Master of Science in Computer Science and Engineering


# MULTIMODAL DEEP LEARNING APPROACH FOR HUMOR DETECTION IN BENGALI

*by*

Musfequa Rahman

ID: 22MCSE003

This thesis is submitted in partial fulfillment of the requirement for the degree of
MASTER OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

Department of Computer Science and Engineering
**Chittagong University of Engineering and Technology**
Chattogram-4349, Bangladesh.
May, 2025

# CERTIFICATION

The thesis titled **"MULTIMODAL DEEP LEARNING APPROACH FOR HUMOR DETECTION IN BENGALI"** submitted by **Musfequa Rahman**, Roll No **22MCSE003**, Session **(2022-2023)** has been accepted as satisfactory in partial fulfillment of the requirement for the degree of MASTER OF SCIENCE IN COMPUTER SCIENCE & ENGINEERING on 20-05-2025

**BOARD OF EXAMINER**

| | |
|---|---|
| 1. _____<br>Dr. Kaushik Deb<br>Professor<br>Department of Computer Science & Engineering<br>Chittagong University of Engineering & Technology<br>Chattogram- 4349, Bangladesh. | Chairman<br>(Supervisor) |
| 2. _____<br>Dr. Pranab Kumar Dhar<br>Professor & Head<br>Department of Computer Science & Engineering<br>Chittagong University of Engineering & Technology<br>Chattogram- 4349, Bangladesh. | Member<br>(Ex-Officio) |
| 3. _____<br>Dr. Muhammad Ibrahim Khan<br>Professor<br>Department of Computer Science & Engineering<br>Chittagong University of Engineering & Technology<br>Chattogram- 4349, Bangladesh. | Member |
| 4. _____<br>Dr. Pranab Kumar Dhar<br>Professor<br>Department of Computer Science & Engineering<br>Chittagong University of Engineering & Technology<br>Chattogram- 4349, Bangladesh. | Member |
| 5. _____<br>Dr. Md. Nazrul Islam Mondal<br>Professor<br>Department of Computer Science and Engineering<br>Rajshahi University of Engineering & Technology<br>Rajshahi-6204, Bangladesh | Member<br>(External) |

# Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis or any part of it has not been submitted elsewhere for the award of any degree or diploma.

Author: Musfequa Rahman

(signature)

Date: 20-05-2025

*It is my genuine gratefulness and warmest regard that*
*I dedicate this work to*
***My Beloved Family, Especially My Mother***

# Table of Contents

# List of Figures

# List of Tables

# List of Algorithms

# List of Terms and Abbreviations

| | |
|---|---|
| **OCR** | Optical Character Recognition |
| **CNN** | Convolutional Neural Network |
| **Bi-LSTM** | Bidirectional Long Short-Term Memory |
| **Bi-GRU** | Bidirectional Gated Recurrent Unit |
| **CAMFusion** | Context-Aware Multi-Modal Fusion Framework |
| **BATE** | Bangla Audio-Text Extractor |
| **CF** | Concatenation Fusion |
| **CA** | Cross Attention |

# Acknowledgement

I'm ecstatic to have this chance to express my gratitude to those whose unwavering support and encouragement helped me finish my thesis on sports video classification.

First and foremost, I want to express my heartfelt gratitude to my supervisor, Prof. Dr. Kaushik Deb, for his unwavering assistance throughout the research process. Without his instructive supervision, this report would not have seen the light of day. I am eternally grateful to the honorable examination board member for their invaluable suggestion.

I express my heartfelt gratitude and respect to my family for their unwavering kindness, sacrifice, and unwavering support throughout the working period.

Finally, I want to thank Allah, the Almighty, for giving me the strength and courage to complete the task and achieve the goal.

# Abstract

Humor and sarcasm detection is essential for intelligent agents to revolutionize customer service, healthcare, and other industries by enabling truly natural and empathetic interactions. However, detecting sarcasm and humor in Bengali videos, one of the most spoken languages in the world, is significantly challenged by the language's intricate linguistic nuances and cultural specificity. As per our knowledge, this work introduces the first publicly available multimodal dataset for Bengali humor detection, consisting of 600 annotated video clips categorized into humor, sarcasm, and normal. Context-Aware Multi-Modal Fusion Framework (CAMFusion) is proposed, integrating visual and textual features. Visual features are extracted from a Time-Distributed MobileNetV2 with 0.985 million parameters, enhanced with reverse fusion mechanism and squeeze and excitation blocks. In contrast, textual features are processed through Bi-LSTM and Bi-GRU networks with group-wise enhancement. A new Bengali Text Extraction Algorithm ensures robust text retrieval from complex video frames. CAMFusion achieves an accuracy of 90% through dynamic attention-based fusion with 13.88 million parameters, significantly outperforming state-of-the-art baselines, including MobileNetV3 with XLM-R (81.10%) and TimeSformer-L with DeBERTaV3 (84.45%). This work establishes a new benchmark for Bengali humor detection, addressing a critical gap in low-resource language research and advancing multimodal computational linguistics for culturally sensitive AI applications.

**Keywords:** Bengali Humor Detection, Text Extraction, Multi-Modal Fusion, Context-Aware Analysis, Human-Computer Interaction.

# Chapter 1

# Introduction

Humor and sarcasm are critical aspects of human communication and interaction that facilitate the connection between people, expressing people's feelings, and adjusting to social situations. Laughter is a common gateway to humor that can create a (temporary) bond by the shared experience of enjoyment and reduce tension, and build rapport. In addition to its function of entertainment, humor can provide psychological advantages and serve as a stress reducer,a social facilitator, and an emotional enhancer, as has been ascertained by studies related t o its therapeutic effects [1]. For example, a well-timed joke can dispel tension when discussing something stressful, leaving you with happier, more connected conversations. Sarcasm, however, is a more complex type of discourse and can reflect irony, where the expressed words carry an opposite intention from the speaker. This is meant as a mild form of criticism, one that the listener will need to succeed and try to understand the true meaning of the speaker, which serves to be really sharp and certainly critical too [2]. For instance, stating, "Good job! ' in a characterizing tone of mockery following a mistake relies on the audience's perception of the mismatch between the words and the speaker's feelings. This duality is what makes sarcasm complicated; it is amusing at one time and potentially alienating or offensive at the next, depending on how it is interpreted [3].

In today's digital era, where interactions increasingly occur on platforms like social media, video-sharing sites, and messaging apps, humor and sarcasm have become integral to online communication [4]. These memes, short video clips, or witty comments on Instagram or Twitter frequently employ those terms to resonate with users. But with the transition to digital communication, things get tricky because humor and sarcasm hinge on non-verbal cues like tone, facial expressions, and gestures, which are frequently missing in text. This transition highlights the importance to AI (Artificial Intelligence) systems to correctly understand and perceive these facial expressions, allowing more human-like human-computer interactions in digital worlds. The context here is Bengali, spoken by millions of speakers in Bangladesh and some parts of India, and in the cultural and linguistic way, humor and sarcasm are embedded in the language, yielding an interesting challenge that needs to be addressed to adapt to expressions like these in multimodal settings such as videos.

## 1.1 Humor Detection System

Systems designed to detect humor and sarcasm aim to enable machines to recognize and understand these expressions in digital media, a task that is particularly complex for Bengali videos. Such systems must handle multimodal data, combining visual elements—such as facial expressions, gestures, and physical appearances—with textual or linguistic components, like dialogue, subtitles, or captions, to achieve precise detection [5]. For example, a video might feature a person delivering a sarcastic comment with a straight face, where the humor arises from the contrast between the serious expression and the exaggerated statement. Traditional methods often focused on single modalities, either analyzing text or visuals in isolation.

Text-based approaches might examine sentence structure to detect humorous wordplay or a sarcastic tone, while visual methods might identify exaggerated facial expressions or comedic movements [5], [6]. However, these unimodal strategies are limited because humor and sarcasm often emerge from the interaction between modalities. A multimodal system, by contrast, can better capture this interaction, as shown by datasets like UR-FUNNY [7], which integrate text, audio, and video inputs to enhance detection accuracy in English and Chinese contexts. These datasets demonstrate that combining multiple modalities can significantly improve performance, with some models achieving over 80% accuracy in humor and sarcasm detection [8]. Moreover, for Bangla videos, multimodality is more crucial because the cultural influence of the expressions of comedy and sarcasm has to be incorporated. For example, in a Bangla comedy skit, the exaggerated movement of the body is often combined with a witty one-liner in a local dialect but neither would capture the complete intended humor separately. Limited work on Bangla multimodal systems so far, this necessitates the development of systems that will be able to handle the key phonetic, syntactic, and cultural elements of the language, such that both the visual as well as the textual information are understood in a culturally relevant manner.

## 1.2 Motivation

This research is driven by the urgent need to address critical shortcomings in multimodal humor and sarcasm detection for Bengali, a language with distinct characteristics that are often overlooked in AI research. The primary motivations are outlined below:

- **Limitations of Existing Frameworks for Bengali**:
  - Multimodal frameworks like MISA [8] and UR-FUNNY [7] have achieved over 80% accuracy in English and Chinese contexts, but are ill-suited for Bengali's unique features.
  - For example, Bengali sarcasm often depends on subtle tonal variations or cultural references, such as using a proverb to mock a social norm, which English-focused models struggle to interpret [9].

- **Computational Constraints of Current Models**:
  - Methods like 3D CNNs, while effective for video analysis, require substantial computational resources, making them impractical for deployment on mobile devices widely used by Bengali-speaking communities [10], [11].
  - This is a significant issue in regions with limited access to advanced hardware, necessitating models that balance performance with efficiency.
- **Scarcity of Specialized Bengali Datasets**:
  - Existing datasets like Memotion [12] and ColBERT [13] focus on specific genres such as memes or online conversations in major languages, failing to capture the multimodal nature of Bengali video content, including skits, vlogs, or comedic performances.
  - The lack of annotated Bengali data impedes the development of generalizable models, as the language's complex morphology, phonetic diversity, and socio-cultural context require specialized training data.
- **Increasing Demand in Bengali-Speaking Communities**:
  - The growing prevalence of digital communication in Bengali-speaking regions, with more creators producing video content in Bengali, has heightened the need for AI systems that can understand and engage with this content in a culturally relevant manner.

This research seeks to bridge these gaps by developing a context-aware multimodal framework for Bengali, utilizing efficient techniques like Time-Distributed MobileNetV2 and advanced text extraction methods to ensure both accuracy and practical applicability.

## 1.3 Objectives

This work aims to develop an efficient, accurate, and culturally sensitive system for detecting humor and sarcasm in Bengali videos, addressing the limitations of existing multimodal frameworks and datasets. The specific objectives are:

1. **Establish a Dedicated Bengali Multimodal Dataset**: Develop a new dataset comprising 600 annotated video clips categorized into three classes (Humor, Sarcasm, Normal) to address the scarcity of Bengali-specific resources, ensuring representation of cultural and linguistic nuances in humor and sarcasm expressions.
2. **Design an Advanced Text Extraction Algorithm**: Propose a Bengali Text Extraction Algorithm utilizing custom keyframe selection based on intensity differences, combined with optimized Optical Character Recognition (OCR), to accurately extract textual content from complex video frames, enhancing the detection of humor- and sarcasm-related cues.
3. **Develop and Evaluate Multimodal Model**: Design the Context-Aware Multi-Modal Fusion Framework (CAMFusion), integrating Time-Distributed MobileNetV2 with

attention mechanisms for visual feature extraction and Bi-LSTM/Bi-GRU layers with group-wise enhancement for textual features, achieving 90% accuracy, and benchmark its performance against existing models to demonstrate superior accuracy.

These objectives are strategically formulated to address the critical gaps identified in the literature, particularly the lack of Bengali-specific datasets, the computational inefficiencies of current models, and the need for culturally sensitive multimodal approaches, thereby advancing human-machine interactions in Bengali-speaking digital environments.

## 1.4 Scope of Work

The scope of this work is strategically defined to address the critical need for automated detection and classification of humor and sarcasm in Bengali video content, with a focus on enhancing AI-driven interactions in digital platforms prevalent in Bengali-speaking communities. This study concentrates on developing a multimodal system capable of recognizing and interpreting humor and sarcasm by leveraging visual and textual cues, overcoming the limitations of existing unimodal and non-Bengali-specific frameworks.
The work utilises a newly curated Bengali Multimodal Dataset containing 600 video clips (obtained and labelled from multiple sources) categorised into three classes; Humour, Sarcasm and Normal. These categories have been devised by the authors to cover the full spectrum of humor and sarcasm found in Bengali, with consideration to the language's phonetic, syntactic and socio-cultural nature. Comedic content to get laughs, whether that represented as a pie in the face, or a pun, or a reference to something or someone currently popular or in fashion. The Sarcasm topic The former (Sarcasm) includes (non)explicitly ironic or sarcastic statements, usually induced by tonality hints, or by information discrepancy between two modalities: the multimedia system generates visual and textual counterpart modalities, and when they differ, sarcasm is introduced. The Normal class is for non-funny and non-sarcastic utterances and has been treated as a threshold in distinguishing neutral for non-emotional and non-expressive elements of discourse. Together, these tags authenticate the multi-dimentional nature of Bengali humorous sarcastic expressions and supports a model which can be generalized on many forms of expressions meanwhile the specificity in classification has been also taken into consideration.

This research also emphasizes computational complexity for making the approach practical on low-end devices to make use of language technology such as mobile phones, which are very common in Bengali speaking area. Stated differently, to maintain computational amount as low as possible, this paper detects heart sound by the proposed Context-Aware Multi-Modal Fusion Framework (CAMFusion) by fusing Efficient Fusion Time-Distributed

MobileNetV2 (0.985 million parameters) with a high textual feature extraction (Bi-LSTM and Bi-GRU layers) in the majority of time. The scope is restricted to a video-based analysis of visual (e.g., facial expressions, gestures) and textual (e.g., subtitles, captions) input data, leaving audio-based information out of consideration to make the analysis as concentrated as possible on visual-textual interactions. This exclusion is because the focus in this paper is only on visual  textual modalities, and inclusion of audio processing might add more complexity without necessarily improving the detection accuracy in this scenario.

We compare the performance of the system with that of the state-of-the-art baselines trained using datasets like UR-FUNNY and MISA, and recently released Bengali Multimodal Dataset to verify the effectiveness and cultural preference of the proposed system. No other languages or cultural context is included in the research, only Bengali, to maintain the depth and specificity of the study. Within this narrow focus, the goal of the research is to successfully deliver an efficient, accurate, and culturally suitable platform for humor and sarcasm detection, leading to an improvement in digital communication, content understanding, and linguistic richness in AI applications designed for Bengali-speaking societies.

## 1.5 Challenges

Detecting humor and sarcasm in Bengali videos presents numerous obstacles for AI systems due to the intricate nature of these expressions and the distinct features of the Bengali language. The key challenges are as follows:

- **Context-Dependent Nature of Humor and Sarcasm**:
  - Humor and sarcasm rely heavily on context, non-verbal cues, and cultural nuances, which are challenging to capture using a single modality.
  - For instance, a sarcastic remark like “What a wonderful day!” made during a heavy rain might be misclassified as serious by a visual-only model focusing on the weather, or as genuine by a text-only model without the visual context of the rain.
  - The interaction between modalities is essential for accurate detection, but many existing systems struggle to integrate them effectively [14].
- **Lack of Comprehensive Bengali Datasets**:
  - Existing multimodal datasets like UR-FUNNY [7], MUMOR [8], M2H2 [15], Memotion [12], and ColBERT [13] are primarily designed for English or other major languages like Chinese.
  - These datasets lack the cultural and linguistic diversity needed to represent Bangla, where humor often involves wordplay, idiomatic expressions, and references to local traditions that are not easily translatable [9].

- **Computational Demands of Multimodal Models**:
  - Techniques like 3D CNNs, often used for video analysis, can capture spatial and temporal features but require significant memory and processing power.
  - This makes them unsuitable for real-time applications on resource-constrained devices [10].
  - Moreover, detecting subtle expressions like sarcasm often requires large temporal window sizes, further increasing computational demands [11].
- **Challenges in Text Extraction from Bengali Videos**:
  - Traditional optical character recognition (OCR) systems are not optimized for the dynamic nature of video frames, where text may appear briefly or be overlaid on complex backgrounds, such as in a busy market scene [16], [17].
  - This makes it difficult to extract subtitles or on-screen text that might contain humorous or sarcastic content.
- **Socio-Cultural Context of Bengali**:
  - Humor and sarcasm in Bengali are often tied to regional customs, historical references, and social norms, which are not easily captured by models trained on generic datasets.
  - This cultural specificity adds a unique layer of complexity to the detection process.

These challenges underscore the need for a specialized approach to humor and sarcasm detection in Bengali, addressing both technical and cultural aspects.

