa: Cohen’s Kappa, F Kappa: Fleiss’ Kappa

Task	Ann Pair	UnWt C Kappa	Wt C Kappa	F Kappa
DH	A1&A2	66.05	-	70.29
	A2&A3	85.88	-	
	A3&A1	58.92	-	
Target	A1&A2	70.31	-	72.13
	A2&A3	83.55	-	
	A3&A1	62.47	-	
Intensity	A1&A2	53.24	74.39	58.57
	A2&A3	76.59	87.26	
	A3&A1	45.89	69.45	

values were significantly higher, reaching up to 87.26. The Fleiss’ Kappa for intensity was 58.57, suggesting moderate agreement overall.

The use of annotator training, expert supervision, and structured annotation guidelines helped ensure consistent and replicable labeling across complex and sensitive humor categories.

7.2.5 Dataset Statistics

Table 7.2 presents a detailed breakdown of our dataset across both training and test splits. The dataset consists of meme samples labeled for the presence or absence of dark humor, along with fine-grained annotations on their target category and humor intensity. A sample of memes from the D-HUMOR dataset is shown in Figure 7.2.

7.2.5.1 Dark Humor Distribution

The training set contains a total of 3,503 memes, with 1,577 labeled as containing dark humor and 1,926 as non-dark humor. The test set comprises 876 memes, including 397 positive and 479 negative samples. This relatively balanced distribution supports effective supervised training and facilitates fair evaluation.

Table 7.2: Summary of D-Humor Dataset

Train Data		Test Data	
Dark Humor	Count	Dark Humor	Count
No	1,926	No	479
Yes	1,577	Yes	397

Train Data: Target vs Intensity					Test Data: Target vs Intensity				
Target	1	2	3	Total	Target	1	2	3	Total
Gender/Sex-Related Topics	303	248	132	683	Gender/Sex-Related Topics	59	74	28	161
Mental Health	169	49	4	222	Mental Health	46	9	3	58
Disability	30	26	19	75	Disability	9	7	3	19
Race/Ethnicity	32	35	19	86	Race/Ethnicity	5	7	6	18
Violence/Death	94	48	18	160	Violence/Death	31	13	3	47
Other	275	60	16	351	Other	79	12	3	94
Total	903	466	208	1,577	Total	229	132	46	397

7.2.5.2 Target Categories

Each meme labeled as dark humor was further annotated with the specific *target* of its humor. In the training set, the most frequently targeted categories are *Gender/Sex-Related Topics* (683 instances), followed by *Other* (351), and *Mental Health* (222). These three categories also dominate the test set distribution, indicating consistent prevalence across both data splits.

7.2.5.3 Intensity Levels

In the training data, the distribution is skewed toward lower intensity: 903 memes are rated as *Mild*, 466 as *Moderate*, and only 208 as *Severe*. A similar trend appears in the test set, with 229, 132, and 46 memes across the three levels, respectively. This distribution reflects the real-world observation that while most dark humor is implicit or moderately suggestive, highly explicit content is comparatively rare.

Overall, this rich annotation framework provides valuable granularity for developing and benchmarking models that aim to understand not only the presence of dark humor but also its target and intensity.

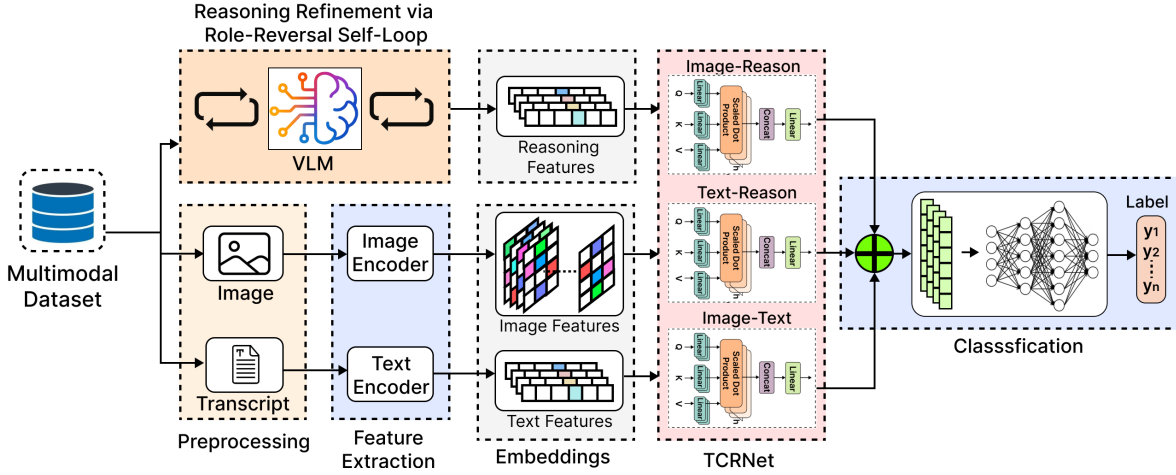


Figure 7.3: Architecture of Reasoning-augmented Multimodal Dark Humor Detection Method

7.3 Methodology

Understanding dark humor in memes requires reasoning beyond literal content, as it often relies on abstract incongruity between text and visuals. To address this, we propose a reasoning-augmented multimodal framework that leverages linguistic cues, image semantics, and inferred commonsense reasoning. The architecture, as shown in Figure 7.3, comprises five components: reasoning refinement, feature extraction, cross-attention fusion, feature aggregation, and final classification.

7.3.1 Structured Reasoning Generation

Understanding dark humor memes requires more than just identifying surface-level content. The humor often hinges on subtle context, cultural subtext, visual incongruities, and emotionally conflicting cues, many of which may be implicit, taboo, or deliberately ambiguous. These characteristics make dark humor memes uniquely challenging to interpret, both for humans and machines.

To address this, we enrich the meme classification process with structured explanatory signals. By explicitly breaking down the reasoning behind a meme’s humor, emotional tone, and target, we provide the model with additional context that aids in disambiguation. This additional supervision helps the model move beyond superficial patterns and capture the deeper intent and affective contrast that are typical in dark humor. Moreover, structured reasoning

allows for improved transparency and interpretability, supporting downstream applications like content moderation, harm detection, and bias analysis.

7.3.1.1 Structured Explanation Schema

We design each explanation using a six-part schema that captures distinct dimensions of the meme’s communicative intent and stylistic choices:

1. **Meme Summary:** A concise description of the meme’s image and text.
2. **Implied Joke:** The core humorous element inferred from the meme.
3. **Narrative Structure:** The rhetorical style (e.g., sarcasm, irony, absurdism).
4. **Emotional Effect:** The emotional response the meme is designed to elicit.
5. **Dark Attributes:** Any presence of taboo, offensive, or socially sensitive themes.
6. **Target:** The entity, group, or concept being referenced or mocked.

This decomposition facilitates fine-grained understanding, and the modular format can be encoded and fused with vision-language features during classification.

7.3.1.2 Iterative Reasoning Refinement via Role-Reversal Self-Loop

Interpreting dark humor requires not only visual-textual understanding but also alignment with nuanced social, cultural, and emotional cues, an area where even large models often fail. To bridge this gap, we propose a novel technique called **Role-Reversal Self-Loop**, a simple yet powerful mechanism for enhancing *AI alignment* in tasks involving implicit reasoning and culturally sensitive content.

After generating an initial structured reasoning using the Qwen-2.5-7B Vision-Language Model (VLM), we prompt the model to assume a reversed role: *it becomes the original author of the meme*. Under this setup, the model is asked to review the reasoning it “received” for its own meme and critically assess whether the fields are faithful and complete as shown in Figure 7.4. This role-flip is psychologically aligned with reflective human behavior, encouraging the model to evaluate explanations with a self-interested perspective, thereby improving alignment.

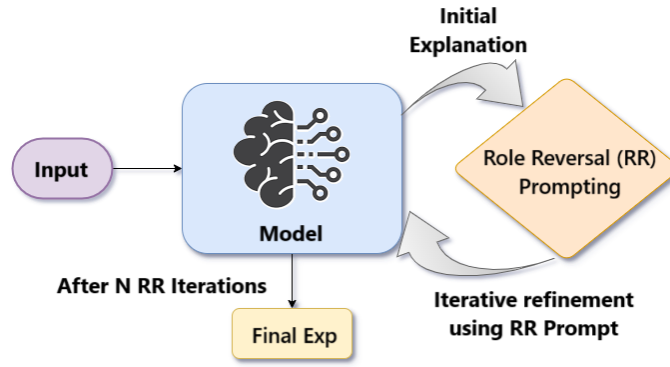


Figure 7.4: Self-loop Mechanism using Role Reversal (RR) Prompting

This approach has two key benefits:

- It ensures that each explanation component is reviewed in context, preventing the model from over-focusing on only one field (e.g., the joke or the target).
- It encourages introspective correction without requiring external feedback or manual interventions, thereby improving scalability and automation.

We apply this refinement loop iteratively for a maximum of 5 epochs. On average, explanations converge and stabilize by approximately 3 epochs, indicating that the model typically needs only a few rounds to self-correct and align its output. Importantly, our manual review of a subset of explanations revealed that most improvements were concentrated in the fields of Narrative Structure, Dark Attributes, and Implied Joke, components that demand deeper contextual reasoning. Fields like Meme Summary, Target, and Emotional Effect were more stable across iterations, suggesting they are more straightforward to interpret initially.

Role-Reversal self-loop prompt template is as follows:

<You are the author and creator of the given dark humor meme. Someone else has provided a detailed explanation of your meme’s meaning, humor, and components, including the Meme Summary, Implied Joke, Narrative Structure, Emotional Effect, Dark Attributes, and Target. Your task is to review their explanation from your perspective as the original author. Analyze how well their reasoning aligns with your intended humor, message, tone, and overall context. For each component, identify any inaccuracies, missing details, or misunderstandings. Then, provide a thorough, corrected explanation that fully reflects your original intent, ensuring clarity, accuracy, and completeness.

Please present your revised explanation below.>

7.3.2 Feature Extraction

We extract embeddings from three modalities: the meme’s original transcript T , the generated reasoning R , and the image I .

Textual Features: We use the BERT model to encode both T and R into contextual token embeddings:

$$\mathbf{F}_T = \text{BERT}(T) \in \mathbb{R}^{197 \times 768} \quad (7.1)$$

$$\mathbf{F}_R = \text{S-BERT}(R) \in \mathbb{R}^{197 \times 768} \quad (7.2)$$

Each row corresponds to a subword token in a 768-dimensional embedding space.

Visual Features: We use the ViT model, which combines vision and language pretraining objectives, to extract features from the meme image:

$$\mathbf{F}_I = \text{ViT}(I) \in \mathbb{R}^{197 \times 768} \quad (7.3)$$

This representation captures visual cues such as objects, expressions, and contextual artifacts relevant to the meme.

7.3.3 Tri-stream Cross-Reasoning Network (TCRNet)

To effectively model inter-modal interactions and capture reasoning-driven dependencies, we introduce a *Tri-stream Cross-Reasoning Network* [220, 221]. This module enables each modality, text (T), image (I), and LLM-generated reasoning (R), to mutually inform and align with one another by attending across pairs.

Specifically, we use scaled dot-product attention to implement pairwise reasoning flows. Let $\mathbf{Q}_X = \mathbf{F}_X \mathbf{W}_Q$, $\mathbf{K}_Y = \mathbf{F}_Y \mathbf{W}_K$, and $\mathbf{V}_Y = \mathbf{F}_Y \mathbf{W}_V$ be the linear projections of modalities X and Y , where $X, Y \in \{T, I, R\}$. The attention from X to Y is computed as:

$$\text{Attn}(X \rightarrow Y) = \text{softmax} \left(\frac{\mathbf{Q}_X \mathbf{K}_Y^\top}{\sqrt{d_k}} \right) \mathbf{V}_Y \quad (7.4)$$

This formulation allows modality X to selectively extract context-aware signals from

modality Y , thereby modeling semantic incongruity, visual-verbal contradictions, and inferential alignment, all of which are critical in dark humor.

We compute such cross-reasoning across all three modality pairs:

$$\tilde{\mathbf{F}}_{T \leftrightarrow I} = \text{Attn}(T \rightarrow I) \quad (7.5)$$

$$\tilde{\mathbf{F}}_{T \leftrightarrow R} = \text{Attn}(T \rightarrow R) \quad (7.6)$$

$$\tilde{\mathbf{F}}_{I \leftrightarrow R} = \text{Attn}(I \rightarrow R) \quad (7.7)$$

7.3.4 Feature Aggregation

Each cross-reasoned output is aggregated using average pooling to yield fixed-length vectors:

$$\mathbf{v}_{T \leftrightarrow I} = \text{AvgPool}(\tilde{\mathbf{F}}_{T \leftrightarrow I}) \quad (7.8)$$

$$\mathbf{v}_{T \leftrightarrow R} = \text{AvgPool}(\tilde{\mathbf{F}}_{T \leftrightarrow R}) \quad (7.9)$$

$$\mathbf{v}_{I \leftrightarrow R} = \text{AvgPool}(\tilde{\mathbf{F}}_{I \leftrightarrow R}) \quad (7.10)$$

We concatenate the resulting vectors to obtain a unified representation:

$$\mathbf{v} = [\mathbf{v}_{T \leftrightarrow I}; \mathbf{v}_{T \leftrightarrow R}; \mathbf{v}_{I \leftrightarrow R}] \in \mathbb{R}^{2,304} \quad (7.11)$$

This vector serves as the multimodal reasoning-aware representation fed into the final classification layer.

7.3.5 Hyperparameters and System Description

Table 7.3 lists the primary hyperparameters used for the proposed framework. Encoder-only text and image models share identical training configurations, while LLMs fine-tuned with QLoRA use 3 epochs and low-rank adaptation settings ($r = 8$, $\alpha = 32$, dropout 0.1). The TCRNet additionally employs dropout (0.3) and eight attention heads in its training setup.

All experiments were conducted on a server equipped with Intel(R) Xeon(R) Silver 4316 CPU @ 2.30GHz, 4× NVIDIA A16 GPUs (each with 16 GB VRAM), running CUDA 12.8 and driver version 570.133.07. The training utilized PyTorch with GPU acceleration, and

Table 7.3: Key Hyperparameter Configurations in D-Humor

Model Type	Hyperparameter	Value
Encoder-Only Text Models	Epochs	5
	Batch Size	16
	Learning Rate	0.00002
LLMs with QLoRA	Epoch	3
	LoRA Rank (r)	8
	LoRA Alpha	32
	LoRA Dropout	0.1
	Target Modules	q_proj, v_proj
Image Models	Epochs	5
	Batch Size	16
	Learning Rate	0.00002
TCRNet	Epochs	5
	Batch Size	16
	Learning Rate	0.00002
	Dropout	0.3
	Attention Heads	8

computations were distributed using a single GPU unless specified otherwise.