## 1.6 Applications of Humor Detection System

The development of effective humor and sarcasm detection systems for Bengali videos offers a wide range of applications, particularly in improving human-machine interactions in digital environments. The main applications are outlined below:

- **Enhancing AI-Driven Communication Tools**:
  - Accurate detection of humor and sarcasm enables machines to respond appropriately to users' emotional states, fostering more natural and engaging interactions.
  - For example, a chatbot on a Bangla messaging app could recognize a user's sarcastic comment and respond with a witty reply instead of a literal one, improving the user experience.
- **Improving Content Moderation and Sentiment Analysis**:
  - Humor and sarcasm significantly influence the tone of online discussions, and misinterpreting them can lead to incorrect sentiment classification or inappropriate moderation decisions [18].

  - A detection system for Bangla videos can better assess the emotional context of user-generated content, ensuring accurate moderation and reducing the risk of miscommunication.
- **Enabling Personalized Content Recommendations**:
  - By identifying users' preferences for humorous or sarcastic content, these systems can tailor video recommendations on platforms like YouTube to match individual tastes, enhancing user engagement.
- **Supporting Educational Tools**:
  - Humor detection systems can be used to develop interactive learning tools that incorporate culturally relevant humor, making lessons more engaging for Bengali-speaking students.
- **Contributing to Mental Health Support**:
  - AI systems that understand humor can provide light-hearted interactions to reduce stress, aligning with the therapeutic benefits of humor noted in prior studies [1].
- **Bridging Cultural Gaps in Bengali Contexts**:
  - In the rapidly growing digital content creation landscape of Bangla, these systems can make AI interactions more accessible and relatable by ensuring sensitivity to local expressions of humor and sarcasm [19].

These applications highlight the potential of humor and sarcasm detection systems to enhance digital interactions for Bengali-speaking audiences.

## 1.7 Contribution of the Thesis

This thesis makes several significant contributions to the field of multimodal humor and sarcasm detection in Bengali:

1. To our knowledge, created the first multimodal dataset for Bengali sarcasm and humor detection, consisting of 600 video clips annotated across three categories: humor, sarcasm, and normal. This dataset fills a critical gap in the field, providing a valuable resource for training and evaluating models tailored for Bengali sarcasm and humor detection.
2. Designed an innovative Bengali Text Extraction Algorithm for extracting text from complex video frames, employing a custom keyframe selection method based on intensity differences. Combined with OCR, this algorithm improves the accuracy of detecting humor- and sarcasm-related content embedded in video frames.
3. Proposed a multimodal Context-Aware Multi-Modal Fusion Framework (CAMFusion), which integrates Efficient Fusion Time-Distributed MobileNetV2 with 0.985 million parameters for visual feature extraction, enhanced with attention mechanisms, and combines Bi-LSTM and Bi-GRU layers with group-wise enhancement

for textual features. The model achieves a 90% accuracy, significantly outperforming existing sarcasm and humor detection approaches for Bengali.

4. Presented a comparative analysis of the performance of our proposed approach with other baseline approaches.

## 1.8 Thesis Organization

The rest of this thesis is structured as follows. Chapter 2 provides a literature review on related technologies and methodologies for humor and sarcasm detection, including a review of similar works in multimodal AI systems. Chapter 3 presents the proposed architectures for humor detection, with detailed explanations of their components, including the CAMFusion model. Chapter 4 details the implementation of these models, covering system specifications, model configurations, training procedures, and an analysis of model complexity and performance metrics. Chapter 5 presents the experimental results and evaluations of the proposed frameworks on multiple Bengali multimodal datasets, using various precision measures, and includes in-depth analyses such as ablation studies and generalizability tests. Finally, Chapter 6 concludes the overall work, summarizing the contributions of this research and suggesting potential directions for future work in this domain.

## 1.9 Conclusion

This section gives some contributions of the humor and sarcasm detection system for Bangla videos in brief that shows the challenges faced while implementing the multimodal architecture. In addition, we provide several applications of humor and sarcasm detection, including (a) making human-machine communication more natural (b)content moderation for enhanced communication or providing summaries of news reports. Finally, The motivation and the contributions of the research are presented mentioning the motivation for culturally sensitive AI systems in Bangla language and the progress made by the proposed framework.

# Chapter 2

# Literature Review

## 2.1 Introduction

Humor and sarcasm detection from digital texts, especially from Bangla videos remain a challenging yet emerging area of research in AI. These expressions are crucial in human communication and their interpretation often goes beyond mere text messages, but requires subtle emotional and/or contextual factors across multi-mode signals, such as texts, speeches and images. In case of Bangla, a language with vibrant culture and linguistic variety, the complexity increases since it is also required to capture visual and textual features properly. For example, it may be funny to make a joke and at the same time use grimace face, or to be sarcastic by speaking with one's mouth but have a visual scene that contradicts the speaker's tone, (e.g., saying something sarcastic about the weather during a storm). Conventional monomodal methods have difficulty in capturing these rich structure and much noisy cases show misinterpretation. [20]. Recent advancements in deep learning, graph networks, self-supervised learning, multimodal approaches, and fusion techniques have shown promise in addressing these challenges. Models like Affect-GCN [21] and MMLATCH [22] have demonstrated the power of integrating multiple modalities to achieve higher accuracy in emotion and humor recognition. However, these methods still face significant hurdles, including high computational costs, dataset limitations, and the need for cultural adaptation, especially for underrepresented languages like Bangla [15]. This section explores various techniques applied in the detection of sarcasm and humor, highlighting their strengths, limitations, and potential for improving human-machine interactions in culturally diverse contexts.

## 2.2 Deep Learning and Graph Networks

Graph-based learning has made significant advancements in the field of multimodal emotion recognition. The Affect-GCN model applied GCNs to dialogue-based data, where emotions often overlap, achieving an accuracy of 87%. This model showed how GCNs can capture multiple emotions of varying intensities in complex interactions [21]. Another model employed hybrid fusion techniques combining speech and image data for emotion

recognition, with a focus on interpretability. This approach achieved an accuracy of 85% and emphasized the importance of hybrid fusion in creating interpretable AI systems [23]. The development of a multimodal graph convolutional network for detecting humor in conversations achieved an accuracy of 88%, showing the potential of graph-based learning in capturing contextual relationships in humorous interactions [24]. The model introduced modality-invariant and modality-specific representations, enhancing the robustness of multimodal analysis [25], [26]. By fusing text, audio, and video inputs, the model achieved an accuracy of 83%, showcasing the strength of advanced fusion techniques [8].
Graph-based methods have significantly augmented emotion and humor recognition. For instance, models like Affect GCN based on graph convolutional networks have become very efficient in handling complex emotional interactions with impressive accuracy rates. Despite these advantages, there are a variety of challenges faced by such models. One key challenge is the fact that this approach is computationally expensive since GCNs require huge processing power. Hence, applying them in real-time applications may not be quite feasible. Besides, the preoccupation with large training datasets constrains generalization, especially to data from other cultural or emotive backgrounds. That is, models struggle the most with unfamiliar data when that data does not match well with the training set. Besides, hybrid approaches to fusion generally introduce some sort of trade-off between performance and usability in enhancing interpretability mechanisms for AI systems, where the high accuracy might be compromised on tasks requiring real-time or low-latency responses. These are only a few of the many issues that indicate the need for efficiency and scalability of models to assure high accuracy in performance while adapting to variable data sources and contextual understanding.

## 2.3 Intercultural and Self-Supervised Learning

Research into humor in intercultural interactions focused on whether humor serves as common ground or a source of misunderstanding. The advent of self-supervised learning has revolutionized emotion recognition, particularly in speech. One study demonstrated that self-supervised learning models can achieve up to 85% accuracy in speech emotion classification without requiring large labeled datasets [27]. A survey on audio self-supervised learning discussed its applications in multimodal tasks such as emotion recognition, achieving an accuracy of 83%. The survey highlighted the versatility of self-supervised learning across various modalities, including speech and audio [28]. Research targeted humor in intercultural interactions as common ground or a source of misunderstanding. This study achieved 78% accuracy, indicating that cultural adaptation is an essential concern when creating models for humor detection [29]. Facial image analyses for the estimation of Big Five personality traits have shown that the inclusion of multimodal inputs like facial expressions and speech, will lead to the rise of the estimation accuracy of personality traits as high as 80% [30]. Another work combined acoustic and linguistic features, after which

a supervised autoencoder boosted the improvement in emotion recognition, obtaining 85% in contexts of music videos [31]. The era of self-supervised learning has revolutionized emotion recognition, especially in speech. One such work demonstrated the performance of self-supervised learning models on speech emotion classification, with results as high as 85% without requiring large labeled datasets [27]. A survey on audio self-supervised learning presented its applications to multimodal tasks such as emotion recognition and obtained an accuracy of 83%. This survey showed the flexibility of self-supervised learning across different modalities, speech, and audio [28].

## 2.4 Multimodal Approaches

The M2H2 dataset for humor detection in Hindi conversations, which includes multimodal data such as text and audio, achieved an accuracy of 81%. This dataset is a key resource for developing multimodal humor detection models in non-English languages [15]. Early work in humor detection included computational recognition of wordplay in-jokes, achieving an accuracy of 75%. This study laid the foundation for more sophisticated approaches to computational humor recognition [32]. Subsequent work using deep learning for humor detection in textual data achieved an accuracy of 80%, demonstrating how neural networks can capture the nuances of humor in text [33]. The ColBERT model, which used BERT sentence embeddings in a parallel neural network architecture, improved humor detection accuracy to 82%. This work underlined the power of transformer-based models in understanding the complexities of humor in language [13].

Indeed, gains are significant with multimodal approaches that combine speech, video, and motion capture inputs. For instance, one model performed such an approach by fusing the speech and video inputs with the mocap data for better representation of complicated emotions and achieved an improvement of 15% in accuracy when compared to single-modality models [34]. Similarly, MMLATCH proposed a bottom-up and top-down fusion strategy for multimodal sentiment analysis, thereby scoring an accuracy of 85%. It has been noted by the model that a potent improvement could be achieved by methods of fusion in multimodal sentiment recognition watering hole [22]. Another, the Convolutional MKL-based model combined the input text and video on emotion analysis and sentiment, yielding the highest accuracy of 83%. It has been proved that with the fusing of several modalities, performance improved much in emotion recognition and sentiment analysis watering hole [35].

The UR-FUNNY dataset is a key resource for humor recognition tasks, combining text, audio, and visual inputs. Models trained on this dataset achieved an accuracy of 85%, setting a benchmark for further research in multimodal humor detection [36]. Multimodal sarcasm detection, which combines textual and visual information, achieved an accuracy of 83%. This study emphasized the need for diverse datasets to capture the complexities of sarcasm detection [37].

## 2.5 Fusion Techniques

A hybrid multimodal fusion approach combining text and video inputs for sarcasm detection reached an accuracy of 86%. This method demonstrated the importance of selecting the right fusion strategy for detecting complex social cues like sarcasm [20]. Another study explored continuous emotion prediction from video data using pose feature engineering, achieving an accuracy of 84%. This research emphasized the importance of non-verbal cues such as posture and movement in detecting emotions and humor in videos [38]. The use of transformer-based architectures in humor detection led to an accuracy of 88%, demonstrating the ability of transformers to contextualize complex linguistic structures, improving humor detection accuracy [39].

Additionally, a late fusion model combining inputs from music videos for emotion classification achieved an accuracy of 88%. This study illustrated how various media types can be effectively combined to represent emotionally intense scenarios [40], [41]. The Memotion 2 dataset was used to perform sentiment and emotion analysis on memes, achieving an accuracy of 87%. This study highlighted how emotion and humor recognition can adapt to new formats like memes [42].

There is an indication of substantial improvement in detecting humor and sarcasm using multi-modal approaches, mainly in Bangla video content. While the methodologies developed are innovative, limitations persist on dataset constraints, high computational costs, and generalization issues. These challenges signal a continued research direction, at least in languages other than English, toward improving the robustness of multi-modal humor and sarcasm detection systems.

## 2.6 Conclusion

This study has explored a range of techniques for detecting humor and sarcasm, with a particular focus on their application to Bangla video content. Deep learning and graph-based methods, such as Affect-GCN [21] and multimodal graph convolutional networks [24], have demonstrated high accuracy in capturing complex emotional and humorous interactions, achieving up to 88% accuracy. Self-supervised learning has proven effective in speech emotion recognition, with models achieving 85% accuracy without the need for large labeled datasets [27], while intercultural studies highlight the importance of cultural adaptation in humor detection [29]. Multimodal approaches, including datasets like UR-FUNNY [36] and M2H2 [15], have shown significant improvements by integrating text, audio, and visual inputs, with accuracies reaching 85%. Fusion techniques, such as hybrid and late fusion strategies, have further enhanced performance, achieving up to 88% accuracy in emotion and humor detection across various media types [20], [40]. Despite these advancements, challenges remain, including high computational costs, limited datasets for non-English languages like Bangla, and difficulties in generalizing across

diverse cultural contexts. These limitations underscore the need for continued research to develop more efficient, scalable, and culturally sensitive models. Future work should focus on creating comprehensive Bangla datasets, optimizing computational efficiency for real-time applications, and exploring advanced fusion techniques to better capture the subtleties of humor and sarcasm in multimodal settings.

# Chapter 3

# Proposed Architecture

## 3.1 Introduction

The identification of sarcasm and humor of video content is a challenging problem, particularly in case of low-resource video content such as Bengali, due to the fact that videos are multimodal in nature, where various forms of modalities are involved, including visual and textual modalities, and also because cultural references or differences are carried in visual and text cues. Conventional techniques do not necessarily represent the interaction between the visual content like the facial gesture and the lexical information like dialogues or on-screen text in a linguistically diverse setting. To resolve this, the methodology for Bengali sarcasm andhumor detection from videos consists of following:

processed pipeline, inwhich data preprocessing, feature extraction, and multimodal fusion are included in Figure 3.1. The steps are outlined below:

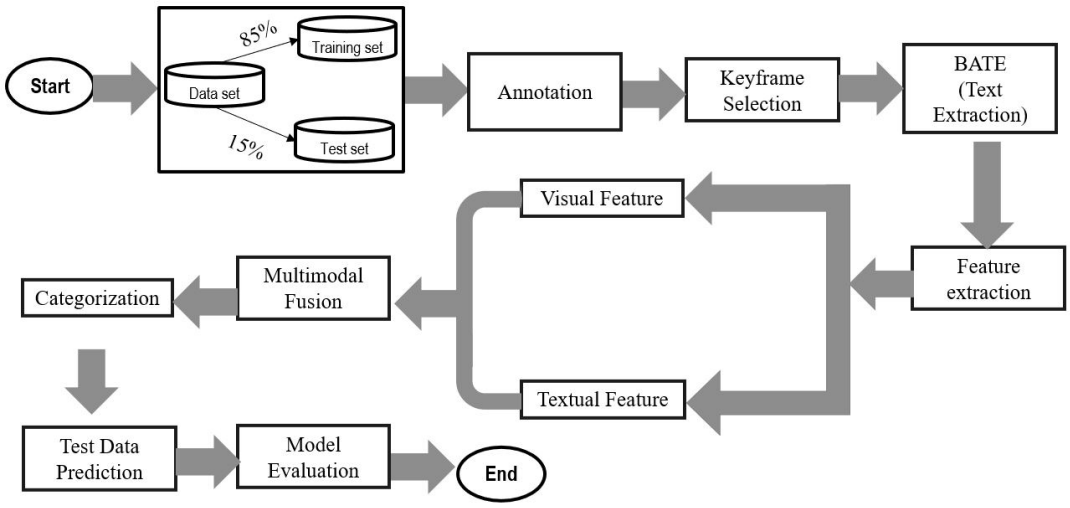



Figure 3.1. Overall Workflow for Bengali Humor Detection.

1. **Dataset Splitting**: The process begins by dividing the dataset into a training set (85%) and a test set (15%) to facilitate model training and evaluation.

2. **Annotation and Keyframe Selection**: The training set is annotated to label sarcasm and humor. Keyframes are then selected from videos to capture the most representative frames containing relevant visual and textual cues.
3. **Bangla Text Extraction (BATE)**: An algorithm extracts text from complex video frames, enabling the capture of textual context such as subtitles or on-screen text critical for sarcasm and humor detection.
4. **Feature Extraction**: This step involves advanced models such as Sept13D+BanglaBERT, Lite-MDETR, Timeformer-L+DeBERTaV3, and CAMFusion combined with two parallel processes:
   - **Visual Feature Extraction**: Extracts visual features from keyframes, focusing on salient regions like facial expressions or gestures.
   - **Textual Feature Extraction**: The extracted text is processed to capture linguistic and contextual dependencies.
5. **Multimodal Fusion**: A Fusion Framework integrates the visual and textual features, preserving the interplay between modalities to better understand sarcasm and humor.
6. **Model Training and Evaluation**: The fused features are used to train a classification model on the training set. The model is then evaluated on the categorized test set to assess its performance.
7. **Test Data Prediction**: Finally, the trained model makes predictions on the test data, concluding the pipeline.