7.4 Results and Discussion

Table 7.4: Performance Analysis of Different Methods on D-Humor Dataset

Model	Dark Humor (Yes/No)			Target			Intensity			
	Accuracy	Macro-F1	Wt-F1	Accuracy	Macro-F1	Wt-F1	Accuracy	Macro-F1	Wt-F1	P-Corr.
OCR Text Only										
BERT [100]	70.05	69.41	69.75	59.69	53.27	59.53	58.91 [†]	46.59 [†]	57.63 [†]	27.18
Distil-BERT [222]	69.81	69.23	69.55	61.50	55.98	60.91	55.81	45.75	55.22	26.25
Llama-3.2-3b-it [223]	71.72	71.14	71.45	54.52	38.20	51.96	57.62	37.41	52.58	17.05
Image Only										
CLIP [160]	57.76	57.69	57.85	37.53	35.43	38.47	40.30	31.18	40.72	2.40
ViT [224]	64.04	62.82	63.45	45.59	32.76	39.86	54.66	36.22	51.14	7.14
BLIP [166]	64.95	64.25	64.72	44.84	40.53	43.79	58.19	41.69	54.40	19.96
VLMs (Prompt based Zero-Shot)										
Qwen-2.5-7b-it [225]	64.27	63.26	63.83	57.68	53.70	56.28	17.63	15.27	14.14	17.50
LLaVA-1.6-mistral-7b [226]	45.32	31.19	28.27	48.87	48.30	49.14	30.23	24.22	29.92	11.48
OCR Text + Structured Explanation (with Role-Reversal Prompting)										
BERT	71.24	70.88	71.13	65.37 [†]	61.67 [†]	65.42 [†]	56.33	43.16	55.10	32.12
Distil-BERT	70.53	70.02	70.32	66.41	62.53	66.07	55.56	43.98	54.95	32.56 [†]
Llama-3.2-3b-it	73.39 [†]	72.91 [†]	73.19 [†]	59.17	50.37	57.62	58.14	37.94	53.76	24.4
VLMs (Prompt based Zero-Shot with Role-Reversal Prompting)										
Qwen-2.5-7b-it	64.04	64.02	63.95	55.67	50.97	55.01	29.22	28.97	29.33	11.84
LLaVA-1.6-mistral-7b	46.80	35.40	32.86	49.62	51.44	50.03	37.03	34.41	39.35	15.22
TCRNet (ours)	75.00	73.55	74.13	64.48	60.54	64.17	62.72	49.71	60.74	38.63

7.4.1 Experimental Findings

Table 7.4 presents a comprehensive evaluation of various models on three tasks: dark humor detection (binary classification), target identification (multi-class classification), and intensity prediction. We report Accuracy, Macro-F1, and Weighted-F1 for all tasks, and additionally include Pearson correlation for intensity prediction, as it is ordinal. When relying solely on OCR-extracted text, pretrained models like BERT, DistilBERT, and LLaMA achieve reasonable performance for dark humor detection. LLaMA leads with 71.72% accuracy and a Macro-F1 of 71.14%. However, these text-only systems struggle to identify the specific target of the humor and to gauge its intensity. For instance, DistilBERT’s target identification Macro-F1 is only 55.98%, and BERT’s intensity Pearson correlation is a modest 27.18%. These results indicate that while textual cues carry considerable information for detecting the presence of dark humor, they lack the broader context and implicit signals needed to pinpoint whom or what the joke is aimed at, and how severe it is.

Purely image-based models, such as CLIP, ViT, and BLIP, underperform compared to text-only baselines. BLIP achieves the highest image-only dark humor accuracy at 64.95%, but target identification accuracy remains below 45% and intensity accuracy under 60%. The Pearson correlation for intensity is especially low, for instance, 19.96% for BLIP. This suggests that, although visual features capture facial expressions, color palettes, and scene context, they fail to convey the nuanced semantics that define dark humor. In other words, imagery alone lacks the explicit linguistic content that often carries the punchline or the taboo reference.

Zero-shot vision-language models, such as Qwen-VL and Next-LLaVA-VL, demonstrate limited effectiveness without task-specific fine-tuning. Qwen-VL achieves 64.27% dark humor accuracy and a 17.50% intensity correlation, while Next-LLaVA-VL lags further behind (45.32% accuracy for humor detection, 11.48% intensity correlation). Even when provided with structured explanations as prompts, these VLMs show only marginal improvements. For example, Next-LLaVA-VL’s intensity correlation rises from 11.48% to 15.22% with explanations, but remains far below the performance of models trained on text-plus-explanation inputs. These observations highlight that, although large VLMs contain rich pretraining knowledge, they still require supervised adaptation to handle the implicit, sensitive nature of

dark humor.

Augmenting OCR text with structured explanations yields consistent and substantive gains across all tasks. Under the “OCR Text + Structured Explanation” setting, BERT’s dark humor accuracy increases from 70.05% to 71.24%, and DistilBERT’s target Macro-F1 jumps from 55.98% to 62.53%. Most notably, intensity Pearson correlation for DistilBERT improves from 26.25% to 32.56%. These results underline that structured explanations, capturing implied jokes, narrative structure, and emotional effect, supplement raw text with crucial contextual cues. Even LLaMA, which already performs well with text alone, sees its dark humor accuracy rise to 73.39% when explanations are included, confirming that reasoning signals complement large language model outputs.

Our proposed **TCRNet** model, which implements a Tri-stream Cross-Reasoning Network to fuse OCR text, image features, and structured explanations, achieves the best overall results. TCRNet attains 75.00% accuracy on dark humor detection and 62.72% accuracy on intensity prediction, both the highest scores among all evaluated methods. By explicitly aligning visual and textual streams with reasoning-derived explanations, TCRNet captures implicit associations and semantic incongruities that single-modality or non-reasoning models miss. These improvements demonstrate that reasoning-aware, multimodal fusion is essential for accurately understanding and categorizing dark humor memes.

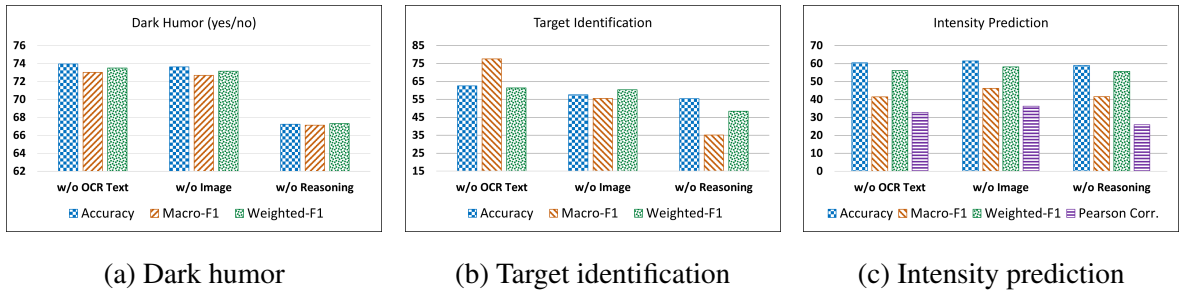


Figure 7.5: Ablation Study of TCRNet on Dark Humor Meme Analysis

7.5 Ablation Study

From the ablation study presented in Figure 7.5, it is evident that the reasoning component based on structured explanations plays a critical role in enhancing the performance of

TCRNet across all three tasks. When the explanation modality is removed, there is a substantial drop in performance, particularly in Target Identification (Macro-F1 drops from 60.54 to 35.11) and Dark Humor Classification (Weighted-F1 drops from 74.13 to 67.31). This clearly indicates that dark humor memes often rely on implicit, nuanced reasoning that goes beyond surface-level textual or visual cues.

Unlike typical classification problems, where raw OCR text or image features might suffice, dark humor detection demands an externalized and structured interpretive view. The explanation-based features help the model contextualize morally ambiguous content, sarcasm, or deception, which are often difficult to infer directly from the raw input. These findings emphasize the importance of integrating explicit reasoning mechanisms into multimodal understanding systems, especially for complex, subjective tasks like dark humor analysis.

7.6 Conclusion

In this work, we introduced the D-Humor dataset, a multimodal collection of 4,397 Reddit memes annotated for dark humor (yes/no), target category (gender, mental health, violence, race, disability, other), and a three-level intensity rating (mild, moderate, severe). We further proposed a reasoning-augmented framework that first generates and self-refines structured explanations using Qwen-2.5-7B VLM, then fuses OCR text, ViT-based image features, and refined reasoning via a Tri-stream Cross-Reasoning Network (TCRNet). Experimental results across three tasks, dark humor detection, target identification, and intensity prediction, demonstrate that (1) text-only or image-only models are insufficient for capturing the nuances of dark humor, (2) zero-shot vision-language models underperform without fine-tuning, and (3) augmenting OCR text with structured explanations yields consistent gains.

Our proposed TCRNet achieves state-of-the-art performance, attaining 75.00% accuracy on dark humor detection and 62.72% accuracy on intensity prediction, outperforming all baselines by a significant margin. An ablation study confirms that the structured explanation component is critical: removing explanations leads to a dramatic drop in Target Identification Macro-F1 (from 60.54% to 35.11%) and Dark Humor Weighted-F1 (from 74.13% to 67.31%). These findings highlight that dark humor in memes relies on implicit, context-

dependent cues, such as implied jokes and emotional effect, that cannot be inferred from raw text or images alone, necessitating explicit reasoning signals.

While we acknowledge that D-Humor is skewed toward the *Gender/Sex-Related Topics* label, likely reflecting the greater availability of such content online, models trained on this data generalize well across all categories and do not exhibit overfitting. For future work, we plan to expand the dataset with additional target classes to enable finer-grained analyses, as well as investigate semi-supervised and self-supervised techniques to further improve generalization to emerging dark humor trends. We also aim to explore more advanced alignment mechanisms within the Tri-stream Cross-Reasoning framework to better capture evolving cultural and social nuances in online humor.

Chapter 8

Hostile Content Detection in Videos

8.1 Introduction

Following the analysis of hostile content in textual and image-based modalities, this chapter advances to the detection of implicit hate speech in videos. Unlike explicit abuse, implicit hate in videos is often conveyed through subtle visual cues, tone of speech, context, and cross-modal interactions rather than overtly hateful language. This complexity necessitates multimodal frameworks that can jointly model audio, visual, and textual signals to uncover latent hostile intent.

While some studies have explored hate detection in videos [87–89], implicit hate speech detection in videos remains an underexplored area. To the best of our knowledge, we are the first to work in implicit hate speech detection in videos. Implicit hate speech is defined as expressions that communicate discriminatory or prejudiced views indirectly, often through coded language, implied meanings, or contextual cues [18]. Unlike overt hate speech, it subtly evades detection by adhering to platform guidelines and may appear innocuous superficially, yet still perpetuates harm or offense. Given the dominance of video content in digital communication, there is a strong need to develop specialized hate detection mechanisms for videos.

Figure 8.1 illustrates examples of both implicit and explicit hate speech in video content. In the implicit hate example, the hateful intent is conveyed through the underlying context and the association of political ideology with aggression. Such indirect expressions make implicit hate speech particularly challenging to detect, as they require an understanding of

8.1 Introduction

Offensive content warning: The presented image is just for illustration. We do not support or promote the views presented in the image.

Video Frame	Transcript	Label
	That's even the best thing to make of it. He'll show exactly why liberals get away with political violence.	Implicit Hate
	... black people apparently are dark stained, love chickens. I am mortified ...	Explicit Hate

Figure 8.1: Implicit and Explicit Hate Videos Example

both textual and visual cues in the video. Conversely, the explicit hate speech example features a cartoonish character who appears to be shouting a racial slur directed at Black individuals. This distinction underscores the necessity of advanced multimodal methods for effective implicit hate speech detection in videos.

In this work, we address the critical challenge of detecting hateful content in videos by proposing a novel method with a particular focus on implicit hate speech, which is often subtle and context-dependent. Our two-stage contrastive learning method first extracts robust, modality-specific features from text, images, and audio, and then aligns them in a shared embedding space using cross-modal encoders with projection heads and supervised contrastive loss. This approach captures subtle cues of implicit hate speech that traditional unimodal or simple fusion methods often miss. By leveraging complementary multimodal information, our framework provides a richer, more nuanced representation for hate speech detection in videos. Our key contributions are as follows:

- We introduce a new dataset specifically curated for implicit hate speech detection in videos. The dataset consists of 2,009 videos and provides a valuable benchmark for future research on multimodal hate speech detection.
- We propose a two-stage contrastive learning approach to effectively model multimodal hateful content in videos. In the first stage, we train three modality-specific encoders (audio, text, and image) using contrastive loss, computed over concatenated feature representations. In the second stage, we train a cross-encoder using contrastive learn-

ing to refine multimodal representations further.

- We evaluate our approach on both our newly curated dataset and the publicly available HateMM dataset. The results demonstrate the effectiveness of our proposed multimodal contrastive learning framework in detecting hateful content in videos, particularly implicit hate speech.

8.2 Data Collection and Annotation

8.2.1 Platforms for Video Collection

For data collection we primarily used BitChute¹, a social video-hosting platform launched in 2017 with minimal content moderation. It has grown in prominence as an alternative to YouTube, hosting a significant amount of hateful content banned from mainstream platforms. In addition to BitChute, we also collected videos from Odysee², another alternative video-sharing platform.

8.2.2 Dataset Statistics

Our dataset consisted of 2,009 videos comprising 86.5 hours of English multimodal content collected from BitChute and Odysee. Among these, 1,000 videos were classified as non-hate, while the remaining 1,009 videos contained hateful content. The hate videos were further categorized into implicit hate and explicit hate, with 509 and 500 videos, respectively, to capture different degrees of hateful expressions. The details of the dataset can be seen in Table 8.1. The dataset is balanced with almost 50% hate and non-hate content by count. Duration-wise, we have 47 hours of hate content compared to 39.5 hours of non-hate content. Within the hate category, the number of samples of implicit and explicit hate is also roughly balanced. On average, all the videos are approximately 2 minutes long, with approximately 143 frames captured per video. Non-hate videos have a little more than double the number of words in transcripts than those in implicit and explicit hate videos.

¹<https://www.bitchute.com/>

²<https://odysee.com/>

8.2 Data Collection and Annotation

Table 8.1: ImpliHateVid Dataset Statistics

Properties	Non Hate	Implicit Hate	Explicit Hate
Video count	1,000 (49.78%)	509 (25.36%)	500 (24.89%)
Total length	39 hours 26 mins 42 secs	18 hours 7 mins 51 secs	28 hours 58 mins 25 secs
Mean video length	2 mins 22 secs	2 mins 8 secs	2 mins 38 secs
Total number of frames	1,42,002	65,271	79,105
Mean number of frames	142.002	128.23	158.21
Mean number of words	175.404	85.166	80.326

8.2.3 Annotation Guidelines

The following labeling scheme provided the main framework for annotators, while a code-book ensured consistency in label interpretation as inspired by Das et al. [87], Wang et al. [88], and Salles et al. [143]. Developed using YouTube’s hate speech policy, the code-book contains detailed annotation guidelines. A video is considered hateful if:

“It promotes discrimination, disparages, or humiliates an individual or group based on characteristics such as race, ethnicity, nationality, religion, disability, age, veteran status, sexual orientation, or gender identity.”

Moreover, the annotators were guided to pinpoint particular segments of a hate video (i.e., frame spans) they deemed hateful and indicate the specific communities targeted by the content.

8.2.4 Annotation Process

Annotator Training

The annotation process was supervised by one Professor and one PhD student with expertise in analyzing harmful content on social media, while the actual annotations were carried out by 1 postgraduate and three undergraduate students who were novice annotators. All annotators were computer science majors and participated voluntarily with full consent. As a token of appreciation, they were rewarded with free access to A100 GPU for 150 hours.

The annotators were trained by creating an initial gold-standard dataset. The expert annotators labeled 50 videos, comprising 30 hate videos and 20 non-hate videos which were then provided to the undergraduate annotators for labeling based on the annotation codebook. After completing their annotations, we reviewed and discussed the incorrect cases with them to refine their understanding and improve their annotation accuracy.

Annotation in Batch Mode

Following the initial training, we adopted a batch-mode approach, releasing a set of 50 videos per week for annotation. Given the potential negative psychological effects of annotating hate content [219], we advised annotators to take a minimum 10-15 minute break after labeling each video. Additionally, we imposed a strict limit of no more than 20 videos per day to prevent cognitive overload. To ensure the well-being of the annotators, we also conducted regular check-in meetings to monitor any potential adverse effects on their mental health.

8.3 Methodology

We propose a two-stage contrastive learning multimodal framework for detecting hateful content in videos as shown in Figure 8.2. Our approach comprises three key steps: (i) preprocessing to extract audio, text, and visual data; (ii) feature extraction using modality-specific encoders; and (iii) contrastive learning to align representations, culminating in a multimodal classifier.

8.3.1 Preprocessing

Videos are converted to WAV audio using FFmpeg and transcribed via speech-to-text conversion. For the visual modality, 100 frames are uniformly sampled (with padding for videos having fewer frames) using VideoCapture, ensuring consistent input dimensions across samples.

8.3.2 Feature Extraction

We extract features from image, text, and audio using a video-based contrastive learning method, ImageBind [227], that maps raw inputs into a shared 1024-dimensional space. We denote the extracted features with f_I , f_T , and f_A for the image, text and audio, respectively.

In addition to extracting the 1,024-dimensional audio, text, and video features, we extract complementary features to enrich our multimodal representation. For a text transcription x_T , we compute emotion features $e = \text{NRCLex}(x_T)$ (with $e \in \mathbb{R}^{d_e}$) and a sentiment score $s = \text{Vader}(x_T)$ ($s \in \mathbb{R}$); these are concatenated into a joint representation $f_{ES} = [e, s] \in \mathbb{R}^{d_e+1}$.

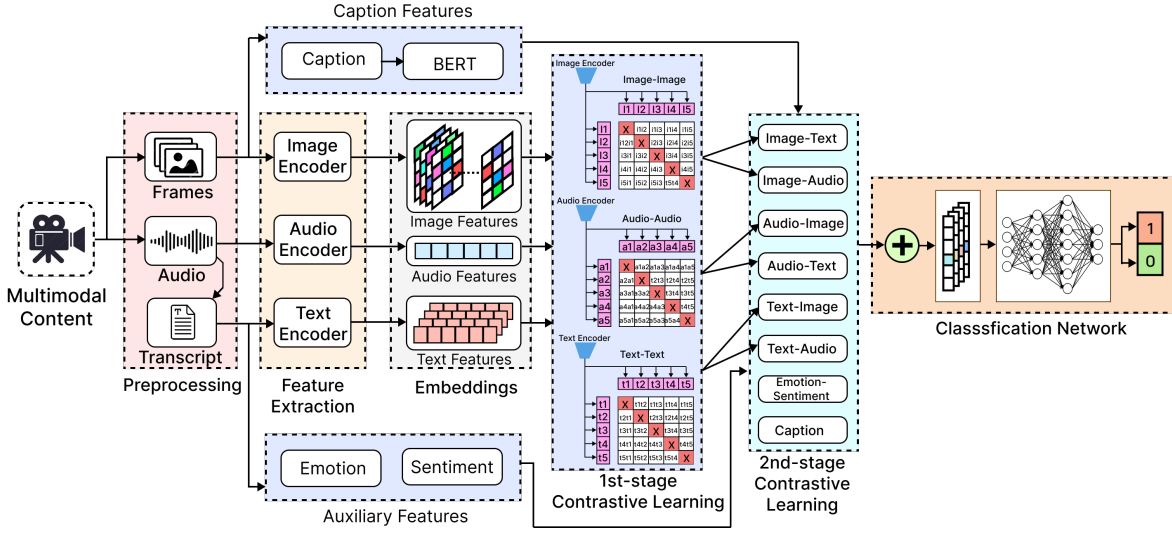


Figure 8.2: ImpliHatevid Method Architecture

Similarly, for an image x_I , we generate a caption $c = \text{caption_gen}(x_I)$ and extract caption features $f_C = \text{BERT}(c)$ where $f_C \in \mathbb{R}^{d_c}$. Together, these features complement the primary modality representations to provide a comprehensive view of the content.

8.3.3 Two-stage Contrastive Learning

8.3.3.1 Stage 1: Modality-specific Encoder Training

To jointly optimize the image, text, and audio encoders from the first stage, denoted by encoder_{II} , encoder_{TT} , and encoder_{AA} , we merge their outputs via a projection head. Let the encoder outputs be defined as f_{II} , f_{TT} , and f_{AA} for the image, text, and audio modalities, respectively, from the first stage encoders. These features, each of dimension d (e.g., 1,024), are concatenated to form a joint representation:

$$f_{ITA} = \text{Concat}(f_{II}, f_{TT}, f_{AA}) \in \mathbb{R}^{3d} \quad (8.1)$$

A projection head $P : \mathbb{R}^{3d} \rightarrow \mathbb{R}^{d'}$ is then applied to map the concatenated features into a shared embedding space:

$$z_{ITA} = P(f_{ITA}) \in \mathbb{R}^{d'} \quad (8.2)$$

The merged encoder (denoted as ITA) is optimized using a supervised contrastive loss. Specifically, for a batch of N samples, the loss is defined as:

$$\mathcal{L}_{\text{sup}} = -\frac{1}{N} \sum_{i=1}^N \log \frac{\sum_{j \in \mathcal{P}(i)} \exp\left(\frac{\text{sim}(z_{ITA}^i, z_{ITA}^j)}{\tau}\right)}{\sum_{k \in \mathcal{N}(i)} \exp\left(\frac{\text{sim}(z_{ITA}^i, z_{ITA}^k)}{\tau}\right) + \epsilon} \quad (8.3)$$

where: $\text{sim}(z_{ITA}^i, z_{ITA}^j)$ is the cosine similarity between embeddings, τ is the temperature parameter, ϵ is a small constant for numerical stability, $\mathcal{P}(i)$, and $\mathcal{N}(i)$ denote the sets of indices corresponding to positive (same class) and negative (different classes) pairs for sample i , respectively. In this way, the individual encoders encoder_{II} , encoder_{TT} , and encoder_{AA} are optimized jointly using the merged ITA representation, thereby aligning multimodal features in a unified embedding space. Moreover, specialized encoders for emotion-sentiment (ES) and image captions (CP) are trained in a similar manner.

8.3.3.2 Stage 2: Cross-Modal Encoder Training

In this stage, we align representations across modalities by training cross-modal encoders that merge outputs from modality-specific encoders via a projection head. We can understand this through an example of image_text cross encoders. The features from 1st-stage encoders are first processed by cross encoders encoder_{IT} and encoder_{TI} to produce refined features f_{IT} and f_{TI} :

$$f_{IT} = \text{encoder}_{IT}(f_{II}) \quad (8.4)$$

$$f_{TI} = \text{encoder}_{TI}(f_{TT}) \quad (8.5)$$

Next, a projection head $P(\cdot)$, implemented as a dense layer with ReLU activation, maps these to a shared embedding space:

$$z_{IT} = P(f_{IT}) \quad (8.6)$$

$$z_{TI} = P(f_{TI}) \quad (8.7)$$

The embeddings are concatenated to form the cross-modal representation:

$$z_{\text{cross}} = \text{Concat}(z_{IT}, z_{TI}) \quad (8.8)$$

Subsequently, the cross-modal encoder is optimized using a supervised contrastive loss. In addition to the image_text encoders, we similarly train cross encoders for image_audio (IA and AI) and text_audio (TA and AT) pairs.

The overall training objective for our system combines the losses from the first stage (modality-specific encoders), the cross-modal encoders, and the specialized ES and CP encoders:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{stage 1}} + \mathcal{L}_{\text{stage 2}} + \mathcal{L}_{\text{sup}}^{ES} + \mathcal{L}_{\text{sup}}^{CP} \quad (8.9)$$

By jointly optimizing these components, the framework effectively aligns and integrates multimodal features from image, text, and audio. This unified representation enhances the model's ability to capture complementary information, thereby improving performance in tasks such as detecting hateful content in videos.

8.3.4 Multimodal Classification

The learned representations f_{IT} , f_{IA} , f_{TI} , f_{TA} , f_{AI} , f_{AT} , f_{ES} , and f_{CP} are concatenated to form a unified feature F . This vector is passed through dense layers with ReLU activations and dropout regularization to produce the final prediction:

$$y = \text{softmax}\left(W_4 \text{Dropout}\left(\sigma\left(W_3 \text{Dropout}\left(\sigma\left(W_2 \text{Dropout}\left(\sigma\left(W_1 F + b_1\right) + b_2\right) + b_3\right) + b_4\right)\right)\right) \quad (8.10)$$

where $y \in \mathbb{R}^C$ represents the predicted probability distribution over C classes.

This streamlined framework effectively integrates multimodal data and contrastive learning to improve the detection of hateful content in videos.

8.4 Results

8.4.1 Datasets

In addition to the proposed ImpliHateVid dataset, we used HateMM [87], another multimodal dataset, to evaluate the performance of our model. HateMM consists of 1,083 videos in total, comprising 43.26 hours of multimodal content in English collected from BitChute and Odysee platforms. Out of the 1,083 videos, 431 videos have been labeled as hate videos and 652 videos have been labeled as non-hate videos. However, we used only 1,035 videos out of 1,083 for our study.

8.4.2 Experimental Setup

We split both datasets into train, validation, and test sets containing the total videos, respectively. The HateMM dataset had 662, 166, and 210 samples in training, validation, and test sets, respectively, while our dataset, ImpliHateVid, had 1,283 samples in the training set, 325 samples in the validation set, and 401 samples in the test set. We experimented with several combinations of hyperparameters. 32 and 64 were used as batch sizes. Our learning rate was in the range $1e-3$, $1e-4$, $1e-5$, and we trained our model for 30, 50, 75, and 100 epochs. Adam optimizer [228] was used. We used Accuracy (Acc), Precision (Prec), Recall (Rec), F1-score (F1), and Macro-F1 metrics to evaluate the performance. The experiments were carried out using a DGX A100 server featuring an AMD EPYC 7742 64-Core Processor, 1000 GB RAM, and a 40 GB GPU.

8.4.3 Compared Methods

We have compared our proposed method with several unimodal and multimodal methods to demonstrate the effectiveness of our approach. For the textual modality, we evaluated the performance of **BERT** [229] alongside large language models (LLMs), including **GPT-4o** [230] and **Llama 3.1-8b** [223]. For the vision modality, we utilized **ViT** [231] and **ViViT** [232], while for the audio modality, we employed **MFCC** [233] and **Wav2Vec2** [234]. To assess video performance holistically, we also incorporated vision-language models such as

8.4 Results

GPT-4³ and **Llama-VL** [235]. Features extracted from these models (excluding GPT and Llama) were fed into a feedforward neural network for classification, comprising four dense layers with 512, 256, 128, and 64 neurons, respectively, and a dropout rate of 0.3. GPT and Llama models were used using APIs with zero-shot prompting. Additionally, we compared our proposed method against other multimodal approaches, including **DeepCNN** [236] and **CMHFM** [237]. The model proposed by Li et al. [212] is denoted as **CSID**, while the one by Li et al. [238] is labeled as **MCMF** in our results. We also examined the performance of the Transformer-based model **MuT** [239] across both datasets.

8.4.4 Effectiveness Comparison

8.4.4.1 Binary Classification

Table 8.2 highlights the performance improvements of our proposed multimodal method over the best performing approaches in each category on the ImpliHateVid and HateMM datasets. On the ImpliHateVid dataset, our method achieves an F1-score of 87.73%, which is approximately 10.5% higher than the best unimodal method, Wav2Vec2, which attains an F1-score of 77.24%. When compared to the leading multimodal baseline, MuT, with an F1-score of 83.52%, our approach shows a 4.21% improvement.