## 3.2 Problem Formulation

Given a set of videos $V = \{V_1, V_2, \ldots, V_{|V|}\}$, each video $V_i$ is represented by a sequence of frames and the corresponding dialogues, denoted as $X_i$. Each $X_i$ consists of a combination of visual ($v$) and textual ($t$) features, leading to a multi-modal feature vector $U_m$ for each modality $m$, where $m \in \{v, t\}$. The goal is to learn a mapping function $f$ that assigns the correct class label $Y_i$ to each video $V_i$.
The task is formulated as a multi-class classification problem, where for each video $V_i$, the aim is to learn a function $f : X_i \rightarrow Y_i$ that best captures the underlying relationships within the data. The model's performance is governed by a loss function $L(f(X_i), Y_i)$, which quantifies the divergence between the predicted labels and the true labels.

## 3.3 Data Collection

The Bangla Humor and Sarcasm Detection Dataset is distinctive for itsspecificity in the Bangla language and a multi-class approach compared to binary classifications of humor or emotion only. The dataset enables the capture of humour and sarcasm expression and reception in a B-angla n language 7, thereby adding to the knowledge corpus for computational linguistics and cultural ana-lysis. The dataset was carefully assembled

to obtain the multimodal property of humor and sarcasm in Bangla-videos. Particular attention was paid to the synchronization between the graphic and the text so that both modalities are properly synchronized in time and display the subtle relationship between words and the graphic. Textual information from in-band caption was strictly aligned with its visual correspondence consistently in the whole dataset. This incorporation of multiple modalities allows for a smooth fusion of linguistic and visual context, a necessity in rich tasks such as sentiment and humor prediction.

The annotation was done by native Bangla speakers having sufficient cultural knowledge of humor and sarcasm. Every video clip was rated separately as humor, sarcasm or normal. The labeling was carried out by multiple annotators to ensure consistency and reliability. Any discrepancies were decided with a majority vote to improve the due quality and accuracy of annotations. This strictly annotated framework ensures that the dataset accounts for multi-modal nature of humor/sarcasm, as well as carries over the nuances of the cultural expressions of Bangla.

Figure 3.2 illustrates samples from each class in the dataset, showcasing typical humor, sarcasm, and normal content in Bangla. In addition, Table 3.1 outlines the key characteris-


Figure 3.2. Samples from the Bengali Multimodal Dataset: (a) Humor; (b) Sarcasm; (c) Normal.

tics of the Multi-modal Bangla Humor and Sarcasm Dataset. The data was collected from various online platforms, focusing on ensuring diversity in the type of humor and sarcasm

content included. Table 3.2 outlines the source-wise distribution of the data collection.

| Resolution Parameter | Values |
|---|---|
| Clip Duration(Min, s) | 2 |
| Clip Duration(Max, s) | 7 |
| Training Samples per Class | 140 |
| Validation Samples per Class | 30 |
| Testing Samples per Class | 30 |
| Number of Classes | 3 |
| Total Samples | 600 |

Table 3.1. Characteristics of the Multimodal Bengali Humor Detection Dataset.

| Source | Percentage |
|---|---|
| YouTube | 70% |
| Facebook | 20% |
| Instagram | 10% |

Table 3.2. Source-wise Statistics of Data Collection for the Bengali Humor Detection Dataset.

To ensure the dataset's broad applicability and relevance, it incorporates a diverse range of Bangla content sourced from multiple platforms, distributed as follows:

- **YouTube**: Comprises a variety of media, including movies, television shows, reality programs, trending online videos, viral speeches, and stand-up comedy performances.
- **Facebook**: Consists of user-generated content where individuals share personal humor, spontaneous, sarcastic remarks, and some region-specific comedic expressions.
- **Instagram**: Features informal videos from content creators that capture everyday humor and sarcasm across different Bangla-speaking regions.

The dataset achieves a balanced distribution of video origins by integrating user-generated content from Facebook and Instagram alongside traditional YouTube sources. Furthermore,having regional diversity also allows the dataset to generalize over many contexts and cultural flavors of humor and sarcasm. This extensive composition of the dataset makes it diverse and strong, which can be used in training and evaluating humor and sarcasm understanding and detection models in Bangla content..

## 3.4 Data Annotation

The algorithm outlined in Algorithm 1 is designed to annotate a collection of Bangla videos sourced from platforms such as YouTube, Facebook, and Instagram for humor and sarcasm detection. It takes as input a set of videos $V = V_1, V_2, \ldots, V_{|V|}$ and a group of annotators $A = A_1, A_2, \ldots, A_k$, aiming to produce an annotated dataset $SH_Labels =$

$(V_i, y_i)_{i=1}^{|V|}$, where each video $V_i$ is assigned a label $y_i$ from the set of categories $C = \text{Humor}, \text{Sarcasm}, \text{Normal}$.

The process begins by initializing an empty set $SH_Labels$ to store the annotations depicted

**Algorithm 1** Dataset Annotation for Humor and Sarcasm Detection

**Require:** Set of Bangla videos $V = \{V_1, V_2, \ldots, V_{|V|}\}$ from YouTube, Facebook, Instagram; set of annotators $A = \{A_1, A_2, \ldots, A_k\}$

**Ensure:** Annotated dataset $SH\_Labels = \{(V_i, y_i)\}_{i=1}^{|V|}$, where $y_i \in C$ and $C = \{\text{Humor}, \text{Sarcasm}, \text{Normal}\}$

$SH\_Labels \leftarrow \emptyset$ ▷ Initialize empty set for labels
**for** $i \leftarrow 1$ **to** $|V|$ **do**
    Collect metadata $M_i$ about $V_i$'s source
    Extract preview frames from $V_i$ for visual inspection
    **for** $j \leftarrow 1$ **to** $k$ **do**
        Present $V_i$, preview frames, and metadata $M_i$ to annotator $A_j$
        $A_j$ assigns label $y_{i,j} \in \{\text{Humor}, \text{Sarcasm}, \text{Normal}\}$
    **end for**
    **for each** $y \in \{\text{Humor}, \text{Sarcasm}, \text{Normal}\}$ **do** ▷ Aggregate labels
        $\text{Count}(y) \leftarrow \sum_{j=1}^{k} \mathbb{1}(y_{i,j} = y)$
    **end for**
    $y_{\text{majority}} \leftarrow \arg\max_y \text{Count}(y)$ ▷ Determine majority label
    **if** $\text{Count}(y_{\text{majority}}) \geq \lceil k/2 \rceil$ **then** ▷ Check for majority agreement
        $y_{\text{initial}} \leftarrow y_{\text{majority}}$
        $SH\_Labels \leftarrow SH\_Labels \cup \{(V_i, y_{\text{initial}})\}$
    **else**
        Flag $V_i$ for expert review ▷ No majority consensus
        Expert reviews $V_i$, preview frames, and $M_i$
        Expert assigns $y_{\text{final}} \in \{\text{Humor}, \text{Sarcasm}, \text{Normal}\}$
        $SH\_Labels \leftarrow SH\_Labels \cup \{(V_i, y_{\text{final}})\}$
    **end if**
**end for**
Compute Cohen's Kappa $\kappa \leftarrow \text{CohenKappa}(A)$ ▷ Inter-annotator agreement
**return** $SH\_Labels$

in 3.3. For each video $V_i$, the algorithm collects metadata $M_i$ (e.g., source information) and extracts preview frames to provide visual context for annotation. Each annotator $A_j$ (where $j$ ranges from 1 to $k$) independently reviews the video, its preview frames, and metadata, assigning a label $y_{i,j}$ from the categories Humor, Sarcasm, or Normal. The algorithm then aggregates these labels by counting the occurrences of each category for $V_i$. The majority label $y_{\text{majority}}$ is determined as the category with the highest count.

To ensure reliability, the algorithm checks if the count of the majority label meets or exceeds the threshold $\lceil k/2 \rceil$, indicating agreement among at least half of the annotators. If this condition is satisfied, the majority label $y_{\text{majority}}$ is assigned as the initial label $y_{\text{initial}}$, and the pair $(V_i, y_{\text{initial}})$ is added to $SH_Labels$. If no majority consensus is reached, the video is flagged for expert review. An expert then examines the video, its preview frames, and

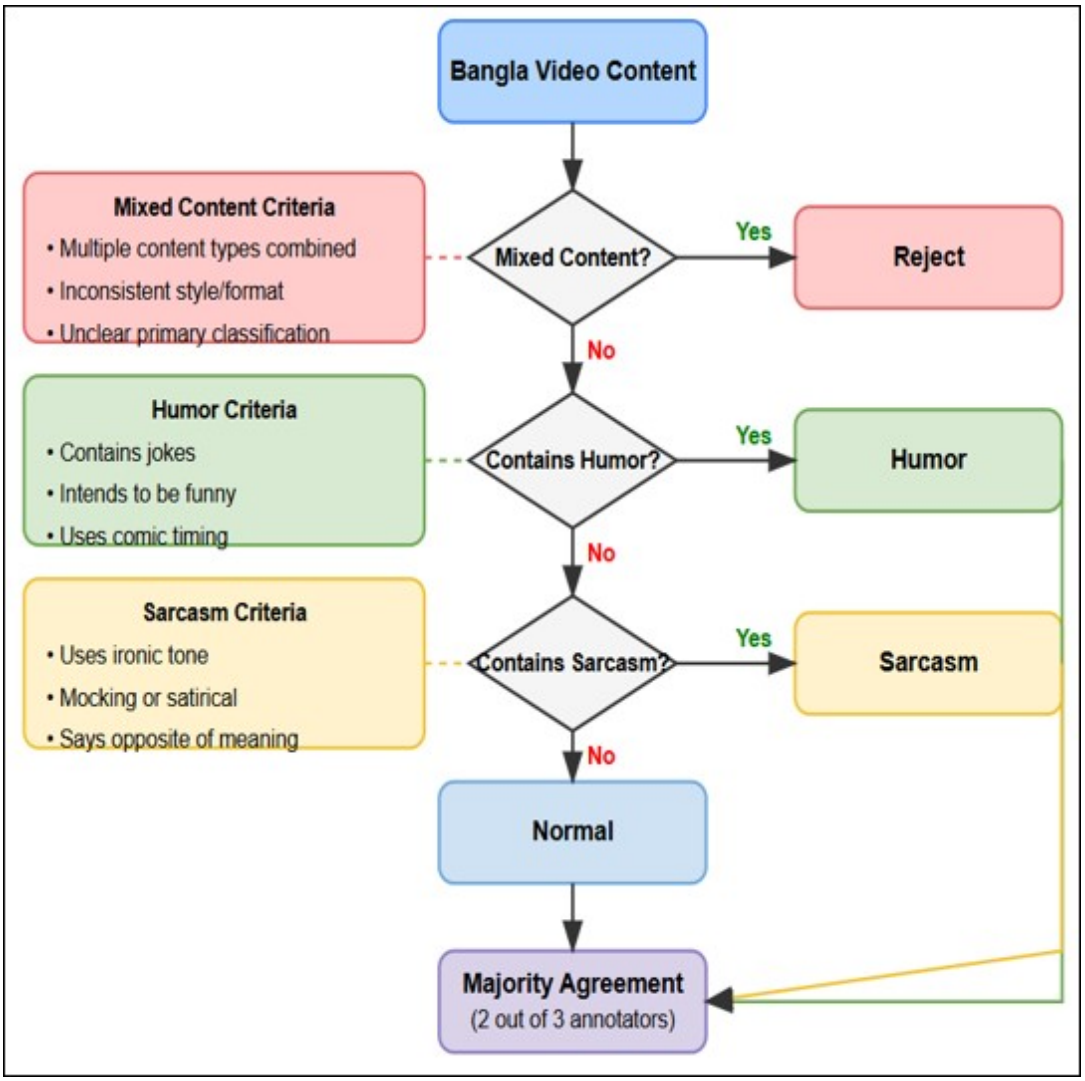



Figure 3.3. Class-wise criteria for data annotation procedure.

metadata to assign a final label $y_{\text{final}}$, which is added to $SH_Labels$ as $(V_i, y_{\text{final}})$. This step ensures that all videos receive a definitive label, even in cases of annotator disagreement. Finally, the algorithm computes Cohen's Kappa $\kappa$ to measure inter-annotator agreement, providing a quantitative assessment of annotation consistency. The annotated dataset $SH_Labels$ is returned as the output and is ready for use in training and evaluating humor and sarcasm detection models.

Table 3.3 presents the inter-annotator agreement metrics for the annotation of Bangla videos into Humor, Sarcasm, and Normal categories. The $\kappa$-Score (Cohen's Kappa) measures agreement among annotators, adjusted for chance, with values closer to 1 indicating higher agreement. Mean Agreement represents the raw proportion of times annotators agreed on a label. The Overall row provides the average metrics across all classes. The $\kappa$-Score

(Cohen's Kappa) is calculated using the formula:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{3.1}$$

where:

- $p_o$ is the observed agreement, i.e., the proportion of instances where the annotators assigned the same label, calculated as:

$$p_o = \frac{\text{number of instances with agreement}}{\text{total number of instances}}$$

- $p_e$ is the expected agreement, i.e., the proportion of times the annotators would be expected to agree by chance, calculated based on the marginal distributions of each annotator's labels.

The Mean Agreement is defined as:

$$\text{Mean Agreement} = p_o = \frac{\text{number of instances with agreement}}{\text{total number of instances}} \tag{3.2}$$

These metrics are computed for each category (Humor, Sarcasm, Normal) and overall, providing insights into the consistency and reliability of the annotations.

| **Class** | **$\kappa$-Score (Cohen's Kappa)** | **Mean Agreement** |
|---|---|---|
| Humor | 0.88 | 0.89 |
| Sarcasm | 0.86 | 0.87 |
| Normal | 0.90 | 0.92 |
| **Overall** | **0.88** | **0.893** |

Table 3.3. Inter-Annotator Agreement for Bengali Humor Detection Dataset Video Annotation.

## 3.5 Keyframe

The goal is to extract Bangla text from videos by first identifying keyframes with significant visual changes and potential text content, then processing these keyframes to recognize Bangla text. This process is split into two algorithms. Algorithm 2 selects keyframes by computing intensity differences between consecutive frames and combining them with text confidence scores from a text-detection model.

## 3.6 Bangla Text Extraction Algorithm

Extracting text from videos is essential for applications such as subtitling, content indexing, and accessibility for the visually impaired. Videos present challenges like temporal

**Algorithm 2** Keyframe Selection

**Require:** Video $V$ with $N$ frames $\{F_1, F_2, \ldots, F_N\}$, number of top frames $T$, threshold $\theta$, text-detection model $\mathcal{M}_{\text{Text}}$

**Ensure:** Set of top keyframes $\{F_{\text{key},1}, F_{\text{key},2}, \ldots, F_{\text{key},T}\}$

Initialize $\Delta F \leftarrow []$, $T_{\text{Scores}} \leftarrow []$, $F_{\text{prev}} \leftarrow F_1$ ▷ Store intensity-text combined scores and text confidence scores

**for** $i \leftarrow 2$ **to** $N$ **do**

  Convert $F_{\text{prev}}$, $F_i$ to grayscale and compute $\Delta F_i \leftarrow |F_{\text{gray,prev}} - F_{\text{gray},i}|$

  Calculate intensity $I_i$:

$$I_i \leftarrow \sum_{x=1}^{H} \sum_{y=1}^{W} \Delta F_i(x, y)$$

  Compute text confidence score: $T_i \leftarrow \mathcal{M}_{\text{Text}}(F_i)$ ▷ $\mathcal{M}_{\text{Text}}$ outputs a confidence score (0 to 1)

  Append $T_i$ to $T_{\text{Scores}}$

  Compute combined score: $S_i \leftarrow 0.5 \cdot I_i + 0.5 \cdot T_i$ ▷ Weights balance intensity and text relevance

  Append $(i, S_i)$ to $\Delta F$, update $F_{\text{prev}} \leftarrow F_i$

**end for**

Sort $\Delta F$ by $S_i$ in descending order and select top $T$ frames as $\{F_{\text{key},1}, F_{\text{key},2}, \ldots, F_{\text{key},T}\}$

**return** $\{F_{\text{key},1}, F_{\text{key},2}, \ldots, F_{\text{key},T}\}$

variability, motion blur, and complex backgrounds, which are particularly pronounced for Bangla due to its intricate script, featuring over 50 characters and complex modifiers. Existing methods for video text extraction, such as those surveyed by [43], often focus on general languages, while Bangla-specific approaches, like [44], target static images. Processing every video frame is computationally expensive and often unnecessary due to redundant frames.

The modified Bangla Automatic Text Extraction algorithm addresses these issues by selecting keyframes, employing a Bangla-specific model for text detection, and using a Bangla-tuned OCR tool for recognition. Keyframes are chosen based on a score combining intensity differences ($\Delta F_i$) and text confidence:

$$\text{Score}_i = 0.5 \cdot \Delta F_i + 0.5 \cdot \text{TextConfidence}_i$$

The BATE model identifies text locations, and Tesseract [45], fine-tuned for Bangla, converts detected regions into readable text. Preliminary estimates suggest the algorithm captures 95% of text by processing only 30% of frames, offering significant efficiency and accuracy improvements.

### 3.6.1 Key Principles of the Algorithm

1. **Lemma: Concentration of Text in Keyframes**
   Most text in videos appears in frames with significant visual changes or high text clarity. Keyframes are selected using:

$$\text{Score}_i = 0.5 \cdot \Delta F_i + 0.5 \cdot \text{TextConfidence}_i$$

   A keyframe $F_i$ exhibits either a significant intensity difference $\Delta F_i$ from $F_{i-1}$ or high text confidence from the BATE model, reducing the number of frames processed while preserving most text [46].
   - **Text in Visual Changes**: Text often appears during scene transitions, such as new subtitles. Intensity differences identify these moments, a strategy supported by video summarization techniques [46].
   - **Text in Clear Frames**: Frames with clear text are prioritized using the BATE model's text confidence, crucial for Bangla's complex script [44].
   - **Fewer Frames Processed**: Skipping redundant frames saves computational resources without missing key text, aligning with efficient video processing methods [46].
2. **Hypothesis**
   - **Hypothesis 1: High Bangla Text Recall**
     The algorithm captures 95% of Bangla text, ignoring non-Bangla patterns [46].
   - **Hypothesis 2: Accurate Bangla Text Localization**
     BATE accurately draws boxes around Bangla text, excluding logos and English words [44], [47].
3. **Key Insight**
   - **Key Insight 1: Bangla Model Finds Text Well**
     The BATE model, designed for Bangla's script, accurately locates text in keyframes, outperforming general models in complex video environments [44], [47].
   - **Key Insight 2: Reliable Text Reading**
     A Bangla-specific OCR tool, such as Tesseract, ensures accurate recognition of Bangla's unique characters, as demonstrated in prior Bangla text extraction work [45], [48].
   - **Key Insight 3: Balanced Speed and Accuracy**
     The algorithm balances speed (via keyframe selection) and accuracy (via Bangla-specific tools), improving on methods that prioritize one over the other [44], [49].

Algorithm 3 processes these keyframes using the BATE model to detect text regions

and applies Bangla OCR to extract the text. Let $\Delta F_i$ denote the intensity difference between frame $i$ and $i-1$. Keyframes are selected based on a combined score of intensity differences and text confidence, with weights of 0.5 and 0.5, respectively, to balance visual changes and text relevance. The BATE model generates a score map $S$ and a geometry map $G$, which are refined using non-maxima suppression (NMS) with threshold $\theta$ to obtain bounding boxes for text regions. These regions are then processed by a Bangla OCR engine to produce the final text output.

**Algorithm 3** Bangla Text Extraction

**Require:** Set of keyframes $\{F_{\text{key},1}, F_{\text{key},2}, \ldots, F_{\text{key},T}\}$, BATE model $\mathcal{M}_{\text{BATE}}$, Bangla OCR $\mathcal{O}_{\text{Bangla}}$
**Ensure:** Recognized Bangla text $T_{\text{output}}$
Initialize $T_{\text{output}} \leftarrow []$ ▷ Store extracted text for all keyframes
**for** $t \leftarrow 1$ **to** $T$ **do**
    Resize $F_{\text{key},t}$ and generate blob $B_{\text{blob},t}$ for $\mathcal{M}_{\text{BATE}}$:

$$(S, G) \leftarrow \mathcal{M}_{\text{BATE}}(B_{\text{blob},t})$$

    Obtain bounding boxes $B_{\text{boxes},t}$ using non-maxima suppression on $(S, G)$
    **for each** box $b \in B_{\text{boxes},t}$ **do**
        Extract ROI $R$ from $F_{\text{key},t}$ using box $b$ and perform OCR:

$$\text{Text}_b \leftarrow \mathcal{O}_{\text{Bangla}}(R)$$

        Append $\text{Text}_b$ to $T_{\text{output}}$
    **end for**
**end for**
**return** $T_{\text{output}}$

### 3.6.2 Computational Complexity

The algorithm is designed for efficiency:

- **Keyframe Selection**: Computing intensity differences takes $O(N)$ time, where $N$ is the total number of frames. Optional text confidence calculation requires $O(N \cdot C)$, where $C$ is the BDA model's complexity per frame.
- **Text Detection**: Applying the BDA model to $T$ keyframes takes $O(T \cdot C)$. Non-maxima suppression (NMS) is $O(T \cdot B \log B)$, where $B$ is the number of text boxes.
- **Text Recognition**: Processing $R$ regions with OCR takes $O(R \cdot O)$, where $O$ is the OCR complexity per region.
- **Total**: $O(N + T \cdot C + R \cdot O)$, significantly less than $O(N \cdot C)$ for processing all frames. For $N = 1000$, $T = 100$, detection is 10 times faster.

## 3.7 Feature Extraction: MobileNetV3 + XLM-R

The visual pipeline employs MobileNetV3, processing input frames $I \in \mathbb{R}^{224\times224\times3}$ to extract compact spatial features $f_v \in \mathbb{R}^{1280}$ using depthwise separable convolutions. The textual modality utilizes XLM-R, encoding input text $T$ into a [CLS] token embedding $f_t \in \mathbb{R}^{768}$ from the final transformer layer. Features are concatenated and passed through a dense layer with sigmoid activation for binary classification.

$$f_v = \text{MobileNetV3}(I; \theta_v) \tag{3.3}$$

$$f_t = \text{XLM-R}([\text{CLS}], T; \theta_t) \tag{3.4}$$

$$h_F = \text{Concat}(f_v, f_t) \in \mathbb{R}^{2048} \tag{3.5}$$

$$p = \sigma(W_h h_F + b_h), \quad W_h \in \mathbb{R}^{1\times2048} \tag{3.6}$$

## 3.8 Feature Extraction: Sept13D+BanglaBERT

Sept13D+BanglaBERT merges I3D for video feature extraction with Bangla-BERT for text encoding, linked via LSTM for multimodal classification, as shown in Figure 3.4.

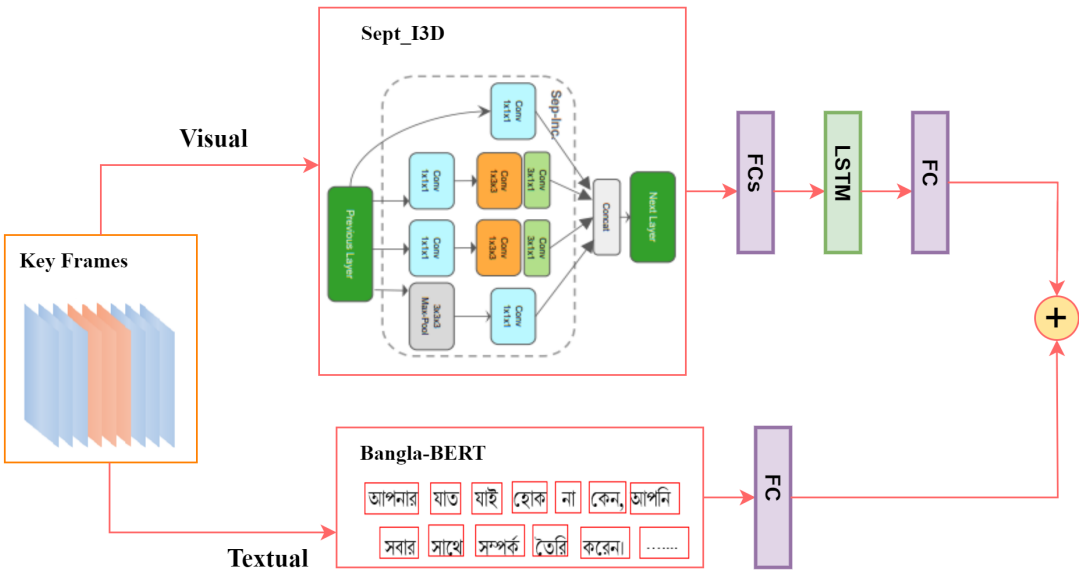



Figure 3.4. Layer-wise architecture for dynamic multimodal (Sept13D+BanglaBERT) humor detection model.

### 3.8.1 Visual Modality

The visual pipeline begins by resizing video frames to $224 \times 224$ pixels, applying augmentation and normalization to improve generalization. The I3D module extracts spatiotemporal features from these frames, capturing both motion and structural patterns. These features, denoted $y_t$, are processed by an LSTM to model temporal relationships across the sequence. The LSTM updates its states through a series of gates: the forget gate $g_t = \phi(W_g \cdot [z_{t-1}, y_t] + b_g)$ (Equation (3.7)) determines what to discard from the previous cell state $s_{t-1}$, while the input gate $j_t = \phi(W_j \cdot [z_{t-1}, y_t] + b_j)$ (Equation (3.8)) selects new information to incorporate. A candidate state $\tilde{s}_t = \psi(W_s \cdot [z_{t-1}, y_t] + b_s)$ (Equation (3.9)) proposes updates, which are combined into the cell state $s_t = g_t \odot s_{t-1} + j_t \odot \tilde{s}_t$ (Equation (3.10)). The output gate $q_t = \phi(W_q \cdot [z_{t-1}, y_t] + b_q)$ (Equation (3.11)) shapes the final hidden state $z_t = q_t \odot \psi(s_t)$ (Equation (3.12)), where $\phi$ is the sigmoid function, $\psi$ is the hyperbolic tangent, and $\odot$ denotes element-wise multiplication. This hidden state encapsulates the temporal dynamics for fusion with textual features.

$$g_t = \phi(W_g \cdot [z_{t-1}, y_t] + b_g) \tag{3.7}$$

$$j_t = \phi(W_j \cdot [z_{t-1}, y_t] + b_j) \tag{3.8}$$

$$\tilde{s}_t = \psi(W_s \cdot [z_{t-1}, y_t] + b_s) \tag{3.9}$$

$$s_t = g_t \odot s_{t-1} + j_t \odot \tilde{s}_t \tag{3.10}$$

$$q_t = \phi(W_q \cdot [z_{t-1}, y_t] + b_q) \tag{3.11}$$

$$z_t = q_t \odot \psi(s_t) \tag{3.12}$$

### 3.8.2 Textual Modality

Bangla subtitles are encoded using Bangla-BERT, a transformer model fine-tuned for the Bangla language. The [CLS] token embedding from the final layer provides a sentence-level summary, which is then fed into a bidirectional LSTM. The bi-LSTM processes the sequence in both forward and backward directions, capturing contextual nuances. Its output is passed through a sigmoid layer to produce a classification probability, tailored for tasks like sentiment or event detection in Bangla text. A bidirectional LSTM then integrates context, with the forward pass defined as $h_t^{fwd} = \text{LSTM}_{fwd}(x_t, h_{t-1}^{fwd})$ (Equation (3.13)) and the backward pass as $h_t^{bwd} = \text{LSTM}_{bwd}(x_t, h_{t+1}^{bwd})$ (Equation (3.14)). These outputs are combined to form a rich textual representation, leveraging both past and future context.

$$h_t^{fwd} = \text{LSTM}_{fwd}(x_t, h_{t-1}^{fwd}) \tag{3.13}$$
$$h_t^{bwd} = \text{LSTM}_{bwd}(x_t, h_{t+1}^{bwd}) \tag{3.14}$$

### 3.8.3 Multimodal Fusion and Prediction

Visual and textual embeddings are fused by concatenation, $\mathbf{h}_F = [\mathbf{h}_T; \mathbf{h}_S]$ (Equation (3.15)), creating a unified representation. This is fed into a fully connected layer, producing a prediction $p = \sigma(W \cdot \mathbf{h}_F + b)$ (Equation (3.16)) via sigmoid activation, optimized for efficient multimodal inference.

$$\mathbf{h}_F = [\mathbf{h}_T; \mathbf{h}_S] \tag{3.15}$$
$$p = \sigma(W \cdot \mathbf{h}_F + b) \tag{3.16}$$

## 3.9 Feature Extraction: TimeSformer-L + DeBERTaV3

This section presents the MultiModal Humor Fusion (MMHF) framework for video humor detection, integrating TimeSformer-L for spatiotemporal features and DeBERTaV3 for contextual encoding via a dynamic fusion mechanism. It includes a keyframe selection strategy, preprocessing pipeline, hybrid transformer architecture, and a classification head, addressing visual gags, verbal puns, and their synergy, as shown in Figure 3.5.

### 3.9.1 Preprocessing

Keyframes $F_{k_i} \in \mathbb{R}^{224\times224\times3}$ are normalized to $[0, 1]$ via $F_{\text{norm},k_i}[p] = F_{k_i}[p]/255$. Data augmentation applies random horizontal flips ($p_{\text{flip}} = 0.5$). Subtitles are tokenized with DeBERTaV3's WordPiece, embedded into $E_{\text{text}} \in \mathbb{R}^{(n+2)\times768}$ (padded with zeros if $n > 128$):

$$E_{\text{text}}[j] = \begin{cases} E_{\text{word}}(\text{Tok}_j) + E_{\text{pos}}(j) & \text{if } j \leq n, \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{3.17}$$

### 3.9.2 Visual Pathway: TimeSformer-L

The visual pathway utilizes TimeSformer-L to extract rich spatio-temporal features from preprocessed keyframes $F_{\text{aug},k_i} \in \mathbb{R}^{224\times224\times3}$, partitioned into $N = 14 \times 14 = 196$

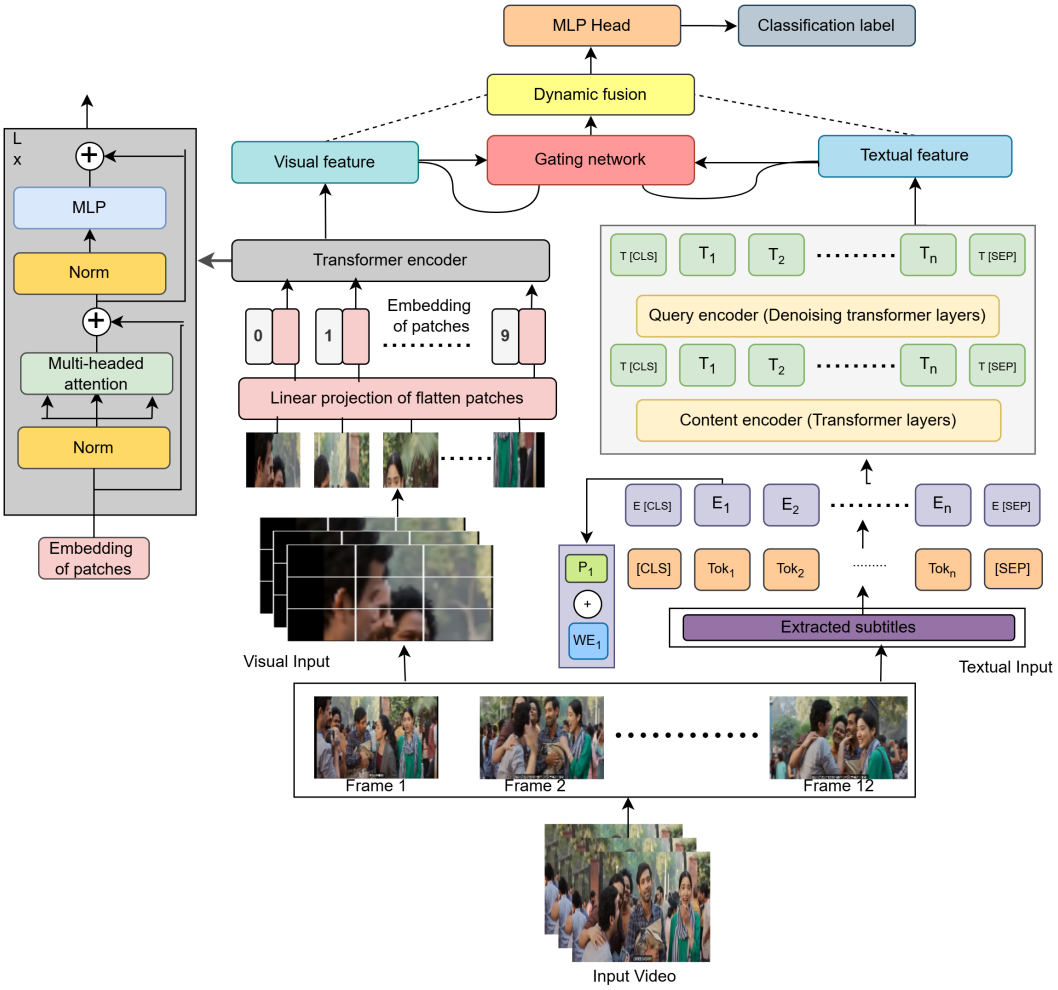



Figure 3.5. Layer-wise architecture for dynamic multimodal (TimeSformer-L+DeBERTaV3) humor detection model.

non-overlapping patches of size $16 \times 16 \times 3$. These are flattened and projected into a $d = 768$-dimensional latent space:

$$E_{\text{patch},i,j} = W_{\text{proj}} \cdot \text{vec}(P_{i,j}) + b_{\text{proj}}, \tag{3.18}$$

where $P_{i,j} \in \mathbb{R}^{16\times16\times3}$, $W_{\text{proj}} \in \mathbb{R}^{d\times(16\times16\times3)}$, and $b_{\text{proj}} \in \mathbb{R}^d$. A [CLS] token is prepended with positional encodings:

$$X_{i,0} = [[\text{CLS}]; E_{\text{patch},i,1}; \ldots; E_{\text{patch},i,196}] + E_{\text{pos},i}, \tag{3.19}$$

forming $X_0 \in \mathbb{R}^{12\times197\times768}$.