Table 8.2: Effectiveness Comparison for Binary Classification

Modality	Method	ImpliHateVid Dataset				HateMM Dataset			
		Acc	F1	Prec	Rec	Acc	F1	Prec	Rec
Text	BERT	0.6907	0.6884	0.6907	0.6907	0.7350	0.6640	0.6750	0.6670
	GPT-4o	0.5312	0.1132	1.0000	0.0600	0.5652	0.1964	0.3793	0.1325
	Llama 3.1-8b	0.5312	0.2034	0.6667	0.1200	0.5459	0.2540	0.3721	0.1928
Image	ViT	0.7655	0.7684	0.7658	0.7656	0.7480	0.6720	0.6950	0.6560
	ViViT	0.4912	0.5255	0.4912	0.4914	0.5293	0.5176	0.5172	0.5182
Audio	MFCC	0.4987	0.6655	0.2493	0.5000	0.6750	0.6220	0.5930	0.6790
	Wav2Vec2	0.7531	0.7724	0.7610	0.7533	0.5810	0.5810	0.5270	0.5160
Video	GPT-4	0.4988	0.6656	0.4988	1.0000	0.4010	0.5724	0.4010	1.0000
	LlamaVL	0.4010	0.5724	0.4010	1.0000	0.3800	0.5300	0.3700	0.9500
Multimodal	DeepCNN	0.7623	0.7800	0.7481	0.7933	0.5622	0.3065	0.4565	0.2307
	CMHFM	0.7922	0.7921	0.7860	0.7980	0.6057	0.5629	0.6184	0.6184
	CSID	0.8150	0.8154	0.8082	0.8233	0.7320	0.7140	0.7200	0.7230
	MCMF	0.8224	0.8220	0.8200	0.8240	0.5769	0.0435	0.5000	0.2422
	MuT	0.8352	0.8352	0.8320	0.8380	0.6571	0.5212	0.4318	0.6571
	Proposed Method	0.8753	0.8773	0.8796	0.8752	0.9758	0.9758	0.9745	0.9710

On the HateMM dataset, the differences are even more pronounced. Our method reaches

³https://cdn.openai.com/papers/GPTV_System_Card.pdf

an F1-score of 97.58%, outperforming the best text-based model, BERT, which records an F1-score of 66.40%, by roughly 31.18%. Similarly, against the top multimodal method in this category, CSID, with an F1-score of 71.40%, our proposed approach gains about 26.18%. These substantial gains in F1-score clearly demonstrate the effectiveness of our multimodal framework in leveraging complementary cues from text, image, and audio modalities, thereby significantly enhancing implicit hate speech detection across different datasets.

8.4.4.2 Multiclass Classification

Table 8.3 presents the multiclass classification results across non-hate, implicit hate, and explicit hate video categories. Our proposed method achieves an overall macro-F1 of 69.18%, which represents a significant improvement over the best unimodal approaches and existing multimodal baselines. For instance, among unimodal models, the text-based BERT attains a macro-F1 of 59.07%, while the image-based ViT and audio-based Wav2Vec2 achieve 58.83% and 57.16%, respectively. This indicates that our model improves the macro-F1 score by roughly 10.1% points over the best unimodal method. GPT and Llama fail to identify instances of implicit hate accurately due to noisy transcriptions.

Table 8.3: Effectiveness Comparison for Multiclass Classification

Modality	Method	Non Hate Videos				Implicit Hate Videos				Explicit Hate Videos				Overall Macro-F1
		Acc	F1	Prec	Rec	Acc	F1	Prec	Rec	Acc	F1	Prec	Rec	
Text	BERT	0.7195	0.7192	0.7122	0.7264	0.7107	0.2927	0.4138	0.2264	0.7207	0.5172	0.4348	0.6383	0.5907
	GPT-4o	0.5362	0.6804	0.5197	0.9851	0.7282	0.0684	0.3636	0.0377	0.7880	0.1748	1.0000	0.0957	0.3078
	Llama 3.1-8b	0.5771	0.5066	0.4514	0.5771	0.0189	0.0231	0.0299	0.0189	0.4043	0.4444	0.4935	0.4043	0.3247
Image	ViT	0.7805	0.7854	0.7703	0.8010	0.7307	0.4906	0.4906	0.4906	0.7706	0.4889	0.5116	0.4681	0.5883
	ViViT	0.5012	0.5745	0.5019	0.6716	0.6559	0.1786	0.2419	0.1415	0.6708	0.1951	0.2286	0.1702	0.3161
Audio	MFCC	0.5262	0.6769	0.5142	0.9900	0.7357	0.2563	0.3180	0.2146	0.7506	0.0741	0.2857	0.0426	0.2503
	Wav2Vec2	0.7781	0.7963	0.7357	0.8657	0.7357	0.3117	0.5000	0.2264	0.7930	0.6066	0.5470	0.6809	0.5716
Video	GPT-4	0.4938	0.6381	0.4972	0.8905	0.7082	0.0488	0.1765	0.0283	0.7556	0.1695	0.4167	0.1064	0.2855
	Llama-VL	0.4250	0.4800	0.4000	0.7800	0.2500	0.0250	0.1000	0.0150	0.3800	0.1500	0.3800	0.1000	0.2180
Multimodal	DeepCNN	0.7623	0.7512	0.7634	0.7398	0.6785	0.6345	0.6189	0.6512	0.6612	0.5803	0.5692	0.5917	0.6587
	CMHFM	0.7645	0.7534	0.7701	0.7405	0.6802	0.6372	0.6241	0.6509	0.6634	0.5814	0.5723	0.5922	0.6604
	CSID	0.7658	0.7556	0.7714	0.7437	0.6814	0.6394	0.6273	0.6534	0.6649	0.5834	0.5745	0.5938	0.6621
	MCMF	0.7661	0.7568	0.7735	0.7452	0.6819	0.6401	0.6289	0.6541	0.6652	0.5845	0.5751	0.5942	0.6625
	MuT	0.7667	0.7574	0.7741	0.7459	0.6823	0.6408	0.6296	0.6548	0.6658	0.5849	0.5756	0.5948	0.6627
	Proposed Method	0.8955	0.8448	0.8000	0.8955	0.6698	0.6605	0.6513	0.6698	0.4894	0.5702	0.6866	0.4894	0.6918

When compared to multimodal baselines, our approach also demonstrates clear superiority. The strongest competing multimodal method, MuT, records a macro-F1 of 66.27%; hence, our proposed model exhibits an absolute improvement of approximately 2.91%. Additionally, our model consistently outperforms other multimodal methods such as DeepCNN, CMHFM, CSID, and MCMF across individual categories such as non-hate, implicit hate, and

8.4 Results

explicit hate, highlighting its balanced performance. These results underscore the effectiveness of our two-stage contrastive learning framework in integrating complementary textual, visual, and audio cues, thereby establishing a new state-of-the-art for multiclass hate speech detection in videos.

8.4.5 Ablation Analysis

8.4.5.1 Impact of Different Modalities

We compared the performance of our model, considering all combinations of the three modalities on the ImpliHateVid dataset. The results of binary classification can be seen in Figure 8.3 and those of three-class classification have been highlighted in Table 8.4.

Table 8.4: Impact of Different Modalities on Multiclass Classification on ImpliHateVid Dataset

Modality	Non Hate Videos				Implicit Hate Videos				Explicit Hate Videos				Overall Macro-F1
	Acc	F1	Prec	Rec	Acc	F1	Prec	Rec	Acc	F1	Prec	Rec	
Text	0.8905	0.8424	0.7991	0.8905	0.2264	0.3038	0.4615	0.2264	0.7447	0.6393	0.5600	0.7447	0.5951
Image	0.8607	0.8317	0.8046	0.8607	0.2453	0.3077	0.4127	0.2453	0.7234	0.6267	0.5587	0.7234	0.5587
Audio	0.7811	0.7677	0.7548	0.7811	0.4151	0.4583	0.5116	0.4151	0.6064	0.5672	0.5327	0.6064	0.5977
Text + Audio	0.8657	0.8208	0.7803	0.8657	0.3113	0.3750	0.4714	0.3113	0.6915	0.6435	0.6019	0.6915	0.6131
Audio + Image	0.8706	0.8413	0.8140	0.8706	0.2642	0.3394	0.4746	0.2642	0.7872	0.6697	0.5827	0.7872	0.6168
Text + Image	0.6816	0.6903	0.6995	0.6816	0.1604	0.2716	0.8500	0.1604	0.5532	0.3728	0.2811	0.5532	0.4448
Proposed Method	0.8955	0.8448	0.8000	0.8955	0.6698	0.6605	0.6513	0.6698	0.4894	0.5702	0.6866	0.4894	0.6918

Examining the table for unimodal results, the text-only model achieves strong performance for non-hate videos with an F1 of 84.24% but struggles with implicit hate content with F1 of 30.38%. Similarly, the image-only and audio-only models show moderate performance, with audio achieving a relatively higher F1 of 45.83% on implicit hate videos compared to text and image modalities.

The text + audio combination boosts performance on implicit hate videos with an of F1 of 37.50% compared to using text alone. The audio + image configuration yields a notable improvement for explicit hate videos with an F1 of 66.97%, highlighting the benefit of integrating complementary visual and auditory cues. Conversely, the text + image combination, despite showing promise in precision, falls short in overall effectiveness, as indicated by a lower macro-F1 of 44.48%.

Our proposed method, which integrates textual, visual, and audio features simultaneously, outperforms all the ablated configurations. It achieves an overall accuracy of 89.55%,

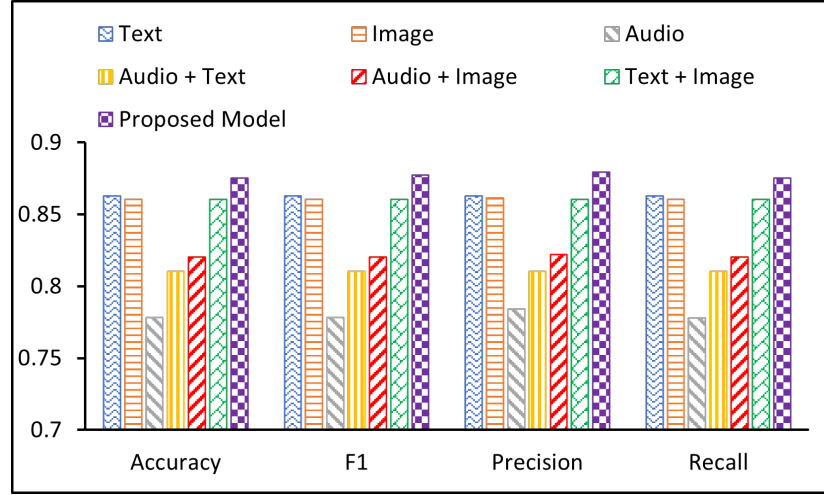


Figure 8.3: Impact of Different Modality Combinations on Binary Classification on Implied-HateVid Dataset

an F1-score of 84.48% for non-hate videos, and significantly higher performance for implicit hate with an F1 of 66.05% and explicit hate videos with an F1 of 57.02%. The overall macro-F1 of 69.18% clearly demonstrates the advantage of fully leveraging multimodal information. Figure 8.3 further illustrates this performance gain, where the proposed model outperforms all other configurations in binary classification. These results emphasize that combining all three modalities leads to a more robust and balanced classification outcome across all categories.

8.4.5.2 Impact of Emotion, Caption, and Sentiment Features

Table 8.5: Impact of Different Features on Multiclass Classification on Implied-HateVid Dataset

Features	Non Hate Videos				Implicit Hate Videos				Explicit Hate Videos				Overall Macro-F1
	Acc	F1	Prec	Rec	Acc	F1	Prec	Rec	Acc	F1	Prec	Rec	
w/o Emotions	0.8209	0.8312	0.8418	0.8209	0.6415	0.6507	0.6602	0.6415	0.5851	0.5609	0.5392	0.5851	0.6809
w/o Captions	0.8258	0.8217	0.8177	0.8258	0.6509	0.6448	0.6389	0.6509	0.5425	0.5542	0.5667	0.5425	0.6736
w/o Sentiments	0.8408	0.8325	0.8244	0.8408	0.6226	0.6438	0.6667	0.6226	0.5638	0.5548	0.5463	0.5638	0.6770
Proposed Method	0.8955	0.8448	0.8000	0.8955	0.6698	0.6605	0.6513	0.6698	0.4894	0.5702	0.6866	0.4894	0.6918

Table 8.5 summarizes the impact of omitting specific features on the three-class classification performance. For each ablation, removing emotions, captions, or sentiments, the table reports accuracy, F1-score, precision, and recall for each video category, along with the overall macro-F1 score and overall accuracy. Notably, the full proposed model, which uses all features, achieves a balanced performance with an overall accuracy of 71.32% and a macro-F1 of 66.29%, outperforming the ablated versions. The binary classification results

are shown in Figure 8.4.

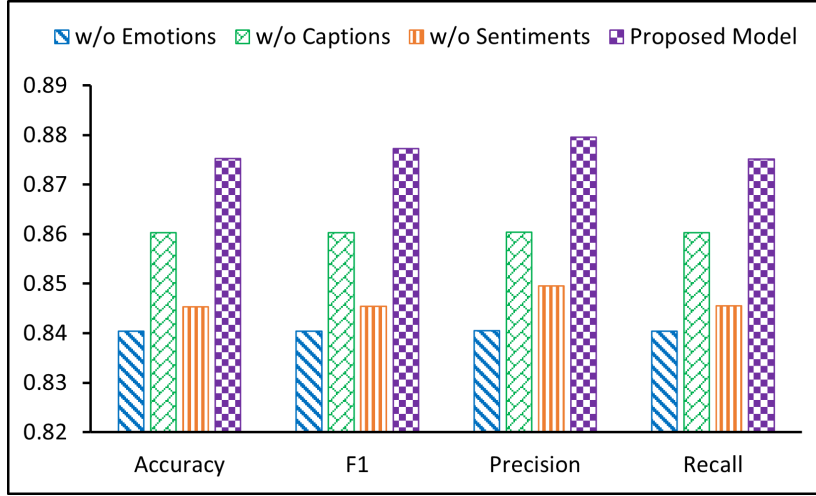


Figure 8.4: Impact of Different Features on Binary Classification on ImpliHateVid Dataset

8.5 Conclusion

This work makes two significant contributions. First, we introduce ImpliHateVid, a novel dataset specifically curated for implicit hate speech detection in videos. Comprising 2,009 videos, ImpliHateVid represents one of the first large-scale benchmarks for video-based implicit hate detection. This dataset not only fills a critical gap in the literature but also provides a valuable resource for advancing research in multimodal hate speech analysis. Second, we propose a two-stage contrastive learning framework that effectively integrates textual, visual, and audio modalities. In the first stage, dedicated encoders extract robust, modality-specific features. These features are then aligned into a unified embedding space via cross-modal encoders in the second stage, using a supervised contrastive loss that clusters similar samples and separates dissimilar ones. Additionally, our model is enhanced by incorporating sentiment, emotion, and image caption features, which capture subtle cues associated with hate speech. Extensive experiments on ImpliHateVid and the HateMM dataset demonstrate that our approach significantly outperforms all the baselines. These findings underscore the effectiveness of leveraging cross-modal information to capture the full context of hateful content in videos. Future work will focus on extending the framework to incorporate multilingualism and exploring real-time detection applications. Overall, our contributions pave

the way for more accurate systems in combating online hate in videos.

8.6 Limitations

Despite the promising results, our approach has several limitations. First, the performance of the multimodal framework depends heavily on the quality and alignment of data across modalities. Noisy or misaligned text, image, or audio inputs can adversely affect the joint embedding space and, consequently, the overall classification accuracy. Second, the reliance on pre-trained encoders means that any shortcomings in these models, such as domain mismatch or insufficient representation of hate speech nuances, can limit the effectiveness of our system.

Third, the supervised contrastive loss, while effective, is sensitive to hyperparameter settings such as the temperature parameter and the strategy for selecting positive and negative pairs. Improper tuning can lead to suboptimal clustering of similar samples and separation of dissimilar ones. Future research should focus on developing more robust and adaptive models and expanding the dataset to cover a broader range of hate speech phenomena.

Chapter 9

Conclusions and Future Research Directions

9.1 Summary of the Thesis

This thesis investigated the problem of hostile content detection in online social media, addressing a broad spectrum of harmful expressions including abusive language, hate speech, misogyny, and dark humor. Motivated by the rapid growth of user-generated content and the limitations of existing moderation systems, the research focused on developing robust, explainable, and multimodal frameworks capable of handling linguistic diversity, multimodal complexity, and implicit hostility. The thesis systematically explored hostile content across three major modalities: text, memes, and videos.

9.1.1 Multilingual Hostile Content Detection and Explanation

In the textual domain, the work addressed low-resource and multilingual settings, with particular emphasis on Indic languages and code-mixed data. By integrating textual representations with user-aware and social-context features, the proposed approaches demonstrated that behavioral and contextual signals significantly enhance abusive content detection in noisy and data-scarce environments.

To address the lack of transparency in hate speech detection systems, the thesis introduced explainability-guided learning through the X-MuTeST framework. By contributing

9.2 Contributions

multilingual benchmark datasets with token-level human rationales and combining n-gram-based explainability with large language model consultation, this work improved both classification performance and alignment between machine-generated explanations and human reasoning.

9.1.2 Multimodal Hostile Content Detection in Memes

Moving beyond text, the thesis examined multimodal hostile content in memes, where meaning often emerges from complex interactions between images and text. The proposed MM-ORIENT framework tackled noise propagation in multimodal fusion by reconstructing unimodal features through cross-modal relation graphs and hierarchical interactive attention. This approach enabled effective semantic comprehension across multiple tasks, including offensive content detection, sarcasm, and humor recognition. The thesis further explored task-specific hostility through multimodal misogyny detection, demonstrating that gender-targeted abuse requires specialized modeling beyond generic offensive content detectors. Additionally, the work addressed the underexplored problem of dark humor understanding by introducing the D-Humor dataset and a reasoning-augmented multimodal framework capable of identifying targets and intensity levels, moving beyond binary classification.

9.1.3 Multimodal Hostile Content Detection in Videos

Finally, the thesis extended hostile content detection to the video modality with a focus on implicit hate speech. Through the introduction of the ImpliHateVid dataset and a two-stage contrastive learning framework, the research showed that aligning audio, visual, and textual representations is crucial for capturing subtle and context-dependent hate expressions in dynamic content.

9.2 Contributions

The key contributions of this thesis can be summarized as follows:

1. **Low-resource and multilingual text analysis:** Proposed user-aware and multilingual

frameworks for abusive content detection that effectively handle code-mixed and low-resource Indic languages by integrating social-context and linguistic features.

2. **Explainable hate speech detection:** Introduced X-MuTeST, a novel explainability-guided framework supported by multilingual benchmark datasets with token-level human rationales, improving both interpretability and detection performance.
3. **Noise-resilient multimodal fusion:** Developed MM-ORIENT, a multimodal-multitask framework that mitigates noise propagation through cross-modal relation graphs and hierarchical monomodal attention for robust meme understanding.
4. **Task-specific multimodal hostility modeling:** Proposed a context-aware and graph-based framework for misogyny detection, highlighting the importance of specialized architectures for gender-targeted hostility.
5. **Dark humor understanding:** Contributed the first multimodal dark humor dataset with target and intensity annotations and introduced a reasoning-augmented framework using vision-language models for fine-grained analysis.
6. **Implicit hate speech detection in videos:** Introduced the ImpliHateVid benchmark dataset and a two-stage contrastive learning framework for capturing implicit hate in videos through multimodal alignment.

Collectively, these contributions advance the state of the art in hostile content detection by combining robustness, explainability, and multimodal reasoning across diverse and underexplored settings.

9.3 Limitations

Despite its contributions, this thesis has certain limitations. While the proposed models were evaluated on multiple benchmark datasets, real-world deployment scenarios may introduce additional challenges such as rapidly evolving language, adversarial user behavior, and platform-specific moderation policies. The reliance on large language models and vision-language models also introduces computational overhead, which may limit scalability in resource-constrained environments.

Furthermore, although explainability was addressed through human rationales and reasoning-based approaches, achieving universally interpretable explanations across different cultures and communities remains an open challenge. The subjective nature of hostility, particularly in cases such as dark humor and implicit hate, also continues to pose difficulties for both annotation and evaluation.

Another important limitation concerns the cultural and regional dependency of hostile content. Expressions of hostility, sarcasm, misogyny, dark humor, and other forms of implicit harmful content are often shaped by cultural norms, social contexts, and local linguistic practices. Consequently, models trained on data from a particular cultural setting may not generalize effectively to other regions or communities. For instance, the use of sarcasm, symbolism, or humor in Indian memes may differ substantially from that observed in Western meme culture. Addressing such cultural variations remains a challenging problem and highlights the need for broader cross-cultural evaluation and adaptation strategies.

9.4 Future Research Directions

The findings of this thesis open several promising directions for future research. First, extending the proposed frameworks to additional low-resource languages and cross-cultural contexts would further improve their generalizability. Future studies should also investigate culturally aware hostile content detection models that can better account for region-specific expressions of hostility, sarcasm, misogyny, and dark humor. Developing benchmark datasets spanning diverse cultural settings and evaluating cross-cultural transferability would help assess the robustness of existing approaches and reduce performance degradation when models are deployed beyond their original training domains.

Incorporating continual and lifelong learning strategies could enable models to adapt to evolving forms of online hostility and emerging slang or coded expressions. In the multi-modal domain, future work may explore finer-grained temporal modeling for videos, richer interactions between vision-language models and task-specific encoders, and the integration of external knowledge sources for improved contextual understanding. From an ethical perspective, developing evaluation protocols that explicitly measure fairness, bias, and societal impact remains an important research direction. Another direction could be to suggest inspir-

ing and hopeful content [240]. Not just filtering out hostile content, but also recommending content that motivates and inspires users.

Overall, this thesis lays a strong foundation for future work toward scalable, explainable, and multimodal hostile content detection systems, contributing to the broader goal of fostering safer and more inclusive online environments.

Bibliography

- [1] A. Sharma, A. Kabra, and M. Jain, “Ceasing hate with moh: Hate speech detection in hindi–english code-switched language,” *Information Processing & Management*, vol. 59, no. 1, p. 102760, 2022.
- [2] E. W. Pamungkas, V. Basile, and V. Patti, “Misogyny detection in twitter: a multilingual and cross-domain study,” *Information processing & management*, vol. 57, no. 6, p. 102360, 2020.
- [3] S. K. R. Kasu, M. Z. U. Rehman, S. S. Dar, R. B. Junghare, D. S. Namboodiri, and N. Kumar, “D-humor: Dark humor understanding via multimodal open-ended reasoning,” in *2025 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2025, pp. 377–386.
- [4] J. Zhang and Y. Wang, “Srcb at semeval-2022 task 5: Pretraining based image to text late sequential fusion system for multimodal misogynous meme identification,” in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 2022, pp. 585–596.
- [5] S. Tuarob, M. Satravisut, P. Sangtunchai, S. Nunthavanich, and T. Noraset, “Falcon: Detecting and classifying abusive language in social networks using context features and unlabeled data,” *Information Processing & Management*, vol. 60, no. 4, p. 103381, 2023.
- [6] C. Burger, “Humor styles, bullying victimization and psychological school adjustment: mediation, moderation and person-oriented analyses,” *International journal of environmental research and public health*, vol. 19, no. 18, p. 11415, 2022.
- [7] M. R. Awal, R. K.-W. Lee, E. Tanwar, T. Garg, and T. Chakraborty, “Model-agnostic meta-learning for multilingual hate speech detection,” *IEEE Transactions on Computational Social Systems*, 2023.
- [8] M. Z. U. Rehman, S. Mehta, K. Singh, K. Kaushik, and N. Kumar, “User-aware multilingual abusive content detection in social media,” *Information Processing & Management*, vol. 60, no. 5, p. 103450, 2023.
- [9] H. Madhu, S. Satapara, S. Modha, T. Mandl, and P. Majumder, “Detecting offensive speech in conversational code-mixed dialogue on social media: A contextual dataset and benchmark experiments,” *Expert Systems with Applications*, vol. 215, p. 119342, 2023.

- [10] B. Mathew, P. Saha, S. M. Yimam, C. Biemann, P. Goyal, and A. Mukherjee, “Hat-explain: A benchmark dataset for explainable hate speech detection,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 17, 2021, pp. 14 867–14 875.
- [11] A. Joshi, P. Bhattacharyya, and M. J. Carman, “Automatic sarcasm detection: A survey,” *ACM Comput. Surv.*, vol. 50, no. 5, p. 1–22, 2017.
- [12] S. Amiriparian, L. Christ, A. König, E.-M. Meßner, A. Cowen, E. Cambria, and B. W. Schuller, “Muse 2022 challenge: Multimodal humour, emotional reactions, and stress,” in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, p. 7389–7391.
- [13] A. Chhabra and D. K. Vishwakarma, “Multimodal hate speech detection via multi-scale visual kernels and knowledge distillation architecture,” *Engineering Applications of Artificial Intelligence*, vol. 126, p. 106991, 2023.
- [14] X. Sun, F. Yao, and C. Ding, “Modeling high-order relationships: Brain-inspired hypergraph-induced multimodal-multitask framework for semantic comprehension,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2023.
- [15] P. Xu, X. Zhu, and D. A. Clifton, “Multimodal learning with transformers: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 10, pp. 12 113–12 132, 2023.
- [16] M. Z. U. Rehman, D. Raghuvanshi, U. Jain, S. Bansal, and N. Kumar, “A multimodal-multitask framework with cross-modal relation and hierarchical interactive attention for semantic comprehension,” *Information Fusion*, p. 103628, 2025.
- [17] S. Mai, Y. Zeng, and H. Hu, “Multimodal information bottleneck: Learning minimal sufficient unimodal and multimodal representations,” *IEEE Transactions on Multimedia*, vol. 25, pp. 4121–4134, 2022.
- [18] M. ElSherief, C. Ziems, D. Muchlinski, V. Anupindi, J. Seybolt, M. De Choudhury, and D. Yang, “Latent hatred: A benchmark for understanding implicit hate speech,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 345–363.
- [19] T. Guo, L. Lin, H. Liu, C. Zheng, Z. Tu, and H. Wang, “Implicit offensive speech detection based on multi-feature fusion,” in *International Conference on Knowledge Science, Engineering and Management*. Springer, 2023, pp. 27–38.
- [20] M. Z. U. Rehman, S. K. R. Kasu, S. R. Koppula, S. R. R. Chirra, S. S. Singh, and N. Kumar, “X-mutest: A multilingual benchmark for explainable hate speech detection and a novel llm-consulted explanation framework,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, pp. 39 134–39 142, Mar. 2026.
- [21] M. Z. U. Rehman, S. Zahoor, A. Manzoor, M. Maqbool, and N. Kumar, “A context-aware attention and graph neural network-based multimodal framework for misogyny detection,” *Information Processing & Management*, vol. 62, no. 1, p. 103895, 2025.

-
- [22] M. Z. U. Rehman, A. Bhatnagar, O. Kabde, S. Bansal, and N. Kumar, “Implihatevid: A benchmark dataset and two-stage contrastive learning framework for implicit hate speech detection in videos,” in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025, pp. 17 209–17 221.
 - [23] M. H. Akram, K. Shahzad, and M. Bashir, “Ise-hate: A benchmark corpus for inter-faith, sectarian, and ethnic hatred detection on social media in urdu,” *Information Processing & Management*, vol. 60, no. 3, p. 103270, 2023.
 - [24] M. Marreddy, S. R. Oota, L. S. Vakada, V. C. Chinni, and R. Mamidi, “Am i a resource-poor language? data sets, embeddings, models and analysis for four different nlp tasks in telugu language,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 1, pp. 1–34, 2022.
 - [25] S. FHA, U. Sharma, and H. Naleer, “Development of an efficient method to detect mixed social media data with tamil-english code using machine learning techniques,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 2, pp. 1–19, 2023.
 - [26] M. Z. U. Rehman, A. Shah, and N. Kumar, “Detecting implicit sexism in digital social networks via contrastive learning-based adaptive network,” *Computers and Electrical Engineering*, vol. 137, p. 111261, 2026.
 - [27] S. Satapara, P. Majumder, T. Mandl, S. Modha, H. Madhu, T. Ranasinghe, M. Zampieri, K. North, and D. Premasiri, “Overview of the hasoc subtrack at fire 2022: Hate speech and offensive content identification in english and indo-aryan languages,” in *Proceedings of the 14th Annual Meeting of the Forum for Information Retrieval Evaluation*, 2022, pp. 4–7.
 - [28] S. Ghosal, A. Jain, D. K. Tayal, V. G. Menon, and A. Kumar, “Inculcating context for emoji powered bengali hate speech detection using extended fuzzy svm and text embedding models,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2023.
 - [29] A. Velankar, H. Patil, and R. Joshi, “Mono vs multilingual bert for hate speech detection and text classification: A case study in marathi,” in *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*. Springer, 2023, pp. 121–128.
 - [30] G. K. Kumar, A. Gehlot, S. S. Mullappilly, and K. Nandakumar, “Mucot: Multilingual contrastive training for question-answering in low-resource languages,” in *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, 2022, pp. 15–24.
 - [31] E. Krupalija, D. Đonko, and H. Šupić, “Usage of user hate speech index for improving hate speech detection in twitter posts,” in *2022 XXVIII International Conference on Information, Communication and Automation Technologies (ICAT)*. IEEE, 2022, pp. 1–6.