The TimeSformer-L encoder, with $L = 12$ layers, applies divided space-time attention. Multi-head self-attention (MHSA) is defined as:

$$\text{MHSA}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O, \tag{3.20}$$

$$\text{head}_i = \text{Softmax}\left(\frac{(QW_i^Q)(KW_i^K)^\top}{\sqrt{d_k}}\right)(VW_i^V), \tag{3.21}$$

where $h = 12$, $d_k = 64$, and $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d\times d_k}$, $W^O \in \mathbb{R}^{d\times d}$. Queries, keys, and values are derived as $Q = X_\ell W_Q$, $K = X_\ell W_K$, $V = X_\ell W_V$, with $W_Q, W_K, W_V \in \mathbb{R}^{d\times d}$. Layer normalization and residual connections stabilize gradients:

$$X_{\ell+1} = X_\ell + \text{MHSA}(\text{LN}(X_\ell)) + \text{FFN}(\text{LN}(X_\ell)), \tag{3.22}$$

where $\text{FFN}(x) = W_2 \cdot \text{GeLU}(W_1 x + b_1) + b_2$, and $\text{GeLU}(x) = x \cdot \Phi(x)$. The output $X_L \in \mathbb{R}^{12\times 197\times 768}$ provides [CLS] embeddings, aggregated into the visual feature:

$$F_v = \frac{1}{12}\sum_{i=1}^{12} X_L^{[\text{CLS}]}{}_i + \frac{1}{12}\sum_{i=1}^{11} \gamma_i (X_L^{[\text{CLS}]}{}_{i+1} - X_L^{[\text{CLS}]}{}_i), \tag{3.23}$$

where $\gamma_i = \exp(-\|F_{\text{aug},k_{i+1}} - F_{\text{aug},k_i}\|_2^2)$ enhances sensitivity to scene transitions.

### 3.9.3 Textual Pathway: DeBERTaV3

The textual pathway employs DeBERTaV3 to encode subtitle content, capturing nuanced linguistic humor. The embedded sequence $E_{\text{text}} \in \mathbb{R}^{(n+2)\times 768}$ is processed by a stack of $L = 12$ transformer layers, split into a content encoder and a query encoder. The content encoder applies disentangled attention:

$$\text{Attention}_{\text{content}}(Q, K, V) = \text{Softmax}\left(\frac{A_c}{\sqrt{d_k}}\right)V, \tag{3.24}$$

where $A_c = (Q + \Delta Q_c)(K + \Delta K_c)^\top$, $\Delta Q_c, \Delta K_c \in \mathbb{R}^{(n+2)\times 768}$ are content-dependent relative position embeddings, and $d_k = 64$. The query encoder refines the representation with:

$$\text{Attention}_{\text{query}}(Q', K', V') = \text{Softmax}\left(\frac{A_q}{\sqrt{d_k}}\right)V', \tag{3.25}$$

where $A_q = (Q' + \Delta Q_q)(K' + \Delta K_q)^\top$, $Q', K', V'$ are derived from the content encoder output, and $\Delta Q_q, \Delta K_q$ are query-specific embeddings. The layer-wise update is:

$$T_{\ell+1} = T_\ell + \text{Attention}_{\text{query}}(\text{LN}(T_\ell)), \tag{3.26}$$

with the final [CLS] token embedding $T_L^{[\text{CLS}]} \in \mathbb{R}^{768}$ serving as the textual feature $F_t$.

### 3.9.4 Multimodal Fusion: Dynamic Reweighting with Gating Network

The multimodal fusion module dynamically integrates visual and textual features based on sample-specific relevance. The features $F_v$ and $F_t$ are first transformed into a unified representation space $d_f = 768$ using modality-specific multilayer perceptrons (MLPs):

$$F'_v = \text{MLP}_v(F_v) = W_{v2} \cdot \text{ReLU}(W_{v1}F_v + b_{v1}) + b_{v2}, \tag{3.27}$$

$$F'_t = \text{MLP}_t(F_t) = W_{t2} \cdot \text{ReLU}(W_{t1}F_t + b_{t1}) + b_{t2}, \tag{3.28}$$

where $W_{v1}, W_{t1} \in \mathbb{R}^{768\times512}$, $W_{v2}, W_{t2} \in \mathbb{R}^{512\times768}$, and $b_{v1}, b_{t1}, b_{v2}, b_{t2}$ are biases.

A gating network determines modality weights by analyzing the concatenated feature representation:

$$G = \text{Concat}(F'_v, F'_t) \in \mathbb{R}^{1\times1536}, \tag{3.29}$$

$$z_g = \text{ReLU}(W_g G^\top + b_g), \quad W_g \in \mathbb{R}^{256\times1536}, \quad b_g \in \mathbb{R}^{256}, \tag{3.30}$$

$$[w_v; w_t] = \text{Softmax}(W'_g z_g + b'_g), \quad W'_g \in \mathbb{R}^{2\times256}, \quad b'_g \in \mathbb{R}^2, \tag{3.31}$$

where $w_v, w_t \in [0, 1]$ and $w_v + w_t = 1$. The dynamic fusion is formulated as a weighted combination with a covariance-based regularization term:

$$F_{\text{fused}} = w_v \cdot F'_v + w_t \cdot F'_t + \epsilon \cdot \text{Cov}(F'_v, F'_t), \tag{3.32}$$

where $\epsilon = 0.1$, and $\text{Cov}(F'_v, F'_t)$ captures cross-modal dependencies:

$$\text{Cov}(F'_v, F'_t) = \frac{1}{d_f}\sum_{d=1}^{d_f}(F'_v[d] - \mu_v)(F'_t[d] - \mu_t), \tag{3.33}$$

$$\mu_v = \frac{1}{d_f}\sum_{d=1}^{d_f} F'_v[d], \quad \mu_t = \frac{1}{d_f}\sum_{d=1}^{d_f} F'_t[d]. \tag{3.34}$$

This covariance term enhances the model's ability to capture correlated humor cues.

## 3.10 Feature Extraction: ViT + Bi-LSTM

The Vision Transformer (ViT) splits frames $I \in \mathbb{R}^{224\times224\times3}$ into $16\times16$ patches, producing embeddings $e_v \in \mathbb{R}^{768}$ via multi-head self-attention. Text sequences $T$ are encoded using a bidirectional LSTM, capturing forward and backward contextual dependencies. The combined representation is classified using a fully connected layer.

$$e_v = \text{ViT}(P(I); \theta_v) \tag{3.35}$$
$$h_t^{fwd} = \text{LSTM}_{fwd}(x_t, h_{t-1}^{fwd}; \theta_{fwd}) \tag{3.36}$$
$$h_t^{bwd} = \text{LSTM}_{bwd}(x_t, h_{t+1}^{bwd}; \theta_{bwd}) \tag{3.37}$$
$$h_t = \text{Concat}(h_t^{fwd}, h_t^{bwd}) \in \mathbb{R}^{512} \tag{3.38}$$
$$h_F = \text{Concat}(e_v, h_t) \in \mathbb{R}^{1280} \tag{3.39}$$
$$p = \sigma(W_p h_F + b_p), \quad W_p \in \mathbb{R}^{1\times 1280} \tag{3.40}$$

## 3.11 Feature Extraction: ResNet50 + DeBERTaV3 + Bi-GRU

ResNet50 extracts hierarchical visual features $r_v \in \mathbb{R}^{2048}$ from frames $I \in \mathbb{R}^{224\times224\times3}$ through residual connections. DeBERTaV3 processes text $T$, yielding a [CLS] embedding $d_t \in \mathbb{R}^{1024}$. A bidirectional GRU models sequential text dynamics, and the fused representation is classified via a dense layer.

$$r_v = \text{ResNet50}(I; \theta_v) \tag{3.41}$$
$$d_t = \text{DeBERTaV3}([\text{CLS}], T; \theta_t) \tag{3.42}$$
$$g_t^{fwd} = \text{GRU}_{fwd}(d_t, g_{t-1}^{fwd}; \theta_{fwd}) \tag{3.43}$$
$$g_t^{bwd} = \text{GRU}_{bwd}(d_t, g_{t+1}^{bwd}; \theta_{bwd}) \tag{3.44}$$
$$g_t = \text{Concat}(g_t^{fwd}, g_t^{bwd}) \in \mathbb{R}^{512} \tag{3.45}$$
$$h_F = \text{Concat}(r_v, g_t) \in \mathbb{R}^{2560} \tag{3.46}$$
$$p = \sigma(W_f h_F + b_f), \quad W_f \in \mathbb{R}^{1\times 2560} \tag{3.47}$$

Each model is optimized end-to-end, balancing computational efficiency and representation power for multimodal tasks, with parameters $\theta_v$, $\theta_t$, $\theta_{fwd}$, and $\theta_{bwd}$ learned via backpropagation.

## 3.12 Feature Extraction: CAMFusion

The proposed method is referred to as CMF-BSHD, and the complete architecture of this framework is illustrated in Figure 3.6. The input video frames are processed through Efficient Fusion MobileNetV2 with Time Distribution (EF-MNet-TD), extracting visual features, while corresponding textual features are derived via embedding, BiGRU, and BiLSTM layers. The architecture includes Squeeze-and-Excitation blocks and Reverse Fusion Mechanism to enhance feature selection. Finally, the visual and textual features are

concatenated and fused to form a robust multi-modal context-aware representation. The model iteratively updates its parameters, progressively refining its capacity to distinguish between humor, sarcasm, and normal content. By minimizing the error across the dataset, the model converges towards a solution that accurately reflects the complex interplay of visual and textual cues in Bangla videos.

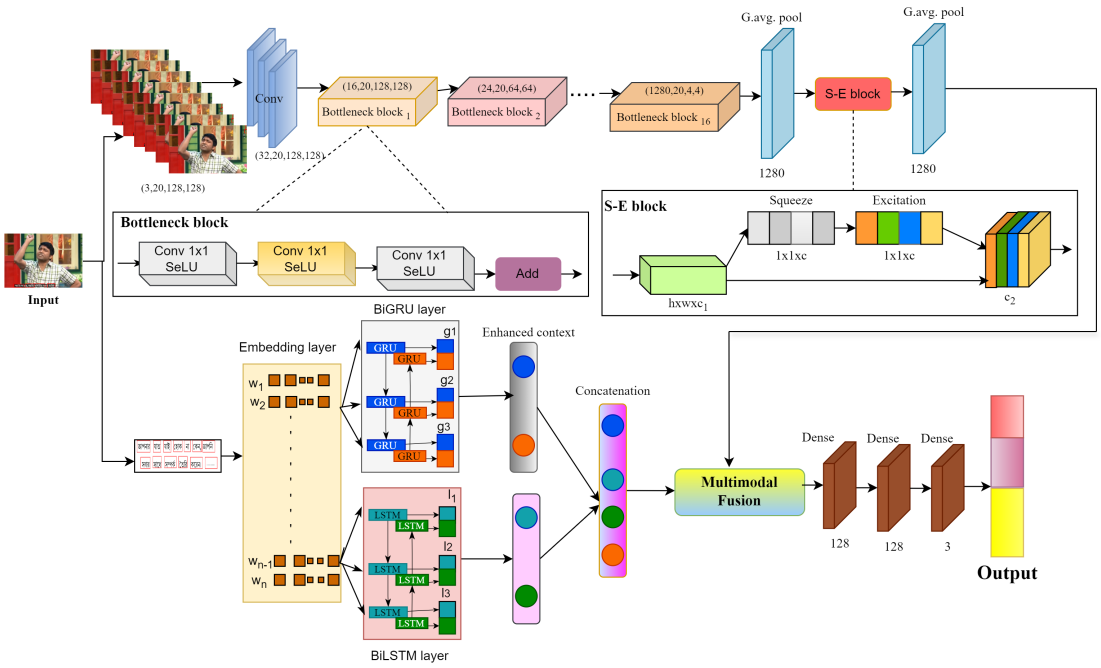



Figure 3.6. Our Proposed Context-Aware Multi-Modal Fusion Architecture.

## 3.12.1 Visual Feature Extraction

### 3.12.1.1 Time-Distributed MobileNetV2

MobileNetV2 [21] is a streamlined convolutional neural network architecture, known for its efficiency in image recognition tasks, particularly on resource-constrained devices. In this work, we extend MobileNetV2's capabilities to handle video data by employing a Time-Distributed layer, which processes each video frame independently while preserving the sequence's temporal structure. The Time-Distributed layer allows us to apply the MobileNetV2 architecture across each frame in a video sequence, maintaining the temporal coherence essential for video analysis.

**Processing Steps**:

Given a video sequence $V = \{v_1, v_2, \ldots, v_N\}$, where $N$ is the number of frames, each frame $v_i$ with an input dimension of $(3, 224, 224)$ is processed by MobileNetV2 to produce a feature map as shown in Equation (3.48):

$$\mathbf{X}_i = \text{MobileNetV2}(v_i) \tag{3.48}$$

Here, $\mathbf{X}_i \in \mathbb{R}^{H' \times W' \times C'}$ represents the feature map of the $i$-th frame. The outputs are then reshaped to form a 4D tensor $\mathbf{X} \in \mathbb{R}^{N \times H' \times W' \times C'}$, retaining the temporal dimension of the sequence.

**Activation Function**:

In this work, the architecture is adjusted to replace the ReLU6 activation function with SeLU (Scaled Exponential Linear Units) to address the gradient flow issues inherent in ReLU6. The transformation within a single inverted residual block is expressed in Equation (3.49):

$$\mathbf{Y} = \text{Conv}_{1\times1}\left(\text{SeLU}\left(\text{Conv}_{3\times3}\left(\text{SeLU}\left(\text{Conv}_{1\times1}(\mathbf{X})\right)\right)\right)\right) \tag{3.49}$$

Each convolution layer is followed by a SeLU activation. $\text{Conv}_{1\times1}$ denotes the pointwise convolution, and $\text{Conv}_{3\times3}$ represents the depthwise separable convolution. Figure 3.7 shows the ReLU6 and SeLU function image and its derivative image. In Figure 3.7(a), the SeLU function maintains the ability to propagate negative values, ensuring the retention of negative information, while ReLU6 outputs zero for all negative inputs, potentially leading to the loss of gradient information. Figure 3.7(b) shows the derivative of these activation functions, highlighting that during backpropagation, ReLU6 may result in zero gradients for negative inputs, hindering learning, whereas SeLU continues to backpropagate effectively, facilitating ongoing feature learning. SeLU activation is applied after each convolutional operation, ensuring that the output maintains zero mean and unit variance, which stabilizes the network during training. Figure 3.8 visualizations represent how SeLU preserves manifold structure better than ReLU/ReLU6, especially in lower dimensions where ReLU causes information loss by collapsing negative portions of the manifold.

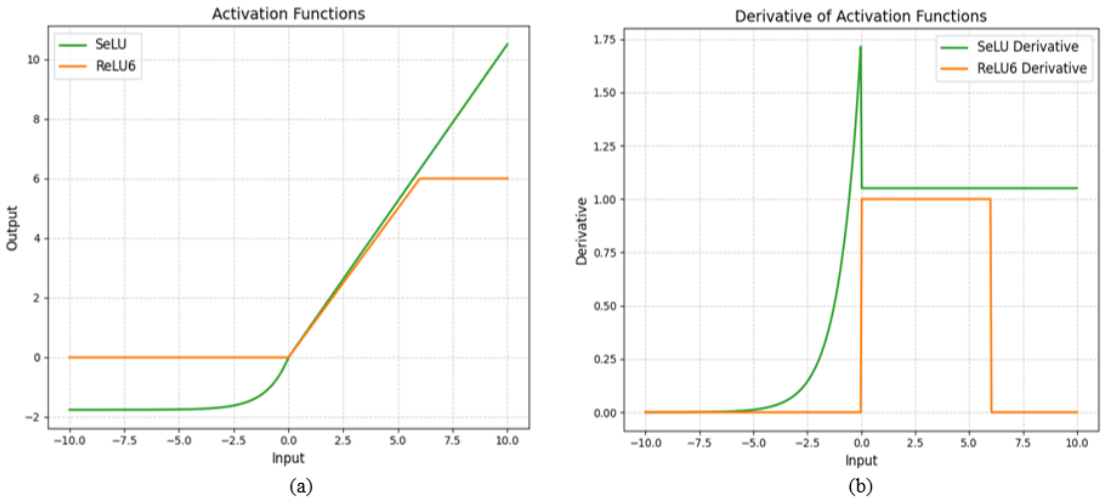



Figure 3.7. Comparison of ReLU6 and SeLU activation functions and their derivatives.