- [32] S. Chopra, R. Sawhney, P. Mathur, and R. R. Shah, “Hindi-english hate speech detection: Author profiling, debiasing, and practical perspectives,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 01, 2020, pp. 386–393.
- [33] G. K. Pitsilis, H. Ramampiaro, and H. Langseth, “Effective hate-speech detection in twitter data using recurrent neural networks,” *Applied Intelligence*, vol. 48, no. 12, pp. 4730–4742, 2018.
- [34] N. Kumar, A. Yadandla, K. Suryamukhi, N. Ranabothu, S. Boya, and M. Singh, “Arousal prediction of news articles in social media,” in *Mining Intelligence and Knowledge Exploration: 5th International Conference, MIKE 2017, Hyderabad, India, December 13–15, 2017, Proceedings 5*. Springer, 2017, pp. 308–319.
- [35] J. F. Sánchez-Rada and C. A. Iglesias, “Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison,” *Information Fusion*, vol. 52, pp. 344–356, 2019.
- [36] E. Mosca, M. Wich, and G. Groh, “Understanding and interpreting the impact of user context in hate speech detection,” in *Proceedings of the Ninth International Workshop on Natural Language Processing for Social Media*, 2021, pp. 91–102.
- [37] E. F. Unsvåg and B. Gambäck, “The effects of user features on twitter hate speech detection,” in *Proceedings of the 2nd workshop on abusive language online (ALW2)*, 2018, pp. 75–85.
- [38] O. Zaidan, J. Eisner, and C. Piatko, “Using “annotator rationales” to improve machine learning for text categorization,” in *Human language technologies 2007: The conference of the North American chapter of the association for computational linguistics; proceedings of the main conference*, 2007, pp. 260–267.
- [39] A. Yessenalina, Y. Choi, and C. Cardie, “Automatically generating annotator rationales to improve sentiment classification,” in *Proceedings of the ACL 2010 Conference Short Papers*, 2010, pp. 336–341.
- [40] B. Wang, J. Ma, H. Lin, Z. Yang, R. Yang, Y. Tian, and Y. Chang, “Explainable fake news detection with large language model via defense among competing wisdom,” in *Proceedings of the ACM on Web Conference 2024*, 2024, pp. 2452–2463.
- [41] H. Lin, Z. Luo, W. Gao, J. Ma, B. Wang, and R. Yang, “Towards explainable harmful meme detection through multimodal debate between large language models,” in *Proceedings of the ACM on Web Conference 2024*, 2024, pp. 2359–2370.
- [42] S. Roy, A. Harshvardhan, A. Mukherjee, and P. Saha, “Probing llms for hate speech detection: strengths and vulnerabilities,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 6116–6128.
- [43] C. Clarke, M. Hall, G. Mittal, Y. Yu, S. Sajeed, J. Mars, and M. Chen, “Rule by example: Harnessing logical rules for explainable hate speech detection,” in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023, pp. 364–376.

-
- [44] M. U. Arshad and W. Shahzad, “Understanding hate speech: the hateinsights dataset and model interpretability,” *PeerJ Computer Science*, vol. 10, p. e2372, 2024.
 - [45] P. Kapil, G. Kumari, A. Ekbal, S. Pal, A. Chatterjee, and B. Vinutha, “Hhld: Hateful posts identification in hindi language leveraging multi task learning,” *IEEE Access*, 2023.
 - [46] M. Sawant, A. Younus, S. Caton, and M. A. Qureshi, “Using explainable ai (xai) for identification of subjectivity in hate speech annotations for low-resource languages,” in *Proceedings of the 4th International Workshop on Open Challenges in Online Social Networks*, 2024, pp. 10–17.
 - [47] S. Yadav, A. Kaushik, and K. McDaid, “Hate speech is not free speech: Explainable machine learning for hate speech detection in code-mixed languages,” in *2023 IEEE International Symposium on Technology and Society (ISTAS)*. IEEE, 2023, pp. 1–8.
 - [48] R. R. Geleta, K. Eckelt, E. Parada-Cabaleiro, and M. Schedl, “Exploring intensities of hate speech on social media: A case study on explaining multilingual models with xai,” in *Proceedings of the 4th Conference on Language, Data and Knowledge*, 2023, pp. 532–537.
 - [49] A. Muti, K. Korre, and A. Barrón-Cedeño, “Unibo at semeval-2022 task 5: A multi-modal bi-transformer approach to the binary and fine-grained identification of misogyny in memes,” in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 2022, pp. 663–672.
 - [50] Y. Gu, I. Castro, and G. Tyson, “Mmvae at semeval-2022 task 5: A multi-modal multi-task vae on misogynous meme detection,” in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 2022, pp. 700–710.
 - [51] R. Gomez, J. Gibert, L. Gomez, and D. Karatzas, “Exploring hate speech detection in multimodal publications,” in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2020, pp. 1470–1478.
 - [52] J. Yuan, Y. Yu, G. Mittal, M. Hall, S. Sajeew, and M. Chen, “Rethinking multimodal content moderation from an asymmetric angle with mixed-modality,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 8532–8542.
 - [53] E. Mahajan, H. Mahajan, and S. Kumar, “Ensmulhatecyb: Multilingual hate speech and cyberbully detection in online social media,” *Expert Systems with Applications*, vol. 236, p. 121228, 2024.
 - [54] A. Bhandari, S. B. Shah, S. Thapa, U. Naseem, and M. Nasim, “Crisishatemmm: Multi-modal analysis of directed and undirected hate speech in text-embedded images from russia-ukraine conflict,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1993–2002.
 - [55] A. Anshul, G. S. Pranav, M. Z. U. Rehman, and N. Kumar, “A multimodal framework for depression detection during covid-19 via harvesting social media,” *IEEE Transactions on Computational Social Systems*, vol. 11, no. 2, pp. 2872–2888, 2024.

- [56] S. Bansal, K. Gowda, and N. Kumar, “A hybrid deep neural network for multimodal personalized hashtag recommendation,” *IEEE transactions on computational social systems*, vol. 10, no. 5, pp. 2439–2459, 2022.
- [57] K. Guo, R. Ma, S. Luo, and Y. Wang, “Coco at semeval-2023 task 10: Explainable detection of online sexism,” in *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, 2023, pp. 469–476.
- [58] G. Attanasio, D. Nozza, E. Pastor, D. Hovy *et al.*, “Benchmarking post-hoc interpretability approaches for transformer-based misogyny detection,” in *Proceedings of NLP Power! The First Workshop on Efficient Benchmarking in NLP*, 2022.
- [59] R. Calderón-Suarez, R. M. Ortega-Mendoza, M. Montes-Y-Gómez, C. Toxqui-Quitl, and M. A. Márquez-Vera, “Enhancing the detection of misogynistic content in social media by transferring knowledge from song phrases,” *IEEE Access*, vol. 11, pp. 13 179–13 190, 2023.
- [60] A. Muti, F. Fericola, and A. Barrón-Cedeno, “Misogyny and aggressiveness tend to come together and together we address them,” in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 2022, pp. 4142–4148.
- [61] A. Muti and A. Barrón-Cedeño, “A checkpoint on multilingual misogyny identification,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, 2022, pp. 454–460.
- [62] N. Dehingia, J. McAuley, L. McDougal, E. Reed, J. G. Silverman, L. Urada, and A. Raj, “Violence against women on twitter in india: Testing a taxonomy for online misogyny and measuring its prevalence during covid-19,” *PLoS one*, vol. 18, no. 10, p. e0292121, 2023.
- [63] L. Han and H. Tang, “Designing of prompts for hate speech recognition with in-context learning,” in *2022 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE, 2022, pp. 319–320.
- [64] C. Luo, R. Bhambhoria, S. Dahan, and X. Zhu, “Legally enforceable hate speech detection for public forums,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 10 948–10 963.
- [65] C. Christodoulou, “Nlpdame at climateactivism 2024: Mistral sequence classification with peft for hate speech, targets and stance event detection,” in *Proceedings of the 7th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2024)*, 2024, pp. 96–104.
- [66] E. Hossain, O. Sharif, M. M. Hoque, and S. M. Preum, “Align before attend: Aligning visual and textual features for multimodal hateful content detection,” in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, 2024, pp. 162–174.
- [67] A. Yadav and V. Singh, “Hatefusion: Harnessing attention-based techniques for enhanced filtering and detection of implicit hate speech,” *IEEE Transactions on Computational Social Systems*, 2024.

-
- [68] F. Yao, X. Sun, H. Yu, W. Zhang, W. Liang, and K. Fu, “Mimicking the brain’s cognition of sarcasm from multidisciplinary for twitter sarcasm detection,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 1, pp. 228–242, 2023.
 - [69] Y. Zhang, J. Wang, Y. Liu, L. Rong, Q. Zheng, D. Song, P. Tiwari, and J. Qin, “A multitask learning model for multimodal sarcasm, sentiment and emotion recognition in conversations,” *Information Fusion*, vol. 93, pp. 282–301, 2023.
 - [70] S. Ghosh, G. V. Singh, A. Ekbal, and P. Bhattacharyya, “COMMA-DEER: COMmon-sense aware multimodal multitask approach for detection of emotion and emotional reasoning in conversations,” in *Proceedings of the 29th International Conference on Computational Linguistics*, oct 2022, pp. 6978–6990.
 - [71] F. Lv, X. Chen, Y. Huang, L. Duan, and G. Lin, “Progressive modality reinforcement for human multimodal emotion recognition from unaligned multimodal sequences,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 2554–2562.
 - [72] A. Bhandari, S. B. Shah, S. Thapa, U. Naseem, and M. Nasim, “Crisishatemmm: Multimodal analysis of directed and undirected hate speech in text-embedded images from russia-ukraine conflict,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2023, pp. 1993–2002.
 - [73] Z. Zhang, K. Chen, R. Wang, M. Utiyama, E. Sumita, Z. Li, and H. Zhao, “Universal multimodal representation for language understanding,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 7, pp. 9169–9185, 2023.
 - [74] W. Ruch, S. Heintz, T. Platt, L. Wagner, and R. T. Proyer, “Broadening humor: Comic styles differentially tap into temperament, character, and ability,” *Frontiers in psychology*, vol. 9, p. 6, 2018.
 - [75] Y. Bao, X. Zhao, P. Zhang, Y. Qi, and H. Li, “Hian: A hybrid interactive attention network for multimodal sarcasm detection,” *Pattern Recognition*, vol. 164, p. 111535, 2025.
 - [76] A. Gupta, A. Mittal, and R. Jain, “A novel sarcasm detection approach for text-image data: Leveraging multimodal fusion and weighted latent factors,” *Information Fusion*, p. 103266, 2025.
 - [77] M. Jia, C. Xie, and L. Jing, “Debiasing multimodal sarcasm detection with contrastive learning,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 16, 2024, pp. 18 354–18 362.
 - [78] B. Liang, L. Gui, Y. He, E. Cambria, and R. Xu, “Fusion and discrimination: A multimodal graph contrastive learning framework for multimodal sarcasm detection,” *IEEE Transactions on Affective Computing*, 2024.
 - [79] A. Saroj and S. Pal, “Ensemble-based domain adaptation on social media posts for irony detection,” *Multimedia Tools and Applications*, vol. 83, no. 8, pp. 23 249–23 268, 2024.

- [80] A. Maladry, E. Lefever, C. Van Hee, and V. Hoste, “The limitations of irony detection in dutch social media,” *Language Resources and Evaluation*, vol. 58, no. 4, pp. 1355–1386, 2024.
- [81] N. Bölücü and B. Can, “Semantically-informed graph neural networks for irony detection in turkish,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 24, no. 1, pp. 1–20, 2025.
- [82] D. Goyal, S. K. Mishra, and V. K. Rai, “A multimodal framework for satire vs. sarcasm detection,” in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. IEEE, 2024, pp. 1–7.
- [83] R. Ortega-Bueno, P. Rosso, and J. E. M. Pagola, “Multi-view informed attention-based model for irony and satire detection in spanish variants,” *Knowledge-Based Systems*, vol. 235, p. 107597, 2022.
- [84] A. Kamal, M. Abulaish *et al.*, “Contextualized satire detection in short texts using deep learning techniques,” *Journal of Web Engineering*, vol. 23, no. 1, pp. 27–52, 2024.
- [85] O. Ogbanufe, M. C. Jones, and J. I. Hancock, “Job demands, identity, and outcomes: The mediating role of cynicism among cybersecurity-focused employees,” *Computers & Security*, vol. 150, p. 104277, 2025.
- [86] Q. Liu, L. Wang, and M. Luo, “When seeing is not believing: self-efficacy and cynicism in the era of intelligent media,” *Humanities and Social Sciences Communications*, vol. 12, no. 1, pp. 1–13, 2025.
- [87] M. Das, R. Raj, P. Saha, B. Mathew, M. Gupta, and A. Mukherjee, “Hatemm: A multi-modal dataset for hate video classification,” in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 17, 2023, pp. 1014–1023.
- [88] H. Wang, T. R. Yang, U. Naseem, and R. K.-W. Lee, “Multihateclip: A multilingual benchmark dataset for hateful video detection on youtube and bilibili,” in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 7493–7502.
- [89] C. Alcântara, V. Moreira, and D. Feijo, “Offensive video detection: dataset and baseline results,” in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 2020, pp. 4309–4319.
- [90] C. S. Wu and U. Bhandary, “Detection of hate speech in videos using machine learning,” in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE, 2020, pp. 585–590.
- [91] Y. Kim, S. Park, and Y.-S. Han, “Generalizable implicit hate speech detection using contrastive learning,” in *Proceedings of the 29th International Conference on Computational Linguistics*, 2022, pp. 6667–6679.
- [92] N. B. Ocampo, E. Cabrio, and S. Villata, “Playing the part of the sharp bully: Generating adversarial examples for implicit hate speech detection,” in *Findings of the Association for Computational Linguistics: ACL 2023*, 2023, pp. 2758–2772.

-
- [93] T. Hartvigsen, S. Gabriel, H. Palangi, M. Sap, D. Ray, and E. Kamar, “Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 3309–3326.
 - [94] A. Nag, B. Samanta, A. Mukherjee, N. Ganguly, and S. Chakrabarti, “Transfer learning for low-resource multilingual relation classification,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 2, pp. 1–24, 2023.
 - [95] S. B. Das, A. Biradar, T. K. Mishra, and B. K. Patra, “Improving multilingual neural machine translation system for indic languages,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2022.
 - [96] M. Zampieri, T. Ranasinghe, M. Chaudhari, S. Gaikwad, P. Krishna, M. Nene, and S. Paygude, “Predicting the type and target of offensive social media posts in marathi,” *Social Network Analysis and Mining*, vol. 12, no. 1, p. 77, 2022.
 - [97] P. Mathur, R. Sawhney, M. Ayyar, and R. Shah, “Did you offend me? classification of offensive tweets in hinglish language,” in *Proceedings of the 2nd workshop on abusive language online (ALW2)*, 2018, pp. 138–148.
 - [98] V. Bansal, M. Tyagi, R. Sharma, V. Gupta, and Q. Xin, “A transformer based approach for abuse detection in code mixed indic languages,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2022.
 - [99] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, B. Gopalan, D. K. Margam, P. Aggarwal, R. T. Nagipogu, S. Dave *et al.*, “Muril: Multilingual representations for indian languages,” *arXiv preprint arXiv:2103.10730*, 2021.
 - [100] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “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, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.
 - [101] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, É. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, “Unsupervised cross-lingual representation learning at scale,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8440–8451.
 - [102] Datathon-IndoML-2021, “Multilingual abusive comment detection,” 2021. [Online]. Available: <https://kaggle.com/competitions/multilingualabusivecomment>
 - [103] R. S. harsha, “Iit-d multilingual abusive comment identification,” 2021. [Online]. Available: <https://kaggle.com/competitions/iiitd-abuse-detection-challenge>
 - [104] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

- [105] P. Lahoti, N. Mittal, and G. Singh, “A survey on nlp resources, tools, and techniques for marathi language processing,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 2, pp. 1–34, 2022.
- [106] B. Roark, L. Wolf-Sonkin, C. Kirov, S. J. Mielke, C. Johny, I. Demirsahin, and K. Hall, “Processing south asian languages written in the latin script: the dakshina dataset,” in *Proceedings of the 12th Language Resources and Evaluation Conference*, 2020, pp. 2413–2423.
- [107] E. E. Manganari, “Emoji use in computer-mediated communication,” *The International Technology Management Review*, vol. 10, no. 1, pp. 1–11, 2021.
- [108] J. Y. Kim, J. Sim, and D. Cho, “Identity and status: When counterspeech increases hate speech reporting and why,” *Information Systems Frontiers*, pp. 1–12, 2022.
- [109] M. Ribeiro, P. Calais, Y. Santos, V. Almeida, and W. Meira Jr, “Characterizing and detecting hateful users on twitter,” in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 12, no. 1, 2018.
- [110] L. Frischlich, T. Schatto-Eckrodt, S. Boberg, and F. Wintterlin, “Roots of incivility: How personality, media use, and online experiences shape uncivil participation,” *Media and Communication*, vol. 9, no. 1, pp. 195–208, 2021.
- [111] D. Kornbrot, “Point biserial correlation,” *Wiley StatsRef: Statistics Reference Online*, 2014.
- [112] C. Bousquet, “Words that harm: Defending the dignity approach to hate speech regulation,” *Canadian Journal of Law & Jurisprudence*, vol. 35, no. 1, pp. 31–57, 2022.
- [113] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, “Activation functions in deep learning: A comprehensive survey and benchmark,” *Neurocomputing*, 2022.
- [114] S. Biradar and S. Saumya, “Iitdwd@ tamilnlp-acl2022: Transformer-based approach to classify abusive content in dravidian code-mixed text,” in *Proceedings of the second workshop on speech and language technologies for Dravidian languages*, 2022, pp. 100–104.
- [115] F. S. Abousaleh, W.-H. Cheng, N.-H. Yu, and Y. Tsao, “Multimodal deep learning framework for image popularity prediction on social media,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 13, no. 3, pp. 679–692, 2020.
- [116] S. Sazed, “Abusive content detection in transliterated bengali-english social media corpus,” in *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, 2021, pp. 125–130.
- [117] S. Modha, P. Majumder, T. Mandl, and C. Mandalia, “Detecting and visualizing hate speech in social media: A cyber watchdog for surveillance,” *Expert Systems with Applications*, vol. 161, p. 113725, 2020.

-
- [118] E. Greevy and A. F. Smeaton, “Classifying racist texts using a support vector machine,” in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, 2004, pp. 468–469.
- [119] T. Davidson, D. Warmusley, M. Macy, and I. Weber, “Automated hate speech detection and the problem of offensive language,” in *Proceedings of the international AAAI conference on web and social media*, vol. 11, no. 1, 2017, pp. 512–515.
- [120] A. K. Yadav, M. Kumar, A. Kumar, D. Yadav *et al.*, “Hate speech recognition in multilingual text: hinglish documents,” *International Journal of Information Technology*, vol. 15, no. 3, pp. 1319–1331, 2023.
- [121] Z. Waseem and D. Hovy, “Hateful symbols or hateful people? predictive features for hate speech detection on twitter,” in *Proceedings of the NAACL student research workshop*, 2016, pp. 88–93.
- [122] N. F. Rajani, B. McCann, C. Xiong, and R. Socher, “Explain yourself! leveraging language models for commonsense reasoning,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 4932–4942.
- [123] O.-M. Camburu, T. Rocktäschel, T. Lukasiewicz, and P. Blunsom, “e-snli: Natural language inference with natural language explanations,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [124] J. DeYoung, S. Jain, N. F. Rajani, E. Lehman, C. Xiong, R. Socher, and B. C. Wallace, “Eraser: A benchmark to evaluate rationalized nlp models,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 4443–4458.
- [125] O. Zaidan and J. Eisner, “Modeling annotators: A generative approach to learning from annotator rationales,” in *Proceedings of the 2008 conference on Empirical methods in natural language processing*, 2008, pp. 31–40.
- [126] D. Khashabi, S. Chaturvedi, M. Roth, S. Upadhyay, and D. Roth, “Looking beyond the surface: A challenge set for reading comprehension over multiple sentences,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 252–262.
- [127] E. Lehman, J. DeYoung, R. Barzilay, and B. C. Wallace, “Inferring which medical treatments work from reports of clinical trials,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 3705–3717.
- [128] C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova, “Boolq: Exploring the surprising difficulty of natural yes/no questions,” in *Proceedings of NAACL-HLT*, 2019, pp. 2924–2936.

- [129] T. Mandl, S. Modha, A. Kumar M, and B. R. Chakravarthi, “Overview of the hasoc track at fire 2020: Hate speech and offensive language identification in tamil, malayalam, hindi, english and german,” in *Proceedings of the 12th annual meeting of the forum for information retrieval evaluation*, 2020, pp. 29–32.
- [130] T. Mandl, S. Modha, G. K. Shahi, A. K. Jaiswal, D. Nandini, D. Patel, P. Majumder, and J. Schäfer, “Overview of the hasoc track at fire 2020: Hate speech and offensive content identification in indo-european languages,” 2020.
- [131] S. Modha, T. Mandl, G. K. Shahi, H. Madhu, S. Satapara, T. Ranasinghe, and M. Zampieri, “Overview of the hasoc subtrack at fire 2021: Hate speech and offensive content identification in english and indo-aryan languages and conversational hate speech,” in *Proceedings of the 13th Annual Meeting of the Forum for Information Retrieval Evaluation*, 2021, pp. 1–3.
- [132] R. Priyadharshini, B. R. Chakravarthi, S. Malliga, C. Subalalitha, S. Kogilavani, B. Premjith, A. Murugappan, and P. K. Kumaresan, “Overview of shared-task on abusive comment detection in tamil and telugu,” in *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, 2023, pp. 80–87.
- [133] B. Premjith, B. R. Chakravarthi, P. K. Kumaresan, S. Rajiakodi, S. Karnati, S. Mangamuru, and C. Janakiram, “Findings of the shared task on hate and offensive language detection in telugu codemixed text (hold-telugu)@ dravidianlangtech 2024,” in *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, 2024, pp. 49–55.
- [134] B. R. Chakravarthi, R. Priyadharshini, S. Thavareesan, E. Sherly, R. Nadarajan, M. Ravikiran *et al.*, “Proceedings of the fourth workshop on speech, vision, and language technologies for dravidian languages,” in *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, 2024.
- [135] Y. Madhani, S. Parthan, P. Bedekar, G. Nc, R. Khapra, A. Kunchukuttan, P. Kumar, and M. M. Khapra, “Aksharantar: Open indic-language transliteration datasets and models for the next billion users,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 40–57.
- [136] S. Schmer-Galunder, R. Wheelock, Z. Jalan, A. Chvasta, S. Friedman, and E. Saltz, “Annotator in the loop: A case study of in-depth rater engagement to create a prosocial benchmark dataset,” in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, vol. 7, 2024, pp. 1319–1328.
- [137] U. Hahn, E. Beisswanger, E. Buyko, E. Faessler, J. Traumüller, S. Schröder, and K. Hornbostel, “Iterative refinement and quality checking of annotation guidelines—how to deal effectively with semantically sloppy named entity types, such as pathological phenomena,” in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, 2012, pp. 3881–3885.
- [138] A. Jacovi and Y. Goldberg, “Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness?” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 4198–4205.

-
- [139] OpenAI, “ChatGPT 4o,” url : <https://platform.openai.com/docs/models/gpt-4o>, 2024.
- [140] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan *et al.*, “The llama 3 herd of models,” *arXiv preprint arXiv:2407.21783*, 2024.
- [141] MistralAI, “Large Enough — mistral.ai,” <https://mistral.ai/news/mistral-large-2407/>, 2024.
- [142] T. H. Nguyen and K. Rudra, “Towards an interpretable approach to classify and summarize crisis events from microblogs,” in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 3641–3650.
- [143] I. Salles, F. Vargas, and F. Benevenuto, “Hatebrxplain: A benchmark dataset with human-annotated rationales for explainable hate speech detection in brazilian portuguese,” in *Proceedings of the 31st International Conference on Computational Linguistics*, 2025, pp. 6659–6669.
- [144] S. Cohen, D. Presil, O. Katz, O. Arbili, S. Messica, and L. Rokach, “Enhancing social network hate detection using back translation and gpt-3 augmentations during training and test-time,” *Information Fusion*, p. 101887, 2023.
- [145] A. A. Firmino, C. de Souza Baptista, and A. C. de Paiva, “Improving hate speech detection using cross-lingual learning,” *Expert Systems with Applications*, vol. 235, p. 121115, 2024.
- [146] S. Ghosh, A. Ekbal, P. Bhattacharyya, T. Saha, A. Kumar, and S. Srivastava, “Sehc: A benchmark setup to identify online hate speech in english,” *IEEE Transactions on Computational Social Systems*, vol. 10, no. 2, pp. 760–770, 2022.
- [147] P. Parikh, H. Abburi, N. Chhaya, M. Gupta, and V. Varma, “Categorizing sexism and misogyny through neural approaches,” *ACM Transactions on the Web (TWEB)*, vol. 15, no. 4, pp. 1–31, 2021.
- [148] C. D. Putra and H.-C. Wang, “Semi-meta-supervised hate speech detection,” *Knowledge-Based Systems*, p. 111386, 2024.
- [149] D. Chaffey, “Global social media statistics research summary 2023 [June 2023] — smartinsights.com,” <https://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/>, 2024, [Accessed 21-01-2024].
- [150] S. J. Dixon, “Social platforms: active user gender distribution 2023 | Statista — statista.com,” <https://www.statista.com/statistics/274828/>, 2024, [Accessed 21-01-2024].
- [151] M. Anzovino, E. Fersini, and P. Rosso, “Automatic identification and classification of misogynistic language on twitter,” in *Natural Language Processing and Information Systems: 23rd International Conference on Applications of Natural Language to Information Systems, NLDB 2018, Paris, France, June 13-15, 2018, Proceedings 23*. Springer, 2018, pp. 57–64.

- [152] E. W. Pamungkas, A. T. Cignarella, V. Basile, V. Patti *et al.*, “Automatic identification of misogyny in english and italian tweets at evalita 2018 with a multilingual hate lexicon,” in *CEUR Workshop Proceedings*, vol. 2263. CEUR-WS, 2018, pp. 1–6.
- [153] A. Bakarov, “Vector space models for automatic misogyny identification,” in *CEUR Workshop Proceedings*, vol. 2263. CEUR-WS, 2018, pp. 211–213.
- [154] J. A. García-Díaz, M. Cánovas-García, R. Colomo-Palacios, and R. Valencia-García, “Detecting misogyny in spanish tweets. an approach based on linguistics features and word embeddings,” *Future Generation Computer Systems*, vol. 114, pp. 506–518, 2021.
- [155] N. S. Samghabadi, P. Patwa, S. Pykl, P. Mukherjee, A. Das, and T. Solorio, “Aggression and misogyny detection using bert: A multi-task approach,” in *Proceedings of the second workshop on trolling, aggression and cyberbullying*, 2020, pp. 126–131.
- [156] G. Rizzi, F. Gasparini, A. Saibene, P. Rosso, and E. Fersini, “Recognizing misogynous memes: Biased models and tricky archetypes,” *Information Processing & Management*, vol. 60, no. 5, p. 103474, 2023.
- [157] E. Fersini, F. Gasparini, G. Rizzi, A. Saibene, B. Chulvi, P. Rosso, A. Lees, and J. Sorensen, “Semeval-2022 task 5: Multimedia automatic misogyny identification,” in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 2022, pp. 533–549.
- [158] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, “Free-form image inpainting with gated convolution,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 4471–4480.
- [159] M. Z. U. Rehman, D. Raghuvanshi, and N. Kumar, “Kisanqrs: A deep learning-based automated query-response system for agricultural decision-making,” *Computers and Electronics in Agriculture*, vol. 213, p. 108180, 2023, doi: <https://doi.org/10.1016/j.compag.2023.108180>.
- [160] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, “Learning transferable visual models from natural language supervision,” in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.
- [161] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, “Detectron2,” 2019.
- [162] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
- [163] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

-
- [164] F.-M. Plaza-Del-Arco, M. D. Molina-González, L. A. Ureña-López, and M. T. Martín-Valdivia, “Detecting misogyny and xenophobia in spanish tweets using language technologies,” *ACM Transactions on Internet Technology (TOIT)*, vol. 20, no. 2, pp. 1–19, 2020.
- [165] E. Bassignana, V. Basile, V. Patti *et al.*, “Hurtlex: A multilingual lexicon of words to hurt,” in *CEUR Workshop proceedings*, vol. 2253. CEUR-WS, 2018, pp. 1–6.
- [166] J. Li, D. Li, C. Xiong, and S. Hoi, “Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation,” in *International Conference on Machine Learning*. PMLR, 2022, pp. 12 888–12 900.
- [167] L. Qu, M. Liu, D. Cao, L. Nie, and Q. Tian, “Context-aware multi-view summarization network for image-text matching,” in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1047–1055.
- [168] P. Li, J. Gu, J. Kuen, V. I. Morariu, H. Zhao, R. Jain, V. Manjunatha, and H. Liu, “Selfdoc: Self-supervised document representation learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5652–5660.
- [169] Y. Wang, H. Yang, X. Qian, L. Ma, J. Lu, B. Li, and X. Fan, “Position focused attention network for image-text matching,” in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019, pp. 3792–3798.
- [170] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [171] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” *Advances in neural information processing systems*, vol. 30, 2017.
- [172] D. Shanmugam, D. Blalock, G. Balakrishnan, and J. Guttag, “Better aggregation in test-time augmentation,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 1214–1223.
- [173] P. Oza, P. Sharma, and S. Patel, “Breast lesion classification from mammograms using deep neural network and test-time augmentation,” *Neural Computing and Applications*, pp. 1–17, 2023.
- [174] S. Shleifer, “Low resource text classification with ulmfit and backtranslation,” *arXiv preprint arXiv:1903.09244*, 2019.
- [175] G. Kurata, B. Xiang, and B. Zhou, “Labeled data generation with encoder-decoder lstm for semantic slot filling,” in *INTERSPEECH*, 2016, pp. 725–729.
- [176] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 386–397, 2018.