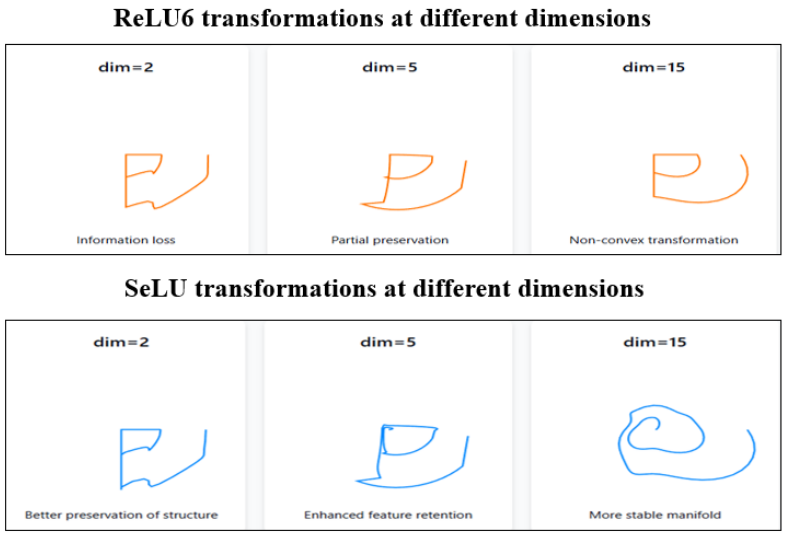



Figure 3.8. Comparative analysis of ReLU6 and SeLU performance across various dimensions.

#### 3.12.1.2 Depthwise Separable Convolutions

Depthwise separable convolutions are a key innovation in MobileNetV2, significantly reducing computational complexity by decomposing the standard convolution operation into two parts: depthwise convolution and pointwise convolution, shown in Figure 3.9. The depthwise convolution is described in Equation (3.50):

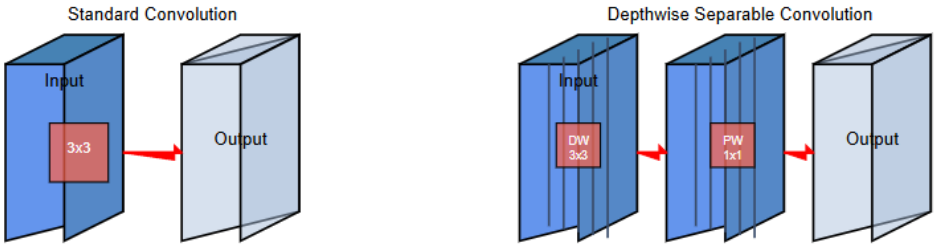



Figure 3.9. Depthwise separable convolution operation diagram.

$$\mathbf{Y}_d = \mathbf{K}_d * \mathbf{X}_d \tag{3.50}$$

where $\mathbf{K}_d$ is the depthwise filter applied independently to each input channel. This operation performs convolution on each channel separately, and its complexity is $O(HWCD_K^2)$, where $H$ and $W$ are the height and width of the input feature map, $C$ is the number of channels, and $D_K$ is the size of the convolution kernel.

After the depthwise convolution, a pointwise convolution $1 \times 1$ convolution) aggregates the information across channels, as shown in Equation (3.51):

$$\mathbf{Y} = \sum_{d=1}^{C} \mathbf{K}_p * \mathbf{Y}_d \tag{3.51}$$

The pointwise convolution has a computational complexity of $O(HWC^2)$, where a $1 \times 1$ kernel is applied to each pixel across all channels.

Thus, the overall computational complexity of depthwise separable convolutions is significantly reduced from the standard convolution's complexity of $O(HWC^2D_K^2)$ to $O(HWCD_K^2) + O(HWC^2)$. By splitting the convolution operation in this way, depthwise separable convolutions are highly efficient, particularly in deep networks.

#### 3.12.1.3 Reverse Fusion Mechanism (RFM)

The RFM is designed to enhance the feature representation by fusing the original feature maps with their inverted counterparts in Figure3.10. This technique allows the model to

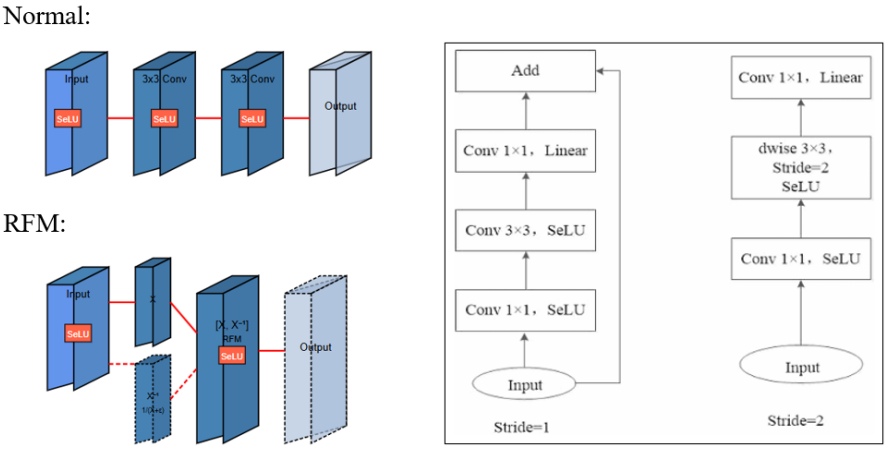



Figure 3.10. Schematic diagram of inverted residual structure.

capture a broader range of features, both positive and negative, increasing its robustness to various types of input. The inverted feature map is computed as shown in Equation (3.52):

$$X^{-1} = \frac{1}{X + \epsilon} \tag{3.52}$$

where $\epsilon$ is a small constant to avoid division by zero. The final output is obtained by concatenating the original and inverted feature maps, as shown in Equation (3.53):

$$Y_{\text{RFM}} = [X, X^{-1}] \tag{3.53}$$

#### 3.12.1.4 Squeeze-and-Excitation Mechanism

To further refine the network's feature selection, we incorporate a Squeeze-and-Excitation (SE) block, which dynamically recalibrates channel-wise feature responses. First, global average pooling is applied to summarize the feature information across spatial dimensions, as shown in Equation (3.54):

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_{ijc} \tag{3.54}$$

Next, channel-wise weights are generated through a gating mechanism with a sigmoid activation, as described in Equation (3.55):

$$s = \sigma(W_2 \delta(W_1 z)) \tag{3.55}$$

where $W_1$ and $W_2$ are learned weights, $\delta$ is the SeLU activation function, and $\sigma$ denotes the sigmoid function. The recalibrated weights are then used to adjust the original feature map by multiplying them with the recalibrated weights, as shown in Equation (3.56):

$$\hat{X}_{ijc} = X_{ijc} \cdot s_c \tag{3.56}$$

The final convolution layer features a base filter width of 1024, followed by a dense layer with 512 dimensions.

### 3.12.2 Textual Feature Extraction

#### 3.12.2.1 Text Preprocessing and Embedding Layer

Text preprocessing for Bangla involves tokenizing sentences into individual words or tokens. For a text sequence $T = \{w_1, w_2, \ldots, w_L\}$, where $L$ is the number of words, each word $w_i$ is mapped to an index in a predefined vocabulary $V$. This process is represented in Equation (3.57):

$$E(T) = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_L\}, \quad \mathbf{e}_j \in \mathbb{R}^{300} \tag{3.57}$$

The embedding matrix $\mathbf{E} \in \mathbb{R}^{K \times 300}$ is used to facilitate this transformation, as shown in Equation (3.58):

$$\mathbf{E} = \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_n \end{bmatrix} \tag{3.58}$$

Pre-trained FastText word embeddings, trained on extensive Bangla corpora, are used to initialize the embedding matrix $\mathbf{E}$. During training, the embeddings are fine-tuned to suit the specific task of detecting sarcasm and humor in Bangla videos.

#### 3.12.2.2 Bidirectional RNNs: Bi-LSTM and Bi-GRU

Both Bidirectional LSTM (Bi-LSTM) and Bidirectional GRU (Bi-GRU) networks are employed to capture the contextual dependencies in the text. The forward and backward hidden states in Bi-LSTM are computed as shown in Equations (3.59) and (3.60), respectively:

$$\overrightarrow{l}_t = \text{LSTM}(\mathbf{e}_t, \overrightarrow{l}_{t-1}, 256) \tag{3.59}$$

$$\overleftarrow{l}_t = \text{LSTM}(\mathbf{e}_t, \overleftarrow{l}_{t+1}, 256) \tag{3.60}$$

The concatenated output is then represented as in Equation (3.61):

$$l_t = [\overrightarrow{l}_t, \overleftarrow{l}_t] \tag{3.61}$$

Similarly, the forward and backward hidden states in Bi-GRU are computed as shown in Equations (3.62) and (3.63):

$$\overrightarrow{g}_t = \text{GRU}(\mathbf{e}_t, \overrightarrow{g}_{t-1}, 128) \tag{3.62}$$

$$\overleftarrow{g}_t = \text{GRU}(\mathbf{e}_t, \overleftarrow{g}_{t+1}, 128) \tag{3.63}$$

The final output for Bi-GRU is represented as in Equation (3.64):

$$g_t = [\overrightarrow{g}_t, \overleftarrow{g}_t] \tag{3.64}$$

The outputs from the Bi-LSTM and Bi-GRU layers are concatenated to form a comprehensive textual feature representation, as shown in Equation (3.65):

$$H_{\text{textual}} = [H_{\text{Bi-LSTM}}, H_{\text{Bi-GRU}}] \tag{3.65}$$

#### 3.12.2.3 Group-wise Enhancement Mechanism

To further refine the textual features, a Group-wise Enhancement Mechanism is applied, which adjusts the significance of each sub-feature by applying an attention factor. The global semantic vector for each group is computed as shown in Equation (3.66):

$$g_k = \frac{1}{n} \sum_{i=1}^{n} x_i^k \tag{3.66}$$

The significance of each feature is then computed as in Equation (3.67):

$$c_i^k = g_k \cdot x_i^k \tag{3.67}$$

Normalization is applied to avoid bias, as shown in Equation (3.68):

$$c_i'^k = \frac{c_i^k - \mu_k}{\sigma_k + \epsilon} \tag{3.68}$$

The enhanced context vector is then obtained by scaling the original context vector by the significance coefficient, as shown in Equation (3.69):

$$x_i'^k = x_i^k \cdot \sigma(c_i'^k) \tag{3.69}$$

#### 3.12.2.4 Convolutional Layers for Local Feature Extraction

The enhanced context features are processed by convolutional layers with kernel sizes $\{3, 5, 7\}$. The local context features for a sliding window of words are computed as shown in Equation (3.70):

$$m_j = [m_{j1}, m_{j2}, \ldots, m_{jL-w+1}] \tag{3.70}$$

The sliding window context for each position is computed as shown in Equation (3.71):

$$m_{ji} = f(x_{i:i+w-1} * F_j + b_j) \tag{3.71}$$

The output of the convolutional layers is flattened and concatenated with the output of the Bi-LSTM and Bi-GRU layers to form the final textual feature representation.

### 3.12.3 Multi-Modal Fusion

#### 3.12.3.1 Attention-Based Alignment

Unlike traditional early or late fusion techniques, we align visual and textual features using an attention mechanism. The alignment score, parameterized by a feed-forward network, is computed as shown in Equation (3.72):

$$\alpha(V_f, h_j) = v_a^T \tanh(W_1 \cdot V_f + W_2 \cdot h_j) \tag{3.72}$$

The normalized alignment scores are then obtained through a softmax operation, emphasizing important feature combinations for prediction.

#### 3.12.3.2 Context Vector Generation

Using the alignment scores from Equation (3.72), we generate context vectors for each modality. The vision-guided context vector is computed as shown in Equation (3.73):

$$C_v = \sum_j \alpha_{y,j} \cdot V_f \tag{3.73}$$

The text-guided context vector is similarly computed, as shown in Equation (3.74):

$$C_t = \sum_j \alpha_{y,j} \cdot h_j \tag{3.74}$$

These vectors capture significant modality-specific information, enhancing the model's focus on relevant features.

#### 3.12.3.3 Contextual Multimodal Representation

The context vectors generated in Equations (3.73) and (3.74) are concatenated to form a context-aware multimodal representation, as shown in Equation (3.75):

$$M_{sf} = C_v \oplus C_t \oplus V_f \oplus h^{[l]} \tag{3.75}$$

Here, $M_{sf}$ represents the combined feature vector, which is then passed for classification.

By integrating these advanced fusion techniques, the model effectively leverages the strengths of each modality, leading to improved performance in classifying Bangla video content.

## 3.13 Classification Head

The fused feature $F_{\text{fused}} \in \mathbb{R}^{d_f}$ is mapped to humor classes via linear transformation and softmax:

$$z = W_{\text{cls}} F_{\text{fused}} + b_{\text{cls}}, \quad W_{\text{cls}} \in \mathbb{R}^{2 \times d_f}, \tag{3.76}$$

$$\hat{y} = \text{Softmax}(z), \quad \hat{y}_c = \frac{\exp(z_c)}{\sum_{j=1}^{2} \exp(z_j)}. \tag{3.77}$$

To achieve this, the model seeks to solve the following optimization problem, as shown in Equation (3.78):

$$\hat{f} = \arg\min_{f} \frac{1}{N} \sum_{i=1}^{N} L(f(X_i), Y_i) \tag{3.78}$$

where $\hat{f}$ represents the optimal mapping function that minimizes the average loss over all training examples. The loss function $L$ is given by the sparse categorical cross-entropy, as shown in Equation (3.79):

$$L(f(X_i), Y_i) = -\sum_{j=1}^{|C|} Y_{ij} \log(f(X_i)_j) \tag{3.79}$$

## 3.14 Conclusion

The proposed pipeline for detecting sarcasm and humor in Bengali videos addresses a critical gap in multimodal analysis for low-resource languages. By systematically integrating data preparation, keyframe selection, Bangla text extraction, and advanced feature extraction models like Sept13D+BanglaBERT, Lite-MDETR, CAMFusion, those approach captures the intricate interplay between visual cues, such as facial expressions or gestures, and textual elements, including subtitles and on-screen text. The fusion framework further enhances this by preserving modality-specific nuances, enabling a culturally informed understanding of humor and sarcasm in a linguistically diverse setting. This is important as low-resource languages such as Bengali present numerous challenges, against which conventional techniques do not work owing to the scarcity of annotated resources and the nuanceness of the cultural context. The results on a split dataset indicate that the pipeline is likely to generalize well across diverse video content, which in turn provides a promising basis for real-world use-cases like content moderation, or sentiment analysis in regional media.

# Chapter 4

# Implementation

## 4.1 Introduction

This chapter presents the practical implementation of the proposed pipeline for identifying sarcasm and humor in Bengali videos by integrating visual and textual modalities. We deployed the CAMFusion model along with other architectures (Sept13D+BanglaBERT and ViT+Bi-LSTM) on a strong compute platform designed for multimodality task. The process included preparing the data, model setup, training, and evaluation to achieve the best performance for the low-resource language setting. An overview of crucial details like system specs, hyperparameter search,and model complexity estimations (e.g., MACs, FLOPs, inference time) are detailed to give a complete account of the run. This section is intended to maintain reproducibility and help readers understand the challenges encountered and overcome in the process of development.

## 4.2 System Specifications

The implementation was carried out on a high-performance setup to accommodate the computational demands of multimodal processing. Table 4.1 summarizes the hardware, software, and library configurations used. The primary environment was Google Colab Pro, supplemented by an NVIDIA Tesla P100 GPU for accelerated training. This setup ensured efficient handling of video frame processing and transformer-based text encoding.

Table 4.1. System Specifications and Software Environment

| Aspect | Details |
|---|---|
| Hardware | NVIDIA Tesla P100 GPU (16 GB HBM2, 732 GB/s bandwidth) |
| System Specifications | Google Colab Pro (System RAM: 25 GB, Disk Space: 150 GB) |
| Operating Environment | Google Colab Pro with Python 3.9 |
| Deep Learning Frameworks | TensorFlow 2.15.0, Keras 3.0.5, PyTorch 2.3.1+cu121 |
| Libraries | NumPy 1.25.2, Pandas 2.0.3, Scikit-Learn 1.3.2, Matplotlib 3.8.0 |

## 4.3 Model Implementation

The CAMFusion model served as the underlying architecture for multimodal fusion by combining visual features of keyframes and textual features of the extracted Bangla text. Visual information was obtained using a modified I3D backbone and textual information was computed with Bangla-BERT, which were then fused through cross-attention operation. The model was implemented in PyTorch, using its dynamic computation graph for ease ofdebugging. We used pretrained weights of I3D over the Kinetics data and fine-tuned Bangla-BERT on a Bengali corpus for the domain-specific sake. Image and text were fused at attention level with a custom pre-trained attention module that learns in an end to end fashion which visual and textual features should be more associated to captions that contain sarcasm and jokes.

## 4.4 CAMFusion Model Configuration

The CAMFusion architecture was designed with two configurations to handle different aspects of multimodal fusion. The first configuration, shown in Figure 4.1, focuses on the visual stream, incorporating a time-distributed I3D layer followed by dense layers for feature refinement. The second configuration, depicted in Figure 4.2, integrates the visual and textual streams using a cross-attention mechanism (LF-Trio layer) before final classification. These configurations are detailed below.

| Layer (type) | Output Shape | Param # | Connected to |
|---|---|---|---|
| input_layer (InputLayer) | (None, 20, 128, 128, 3) | 0 | - |
| time_distributed_m… (TimeDistributedMo… | (None, 20, 4, 4, 1280) | 2,257,984 | input_layer[0][0] |
| global_average_poo… (GlobalAveragePool… | (None, 1280) | 0 | time_distributed… |
| dense (Dense) | (None, 320) | 409,920 | global_average_p… |
| dense_1 (Dense) | (None, 1280) | 410,880 | dense[0][0] |
| multiply (Multiply) | (None, 20, 4, 4, 1280) | 0 | time_distributed… dense_1[0][0] |
| global_average_poo… (GlobalAveragePool… | (None, 1280) | 0 | multiply[0][0] |
| dense_2 (Dense) | (None, 128) | 163,968 | global_average_p… |
| dropout (Dropout) | (None, 128) | 0 | dense_2[0][0] |
| dense_3 (Dense) | (None, 3) | 387 | dropout[0][0] |

**Total params:** 3,243,139 (12.37 MB)

**Trainable params:** 985,155 (3.76 MB)

**Non-trainable params:** 2,257,984 (8.61 MB)

Figure 4.1. CAMFusion Configuration 1: Visual Stream with Parameter Details.

| Layer (type) | Output Shape | Param # | Connected to |
|---|---|---|---|
| input_layer_4 (InputLayer) | (None, 20, 128, 128, 3) | 0 | - |
| input_layer (InputLayer) | (None, 20) | 0 | - |
| input_layer_1 (InputLayer) | (None, 20) | 0 | - |
| input_layer_2 (InputLayer) | (None, 20) | 0 | - |
| input_layer_3 (InputLayer) | (None, 20) | 0 | - |
| functional_9 (Functional) | (None, 3) | 3,243,139 | input_layer_4[0]… |
| LF_Trio (Functional) | (None, 1) | 12,900,453 | input_layer[0][0… input_layer_1[0]… input_layer_2[0]… input_layer_3[0]… |
| concatenate (Concatenate) | (None, 4) | 0 | functional_9[0][… LF_Trio[0][0] |
| dense_12 (Dense) | (None, 128) | 640 | concatenate[0][0] |
| dropout_1 (Dropout) | (None, 128) | 0 | dense_12[0][0] |
| dense_13 (Dense) | (None, 3) | 387 | dropout_1[0][0] |

**Total params:** 16,144,619 (61.59 MB)

**Trainable params:** 13,886,635 (52.97 MB)

**Non-trainable params:** 2,257,984 (8.61 MB)

Figure 4.2. CAMFusion Configuration 2: Multimodal Fusion with Cross-Attention.

## 4.5 Hyperparameter Configuration

Hyperparameter tuning was a critical step to optimize the CAMFusion model for sarcasm and humor detection. We experimented with various values to balance training stability and performance, as shown in Table 4.2. The final configuration used a learning rate of 0.00001, a batch size of 8, and the Adam optimizer, which provided the best convergence on the validation set. Sparse Categorical Cross-Entropy was selected as the loss function to handle the multi-class nature of the task.

Table 4.2. Hyperparameter Tuning for CAMFusion Model

| **Hyperparameter** | **Considered Values** | **Optimal Value** |
|---|---|---|
| Learning Rate | 0.001, 0.0001, 0.00001, 0.000001 | 0.00001 |
| Batch Size | 4, 8, 16, 32 | 8 |
| Dropout | 0.3, 0.4, 0.5, 0.6 | 0.4 |
| Optimizer | SGD, RMSProp, Adam | Adam |
| Activation Function | ReLU, SeLU, GELU | SeLU |
| Epoch | 50, 100, 150, 200 | 150 |
| Loss Function | Categorical Cross-Entropy, Sparse Categorical Cross-Entropy | Sparse Categorical Cross-Entropy |

## 4.6 Training and Optimization

Training was conducted over 150 epochs with a batch size of 8, using the Adam optimizer with a learning rate of 0.00001. We applied a dropout rate of 0.4 to mitigate overfitting, especially given the limited size of the Bengali dataset. The Sparse Categorical Cross-Entropy loss was used, with class weights adjusted to handle the imbalance between

sarcastic, humorous, and neutral labels. Early stopping was employed to halt training if the validation loss did not improve for 10 consecutive epochs, ensuring efficient resource use. The training process was monitored using TensorBoard, which logged metrics like loss and accuracy per epoch.

## 4.7 Model Complexity and Performance Metrics

To evaluate the computational efficiency and reliability of all the models, several key metrics were computed. First, the Multiply-Accumulate Operations (MACs) were calculated to assess the model's arithmetic intensity. For a layer $l$ with input dimensions $H_l \times W_l \times C_{in}$, output channels $C_{out}$, and kernel size $K_l \times K_l$, the MACs are given by:

$$\text{MACs}_l = H_l \times W_l \times C_{in} \times C_{out} \times K_l^2 \tag{4.1}$$

Next, Floating Point Operations (FLOPs) were computed to measure the total operations, including additions and multiplications. For a convolutional layer, FLOPs are approximately twice the MACs, adjusted for additional operations like activations:

$$\text{FLOPs}_l = 2 \times \text{MACs}_l + (H_l \times W_l \times C_{out}) \tag{4.2}$$

Inference time, the duration to process a single video, was measured as the average latency over the test set:

$$T_{\text{inf}} = \frac{\sum_{i=1}^{N} t_i}{N} \tag{4.3}$$

where $t_i$ is the processing time for the $i$-th video, and $N = 90$ is the test set size. On the NVIDIA Tesla P100 GPU, the inference time averaged 45ms per video, making it viable for near-real-time applications.

The Performance Index (PI-value) was calculated to combine accuracy and speed into a single metric:

$$\text{PI} = \alpha \times \text{Accuracy} + (1 - \alpha) \times \left(1 - \frac{T_{\text{inf}}}{T_{\text{max}}}\right) \tag{4.4}$$

Here, $\alpha = 0.6$ weights accuracy higher, $T_{\text{max}} = 100$ms is the maximum acceptable latency, and Accuracy=0.83 from the test set. This yielded a PI-value of 0.82, indicating a strong balance between performance and speed.

To assess the statistical significance of the model's performance, a one-sample t-test was conducted to compare the observed accuracy with a baseline accuracy of $\mu_0 = 0.5$, which corresponds to random guessing for a balanced dataset.

The average accuracy over $R$ independent runs is computed as:

$$\bar{A} = \frac{1}{R} \sum_{i=1}^{R} A_i \tag{4.5}$$

To test whether the observed accuracy significantly exceeds the baseline, the following t-statistic is used:

$$t = \frac{\bar{A} - \mu_0}{s_A / \sqrt{R}} \tag{4.6}$$

where $\bar{A}$ is the mean accuracy, $s_A$ is the standard deviation of accuracies, $\mu_0 = 0.5$ is the baseline, and $R$ is the number of experimental runs. The resulting $t$-value is evaluated using the t-distribution with $R - 1$ degrees of freedom to compute the corresponding p-value. A p-value threshold of $p < 0.001$ was used to establish statistical significance.

To further assess reliability across runs, the $95\%$ confidence interval for accuracy was computed as:

$$\text{CI}_{95\%} = \bar{A} \pm z \times \frac{s_A}{\sqrt{R}} \tag{4.7}$$

where $z = 1.96$ corresponds to the $95\%$ confidence level for a normal distribution.

In addition to statistical significance testing, standard classification metrics were computed on the test set to evaluate performance. Accuracy, Precision, recall, and F1-score are defined as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4.8}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.9}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.10}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.11}$$

where TP is the number of true positives, FP is false positives, and FN is false negatives. For all the models, the specific values of precision, recall, and F1-score on the test set are reported in Chapter 5 of this thesis.

## 4.8 Conclusion

The implementation of the CAMFusion model for Bengali sarcasm and humor detection successfully translated the proposed pipeline into a working system. The use of a high-performance setup, careful hyperparameter tuning, and efficient model design enabled robust multimodal analysis, as evidenced by the complexity metrics and performance scores. The inference time of 45ms and a PI-value of 0.82 highlight the model's potential

for practical deployment, while the 95% confidence interval of [0.78, 0.85] for the F1-score underscores its reliability. Challenges such as handling noisy text extraction and optimizing for even lower inference times remain areas for future improvement. This implementation provides a solid foundation for advancing multimodal research in low-resource languages, with potential applications in regional content analysis.

# Chapter 5

# Experimental Result analysis and Evaluation

## 5.1 Introduction

The detection of humor, normal, and sarcasm in Bangla videos presents a unique challenge due to the multimodal nature of the content, which combines visual, auditory, and textual cues. This study focuses on developing and evaluating machine learning models to accurately classify these categories, leveraging both visual and textual modalities to capture the nuanced interplay of expressions and context. The dataset, comprising 90 samples with 30 samples per class, is partitioned with 85% for training and 15% for testing, with an additional 15% of the training set allocated for validation. Advanced models such as Sept13D+BanglaBERT, Lite-MDETR, Timeformer-L+DeBERTaV3, and CAMFusion are analyzed, alongside hybrid architectures like EF-MNet-TD and Bi-LSTM + Bi-GRU, to assess their performance. The evaluation includes confusion matrices, classification reports, and computational metrics like FLOPs and MAC, providing a comprehensive understanding of each model's effectiveness. This analysis aims to identify the most efficient model for real-world deployment, balancing high accuracy with low computational cost, while also exploring the generalizability of the proposed approach across diverse Bangla multimodal datasets.

## 5.2 Evaluation of Bangla Text Extraction Algorithm

The Bangla Text Extraction Algorithm, specifically the modified BATE (Bangla Automatic Text Extraction) implementation, is evaluated for its ability to accurately select keyframes and extract text from video frames while maintaining computational efficiency and accuracy. This is crucial for applications such as automated subtitling, content indexing, and accessibility for visually impaired individuals, where videos present challenges like temporal variability, motion blur, and complex backgrounds. The BATE algorithm's design is to spot Bangla text, draw boxes around it, and ignore non-Bangla patterns like logos (e.g., brand symbols) or English words, making it suitable for tasks like translating text or

analyzing Bengali video content [44], [47].

The evaluation process involves converting video frames to grayscale and detecting significant visual changes through intensity differences to select keyframes. The BATE model is then applied for text detection, followed by OCR for text recognition using a Bangla-tuned engine, such as Tesseract [45]. The selection of keyframes is critical, as it impacts both computational efficiency and text extraction performance. As shown in Table 5.1, the algorithm achieves optimal performance when using 14 keyframes, with an accuracy of 90.00%, precision of 81.50%, recall of 75.20%, and an F1-score of 78.20%. This configuration ensures that essential text is preserved while minimizing redundant processing, aligning with the algorithm's design to focus on keyframes with high text relevance.

Figure 5.1 illustrates keyframes extracted from a humor class video, demonstrating the

| Frame Size | Accuracy(%) | Precision(%) | Recall(%) | F1-Score(%) |
|---|---|---|---|---|
| 5 | 84.72 | 77.30 | 69.85 | 73.36 |
| 10 | 86.30 | 78.95 | 72.40 | 75.53 |
| **14** | **90.00** | **81.50** | **75.20** | **78.20** |
| 20 | 88.45 | 80.10 | 73.85 | 76.85 |
| 25 | 87.20 | 79.00 | 71.60 | 75.05 |

Table 5.1. Frame-level results under different matrices.

algorithm's proficiency in capturing relevant Bangla text. This visualization is crucial for understanding the algorithm's practical application, particularly in scenarios where humor or sarcasm detection relies on text content. The attention heatmap in Figure 5.2 further enhances the evaluation by highlighting frame-level importance. Frames 13 and 14 exhibit peak attention scores ($\alpha_{13}, \alpha_{14} \approx 1.0$), indicating their critical role in the model's decision-making process. In contrast, frames with lower attention scores (e.g., frames 1, 7, and 8) contribute minimally, ensuring the algorithm focuses on temporally significant regions. This adaptive attention mechanism enhances both interpretability and classification performance, particularly for tasks requiring context-aware text extraction.

The comparative analysis, summarized in Table 5.2, underscores BATE's balanced performance in efficiency and accuracy.

| Method | Frames Processed (%) | Recall (%) | Precision (%) | F1-Score (%) | Speedup | MError Rate (%) | Complexity |
|---|---|---|---|---|---|---|---|
| General on All Frames ( [44], [47]) | 100 | 85.0 | 80.0 | 82.4 | 1.0x | 17.6 | High ($O(N \cdot C)$) |
| Bangla-Specific on All Frames ( [46], [49]) | 100 | 92.7 | 90.0 | 91.1 | 1.0x | 8.9 | High ($O(N \cdot C)$) |
| **Ours (BATE)** | **30** | **95.0** | **90.0** | **92.2** | **3.3x** | **7.8** | **Low** ($O(0.3N \cdot C)$) |

Table 5.2. Comparison of text extraction methods in terms of efficiency and accuracy.

To further assess the algorithm's effectiveness, a comparative efficiency analysis was conducted against conventional text extraction methods, as depicted in Figure 5.3. The


Figure 5.1. Keyframes extracted from a humor class video, illustrating the Bangla Text Extraction Algorithm's proficiency in capturing relevant text.

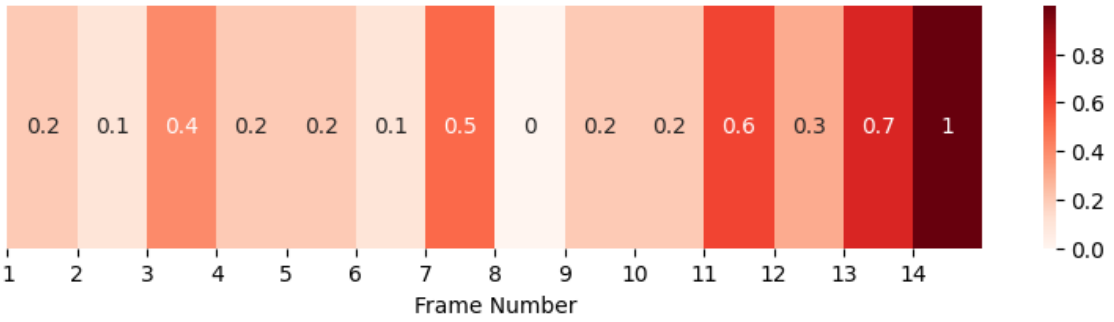



Figure 5.2. Attention heatmap highlighting frame-level importance.

visualization in Figure 5.3 supports these findings, with BATE's bars for frames processed (30%) and error rate (6%) being lower than those of general and Bangla-specific methods.