- [177] A. Gandhi, K. Adhvaryu, S. Poria, E. Cambria, and A. Hussain, “Multimodal sentiment analysis: A systematic review of history, datasets, multimodal fusion methods, applications, challenges and future directions,” *Information Fusion*, vol. 91, pp. 424–444, 2023.
- [178] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, “Visualbert: A simple and performant baseline for vision and language,” *arXiv preprint arXiv:1908.03557*, 2019.
- [179] W. Kim, B. Son, and I. Kim, “Vilt: Vision-and-language transformer without convolution or region supervision,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 5583–5594.
- [180] S. Singh, A. Haridasan, and R. Mooney, ““female astronaut: Because sandwiches won’t make themselves up there”: Towards multimodal misogyny detection in memes,” in *The 7th Workshop on Online Abuse and Harms (WOAH)*, 2023, pp. 150–159.
- [181] L. Chen, “Rit boston at semeval-2022 task 5: Multimedia misogyny detection by using coherent visual and language features from clip model and data-centric ai principle,” in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 2022, pp. 636–641.
- [182] Z. Zhou, H. Zhao, J. Dong, N. Ding, X. Liu, and K. Zhang, “Dd-tig at semeval-2022 task 5: Investigating the relationships between multimodal and unimodal information in misogynous memes detection and classification,” in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 2022, pp. 563–570.
- [183] U. Sahin, I. E. Kucukkaya, O. Ozcelik, and C. Toraman, “Arc-nlp at multimodal hate speech event detection 2023: Multimodal methods boosted by ensemble learning, syntactical and entity features,” in *Proceedings of the 6th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text*, 2023, pp. 71–78.
- [184] Z. Peng, W. Wang, L. Dong, Y. Hao, S. Huang, S. Ma, and F. Wei, “Kosmos-2: Grounding multimodal large language models to the world,” *arXiv preprint arXiv:2306.14824*, 2023.
- [185] P. Wang, A. Yang, R. Men, J. Lin, S. Bai, Z. Li, J. Ma, C. Zhou, J. Zhou, and H. Yang, “Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework,” in *International Conference on Machine Learning*. PMLR, 2022, pp. 23 318–23 340.
- [186] OpenAI, “ChatGPT 4o,” url : <https://platform.openai.com/docs/models/gpt-4o>, 2024.
- [187] J. Yu, K. Chen, and R. Xia, “Hierarchical interactive multimodal transformer for aspect-based multimodal sentiment analysis,” *IEEE Transactions on Affective Computing*, vol. 14, no. 3, pp. 1966–1978, 2023.

-
- [188] N. Majumder, D. Hazarika, A. Gelbukh, E. Cambria, and S. Poria, “Multimodal sentiment analysis using hierarchical fusion with context modeling,” *Knowledge-Based Systems*, vol. 161, pp. 124–133, 2018.
- [189] A. Gandhi, K. Adhvaryu, S. Poria, E. Cambria, and A. Hussain, “Multimodal sentiment analysis: A systematic review of history, datasets, multimodal fusion methods, applications, challenges and future directions,” *Information Fusion*, vol. 91, pp. 424–444, 2023.
- [190] K. Maity, P. Jha, S. Saha, and P. Bhattacharyya, “A multitask framework for sentiment, emotion and sarcasm aware cyberbullying detection from multi-modal code-mixed memes,” in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2022, p. 1739–1749.
- [191] Y. Zhang, P. Tiwari, Q. Zheng, A. E. Saddik, and M. S. Hossain, “A multimodal coupled graph attention network for joint traffic event detection and sentiment classification,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8542–8554, 2023.
- [192] P. Chaudhari, P. Nandeshwar, S. Bansal, and N. Kumar, “Mahaemosen: Towards emotion-aware multimodal marathi sentiment analysis,” *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 22, no. 9, p. 24, 1-24 2023.
- [193] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, “Multimodal transformer for unaligned multimodal language sequences,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 6558–6569.
- [194] S. Bansal, K. Gowda, and N. Kumar, “A hybrid deep neural network for multimodal personalized hashtag recommendation,” *IEEE Transactions on Computational Social Systems*, vol. 10, no. 5, pp. 2439–2459, 2023.
- [195] J. Mu and P. Viswanath, “All-but-the-top: Simple and effective post-processing for word representations,” in *6th International Conference on Learning Representations, ICLR 2018*, 2018.
- [196] S. Mai, H. Hu, and S. Xing, “Divide, conquer and combine: Hierarchical feature fusion network with local and global perspectives for multimodal affective computing,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 481–492.
- [197] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, “Hierarchical attention networks for document classification,” in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 1480–1489.
- [198] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, “The stanford corenlp natural language processing toolkit,” in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55–60.

- [199] C. Sharma, D. Bhageria, W. Scott, S. Pykl, A. Das, T. Chakraborty, V. Pulabaigari, and B. Gambäck, “Semeval-2020 task 8: Memotion analysis-the visuo-lingual metaphor!” in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 2020, pp. 759–773.
- [200] R. Gomez, J. Gibert, L. Gomez, and D. Karatzas, “Exploring hate speech detection in multimodal publications,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2020.
- [201] S. Pramanick, D. Dimitrov, R. Mukherjee, S. Sharma, M. S. Akhtar, P. Nakov, and T. Chakraborty, “Detecting harmful memes and their targets,” in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 2783–2796.
- [202] X. Wei, T. Zhang, Y. Li, Y. Zhang, and F. Wu, “Multi-modality cross attention network for image and sentence matching,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 941–10 950.
- [203] R. K.-W. Lee, R. Cao, Z. Fan, J. Jiang, and W.-H. Chong, “Disentangling hate in online memes,” in *Proceedings of the 29th ACM International Conference on Multimedia*, ser. MM ’21, 2021, p. 5138–5147.
- [204] S. Pramanick, S. Sharma, D. Dimitrov, M. S. Akhtar, P. Nakov, and T. Chakraborty, “Momenta: A multimodal framework for detecting harmful memes and their targets,” in *Findings of the Association for Computational Linguistics: EMNLP 2021*, 2021, pp. 4439–4455.
- [205] H. Lin, Z. Luo, J. Ma, and L. Chen, “Beneath the surface: Unveiling harmful memes with multimodal reasoning distilled from large language models,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 9114–9128.
- [206] N. Xu, W. Mao, and G. Chen, “Multi-interactive memory network for aspect based multimodal sentiment analysis,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 371–378.
- [207] H. Pan, Z. Lin, P. Fu, Y. Qi, and W. Wang, “Modeling intra and inter-modality incongruity for multi-modal sarcasm detection,” in *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020, pp. 1383–1392.
- [208] Z. Liu, A. Braytee, A. Anaissi, G. Zhang, L. Qin, and J. Akram, “Ensemble pretrained models for multimodal sentiment analysis using textual and video data fusion,” in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 1841–1848.
- [209] E. F. Ayetiran and Ö. Özgöbek, “An inter-modal attention-based deep learning framework using unified modality for multimodal fake news, hate speech and offensive language detection,” *Information Systems*, vol. 123, p. 102378, 2024.
- [210] P. K. Roy, “Mmffhs: Multi-modal feature fusion for hate speech detection on social media,” *IEEE Transactions on Big Data*, vol. 11, no. 03, pp. 1247–1258, 2025.

-
- [211] S. Sharma, S. Ramaneswaran, M. S. Akhtar, and T. Chakraborty, “Emotion-aware multimodal fusion for meme emotion detection,” *IEEE Transactions on Affective Computing*, vol. 15, no. 3, pp. 1800–1811, 2024.
 - [212] Y. Li, Y. Li, S. Zhang, G. Liu, Y. Chen, R. Shang, and L. Jiao, “An attention-based, context-aware multimodal fusion method for sarcasm detection using inter-modality inconsistency,” *Knowledge-Based Systems*, vol. 287, p. 111457, 2024.
 - [213] Z. Xi, B. Yu, and H. Wang, “Multimodal sarcasm detection based on sentiment-clue inconsistency global detection fusion network,” *Expert Systems with Applications*, vol. 275, p. 127020, 2025.
 - [214] M. Rahman, M. A.-M. Provath, K. Deb, P. K. Dhar, and T. Shimamura, “Camfusion: Context-aware multi-modal fusion framework for detecting sarcasm and humor integrating video and textual cues,” *IEEE Access*, 2025.
 - [215] A. K. Singh, G. Kim, J. Kim, H.-m. Park, B. J. Choi, and W. De Neve, “Ramm: A residual attention multimodal model for humor detection,” in *International Conference on Intelligent Human Computer Interaction*. Springer, 2025, pp. 229–240.
 - [216] A. Szabo, “The acute effects of humor and exercise on mood and anxiety,” *Journal of leisure research*, vol. 35, no. 2, pp. 152–162, 2003.
 - [217] J. Basler, D. Potó, M. Ferincz, S. Kárpáti, and A. N. Zsidó, “Why aren’t you laughing?—the effect of dark and light humor on anxiety and affective state,” *Personality and Individual Differences*, vol. 240, p. 113133, 2025.
 - [218] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, “Language models are few-shot learners,” *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
 - [219] M. L. Ybarra, K. J. Mitchell, J. Wolak, and D. Finkelhor, “Examining characteristics and associated distress related to internet harassment: findings from the second youth internet safety survey,” *Pediatrics*, vol. 118, no. 4, pp. e1169–e1177, 2006.
 - [220] Y. Zhou, Z. Chen, and H. Yang, “Multimodal learning for hateful memes detection,” in *2021 IEEE International conference on multimedia & expo workshops (ICMEW)*. IEEE, 2021, pp. 1–6.
 - [221] Y. Tang, L. Zhang, Q. Teng, F. Min, and A. Song, “Triple cross-domain attention on human activity recognition using wearable sensors,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 6, no. 5, pp. 1167–1176, 2022.
 - [222] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter,” *arXiv preprint arXiv:1910.01108*, 2019.
 - [223] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, “Llama: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971*, 2023.

- [224] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *International Conference on Learning Representations*, 2021.
- [225] Qwen Team, “Qwen2.5-7B-Instruct,” <https://huggingface.co/Qwen/Qwen2.5-7B-Instruct>, 2025, [Online; accessed 2-June-2025].
- [226] H. Liu, C. Li, Q. Wu, and Y. J. Lee, “Visual instruction tuning,” *Advances in neural information processing systems*, vol. 36, pp. 34 892–34 916, 2023.
- [227] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, “Imagebind: One embedding space to bind them all,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 15 180–15 190.
- [228] P. K. Diederik, “Adam: A method for stochastic optimization,” (*No Title*), 2014.
- [229] J. D. M.-W. C. Kenton and L. K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of naacL-HLT*, vol. 1, no. 2. Minneapolis, Minnesota, 2019.
- [230] A. Radford, “Improving language understanding by generative pre-training,” 2018.
- [231] A. Dosovitskiy, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.
- [232] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, “Vivit: A video vision transformer,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6836–6846.
- [233] S.-Y. Jung, C.-H. Liao, Y.-S. Wu, S.-M. Yuan, and C.-T. Sun, “Efficiently classifying lung sounds through depthwise separable cnn models with fused stft and mfcc features,” *Diagnostics*, vol. 11, no. 4, p. 732, 2021.
- [234] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.
- [235] H. Zhang, X. Li, and L. Bing, “Video-llama: An instruction-tuned audio-visual language model for video understanding,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2023, pp. 543–553.
- [236] C. Dixit and S. M. Satapathy, “Deep cnn with late fusion for real time multimodal emotion recognition,” *Expert Systems with Applications*, vol. 240, p. 122579, 2024.
- [237] L. Wang, J. Peng, C. Zheng, T. Zhao *et al.*, “A cross modal hierarchical fusion multimodal sentiment analysis method based on multi-task learning,” *Information Processing & Management*, vol. 61, no. 3, p. 103675, 2024.

- [238] Z. Li, Q. Guo, Y. Pan, W. Ding, J. Yu, Y. Zhang, W. Liu, H. Chen, H. Wang, and Y. Xie, “Multi-level correlation mining framework with self-supervised label generation for multimodal sentiment analysis,” *Information Fusion*, vol. 99, p. 101891, 2023.
- [239] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, “Multimodal transformer for unaligned multimodal language sequences,” in *Proceedings of the conference. Association for computational linguistics. Meeting*, vol. 2019. NIH Public Access, 2019, p. 6558.
- [240] M. Z. U. Rehman, D. Raghuvanshi, H. Pachar, C. S. Raghaw, and N. Kumar, “Hierarchical attention-enhanced contextual capsulenet for multilingual hope speech detection,” *Expert Systems with Applications*, vol. 268, p. 126285, 2025.