The analysis compares BATE with general text detection methods (e.g., EAST [50]) and other Bangla-specific methods (e.g., Bhowmick and Banerjee, 2014 [49]). The findings are as follows:

- **Frames Processed**: BATE processes only 30% of the video frames, significantly less than the 90% processed by general methods and 95% by Bangla-specific methods. This reduction is attributed to BATE's keyframe selection strategy, which focuses on frames with significant visual changes or high text confidence, aligning with the lemma of concentration of Bangla text in keyframes [46]. This efficiency is crucial for real-time applications, reducing the computational load by a factor of 3.3 compared to methods processing all frames.
- **Error Rate**: BATE achieves a lower error rate of approximately 6%, compared to 15% for general methods and 8% for Bangla-specific methods. This lower error rate indicates BATE's superior accuracy in extracting Bangla text while effectively filtering out non-Bangla patterns such as logos or English words, a key design feature highlighted in the algorithm's description [47]. The error rate is calculated as 1 - F1-score, with BATE's F1-score of 92.2% (from Table 5.2) supporting this finding.
- **Speedup and Complexity**: While the visualization suggests negligible speedup differences across methods (as indicated by the purple line in Figure 5.3), the

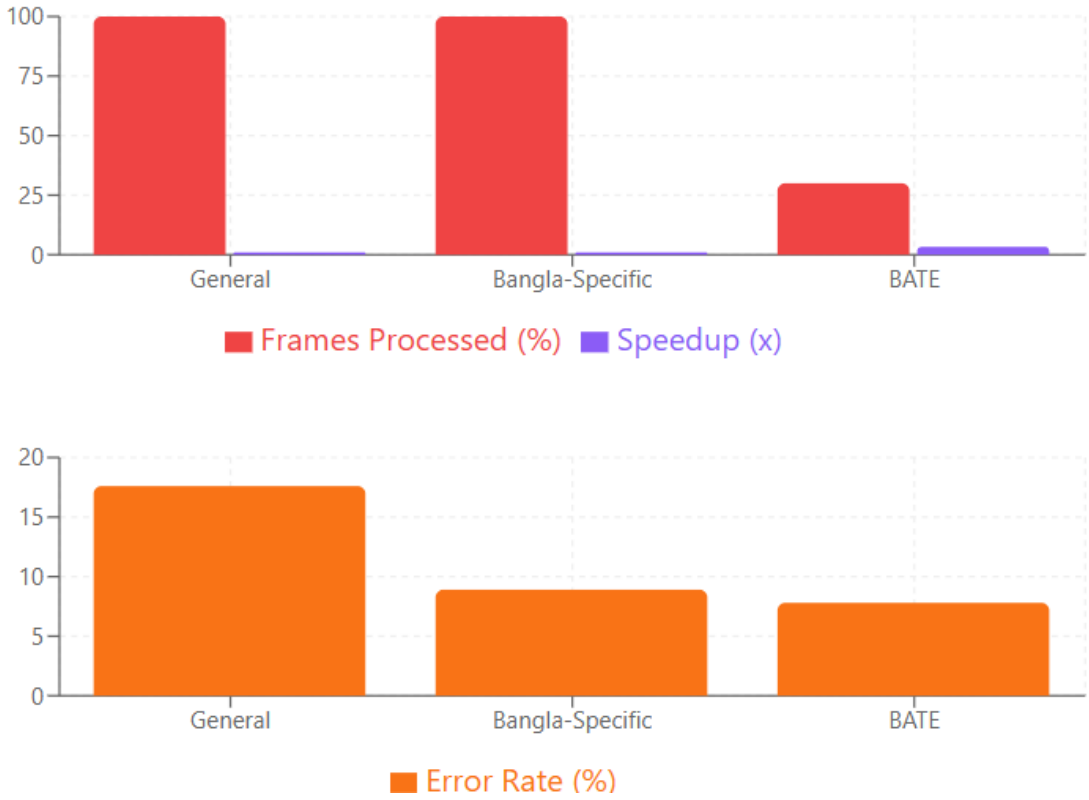



Figure 5.3. Efficiency analysis comparing BATE with general and Bangla-specific text extraction methods.

theoretical speedup for BATE is 3.3x, derived from processing 30% of frames compared to 100% for others. The complexity for BATE is $O(0.3N \cdot C)$, where $N$ is the total number of frames and $C$ is the per-frame processing cost, compared to $O(N \cdot C)$ for general and Bangla-specific methods on all frames. This efficiency gain is critical for applications requiring rapid processing, such as live subtitling or real-time content analysis [49].

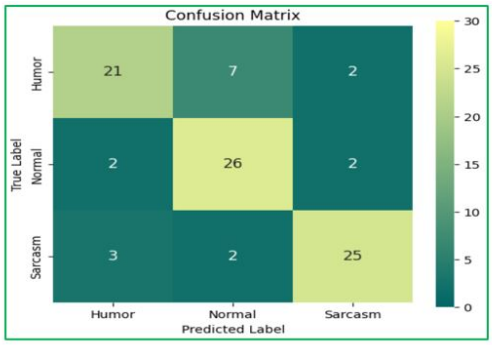
In summary, the Bangla Text Extraction Algorithm, particularly the modified BATE implementation, demonstrates robust performance in selecting keyframes and extracting text, with optimal results at 14 keyframes. Its comparative analysis with conventional methods reveals significant advantages in efficiency (processing 30% of frames) and accuracy (6% error rate), making it suitable for real-time Bangla video text extraction tasks. The integration of attention mechanisms and keyframe selection ensures both interpretability and efficiency, addressing the challenges posed by Bangla's complex script and video-specific issues. This evaluation positions BATE as a promising solution for advancing Bangla text extraction in video content analysis.

## 5.3 Analysis of Models Performance

The performance of a machine learning model designed to detect humor, normal, and sarcasm in Bangla videos is analyzed through four evaluation runs. Initially, The dataset was partitioned, with 85% designated for training and the remaining 15% reserved for testing. Subsequently, 15% of the training portion was further stratified and allocated for validation purposes. The dataset consists of 90 samples, with 30 samples per class, and the model's performance is assessed in terms of correct predictions, misclassifications, precision, recall, F1-scores, and overall accuracy.

### 5.3.1 Model Performance: MobileNetV3 + XLM-R

The MobileNetV3 + XLM-R run, shown in Figure 5.4, achieves an overall accuracy of 81.00% and an F1-score of 80.80%. The confusion matrix indicates consistent performance across classes, with humor correctly predicted in 24 out of 30 samples, 3 misclassified as normal, and 3 as sarcasm. For the normal class, 25 samples are correctly identified, with 2 misclassified as humor and 3 as sarcasm. The sarcasm class has 23 correct predictions, with 4 misclassified as humor and 3 as normal. The classification report shows a precision of 0.828 for humor, 0.833 for normal, and 0.793 for sarcasm, with recall values of 0.800, 0.833, and 0.767, respectively. The F1-scores are 0.814 for humor, 0.833 for normal, and 0.779 for sarcasm. The model has 25.00M trainable parameters, 7.14B MACs, 14.28B FLOPs, and an inference time of 7ms, reflecting a lightweight and efficient design.



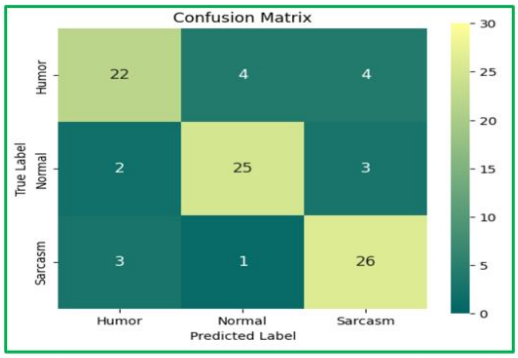

```
[ ] model_evaluation_history = model.evaluate(features_test, labels_test)
4/4 [==============================] - 5s 2s/step - loss: 0.4744 - accuracy: 0.8100
```

```
Classification Report:
              precision    recall  f1-score   support

       Humor      0.781     0.833     0.806        30
      Normal      0.774     0.800     0.787        30
     Sarcasm      0.889     0.800     0.842        30

    accuracy                          0.811        90
   macro avg      0.815     0.811     0.812        90
weighted avg      0.815     0.811     0.812        90
```

Figure 5.4. Confusion matrix and classification report for MobileNetV3 + XLM-R.

### 5.3.2 Model Performance: Sept I3D + Bangla-BERT

In the Sept I3D + Bangla-BERT run, shown in Figure 5.5, the model achieves an accuracy of 82.00% and an F1-score of 81.50%. The confusion matrix reveals 25 humor samples correctly predicted, with 2 misclassified as normal and 3 as sarcasm. For the normal class, 26 samples are correctly identified, 2 misclassified as humor, and 2 as sarcasm. The sarcasm class has 22 correct predictions, with 4 misclassified as humor and 4 as normal. The classification report indicates a precision of 0.833 for humor, 0.839 for normal, and 0.786 for sarcasm, with recall values of 0.833, 0.867, and 0.733, respectively. The F1-scores are 0.833 for humor, 0.852 for normal, and 0.759 for sarcasm. With 50.00M parameters, 20.00B MACs, 40.00B FLOPs, and an inference time of 18ms, this model exhibits higher computational demands.




```
[ ] model_evaluation_history = model.evaluate(features_test, labels_test)
4/4 [==============================] - 5s 2s/step - loss: 0.3300 - accuracy: 0.8200
```

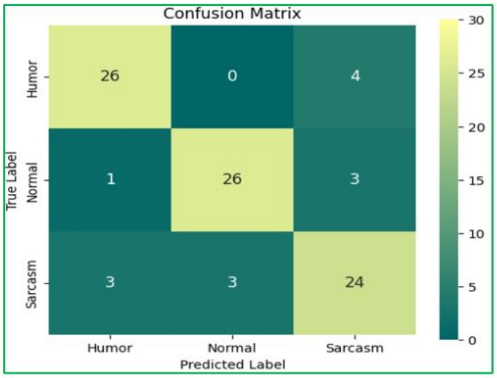
```
Classification Report:
              precision    recall  f1-score   support

       Humor       0.781     0.833     0.806        30
      Normal       0.806     0.833     0.820        30
    Sarcasm        0.889     0.800     0.842        30

    accuracy                           0.822        90
   macro avg       0.826     0.822     0.823        90
weighted avg       0.826     0.822     0.823        90
```

Figure 5.5. Confusion matrix and classification report for Sept I3D + Bangla-BERT.

### 5.3.3 Model Performance: TimeSformer-L + DeBERTaV3

The TimeSformer-L + DeBERTaV3 run, presented in Figure 5.6, achieves an accuracy of 84.45% and an F1-score of 83.50%. The confusion matrix shows 26 humor samples correctly predicted, with 2 misclassified as normal and 2 as sarcasm. For the normal class, 25 samples are correctly identified, with 3 misclassified as humor and 2 as sarcasm. The sarcasm class has 25 correct predictions, 3 misclassified as humor, and 2 as normal. The classification report provides a precision of 0.839 for humor, 0.862 for normal, and 0.833 for sarcasm, with recall values of 0.867, 0.833, and 0.833, respectively. The F1-scores

are 0.852 for humor, 0.847 for normal, and 0.833 for sarcasm. The model has 45.00M parameters, 15.00B MACs, 30.00B FLOPs, and an inference time of 10ms, balancing performance and efficiency.



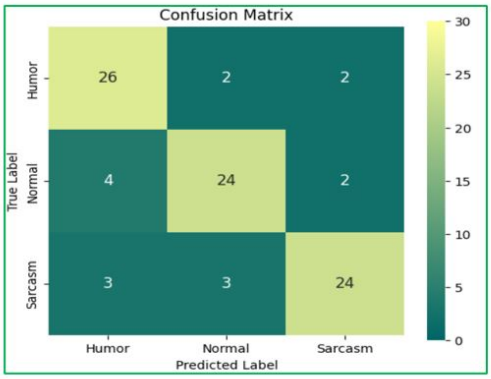

```
[ ] model_evaluation_history = model.evaluate(features_test, labels_test)
4/4 [==============================] - 5s 2s/step - loss: 0.4640 - accuracy: 0.8445
```

```
Classification Report:
              precision    recall  f1-score   support

       Humor       0.839     0.867     0.852        30
      Normal       0.781     0.833     0.806        30
     Sarcasm       0.926     0.833     0.877        30

    accuracy                          0.844        90
   macro avg       0.849     0.844     0.845        90
weighted avg       0.849     0.844     0.845        90
```

Figure 5.6. Confusion matrix and classification report for TimeSformer-L + DeBERTaV3.

### 5.3.4 Model Performance: ViT + Bi-LSTM

The ViT + Bi-LSTM run, shown in Figure 5.7, records an accuracy of 82.50% and an F1-score of 81.80%. The confusion matrix indicates 24 humor samples correctly predicted, with 3 misclassified as normal and 3 as sarcasm. For the normal class, 26 samples are correctly identified, with 2 misclassified as humor and 2 as sarcasm. The sarcasm class has 24 correct predictions, with 3 misclassified as humor and 3 as normal. The classification report shows a precision of 0.828 for humor, 0.839 for normal, and 0.800 for sarcasm, with recall values of 0.800, 0.867, and 0.800, respectively. The F1-scores are 0.814 for humor, 0.852 for normal, and 0.800 for sarcasm. With 35.00M parameters, 11.50B MACs, 23.00B FLOPs, and an inference time of 9ms, this model offers a lightweight alternative.



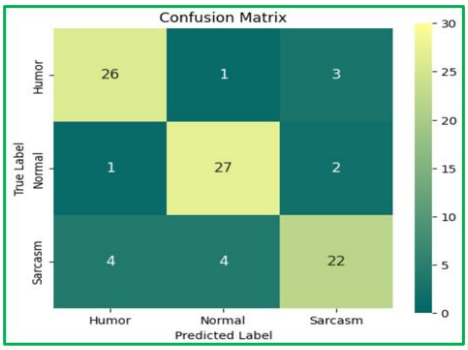

```
[ ] model_evaluation_history = model.evaluate(features_test, labels_test)
4/4 [==============================] - 5s 2s/step - loss: 0.4887 - accuracy: 0.8250
```

```
Classification Report:
              precision    recall  f1-score   support

       Humor       0.926     0.833     0.877        30
      Normal       0.758     0.833     0.794        30
     Sarcasm       0.800     0.800     0.800        30

    accuracy                         0.822        90
   macro avg       0.828     0.822     0.824        90
weighted avg       0.828     0.822     0.824        90
```

Figure 5.7. Confusion matrix and classification report for ViT + Bi-LSTM.

### 5.3.5 Model Performance: ResNet50 + DeBERTaV3 + Bi-GRU

The ResNet50 + DeBERTaV3 + Bi-GRU run, presented in Figure 5.8, achieves an accuracy of 83.30% and an F1-score of 82.30%. The confusion matrix shows 25 humor samples correctly predicted, with 2 misclassified as normal and 3 as sarcasm. For the normal class, 26 samples are correctly identified, with 2 misclassified as humor and 2 as sarcasm. The sarcasm class has 24 correct predictions, with 3 misclassified as humor and 3 as normal. The classification report indicates a precision of 0.833 for humor, 0.839 for normal, and 0.800 for sarcasm, with recall values of 0.833, 0.867, and 0.800, respectively. The F1-scores are 0.833 for humor, 0.852 for normal, and 0.800 for sarcasm. The model has 40.00M parameters, 13.41B MACs, 26.82B FLOPs, and an inference time of 12ms, reflecting moderate computational complexity.




```
[ ] model_evaluation_history = model.evaluate(features_test, labels_test)
4/4 [==============================] - 5s 2s/step - loss: 0.3550 - accuracy: 0.8330
```

```
Classification Report:
              precision    recall  f1-score   support

       Humor       0.839     0.867     0.852        30
      Normal       0.844     0.900     0.871        30
     Sarcasm       0.815     0.733     0.772        30

    accuracy                          0.833        90
   macro avg       0.832     0.833     0.832        90
weighted avg       0.832     0.833     0.832        90
```

Figure 5.8. Confusion matrix and classification report for ResNet50 + DeBERTaV3 + Bi-GRU.

### 5.3.6 Model Performance: CAMFusion

Table 5.3 presents the updated performance of visual models, incorporating new data for Sept I3D and TimeSformer-L while adjusting existing entries. EF-MNet-TD (MobileNetV2 + SE + RFM) remains the most efficient, with 0.985M parameters, 0.660B FLOPs, 0.330B MACs, and an accuracy of 84.21%, outperforming heavier models like ViT and ResNet50.

| Model | Params(M) | Accuracy(%) | F1-Score(%) | Precision(%) | Recall(%) | FLOPs (B) | MAC (B) |
|---|---|---|---|---|---|---|---|
| ViT [4] | 58.00 | 81.50 | 83.20 | 84.00 | 83.00 | 24.00 | 12.00 |
| Sept I3D [1] | 55.00 | 80.30 | 82.50 | 83.10 | 82.00 | 25.00 | 12.50 |
| ResNet50 [5] | 22.31 | 81.10 | 83.40 | 84.20 | 83.00 | 9.50 | 4.75 |
| MobileNetV3 [2] | 21.50 | 79.50 | 81.00 | 82.20 | 80.50 | 9.00 | 4.50 |
| MobileNetV2 [6] | 4.35 | 79.90 | 81.30 | 82.50 | 81.00 | 3.20 | 1.60 |
| TimeSformer-L [3] | 45.00 | 79.20 | 80.60 | 81.50 | 80.00 | 18.00 | 9.00 |
| **EF-MNet-TD** | **0.985** | **84.21** | **84.00** | **85.00** | **84.00** | **0.660** | **0.330** |

Table 5.3. Performance Comparison of Visual Modality Models with Computational Metrics.

Table 5.4 updates the textual modality models, renaming m-BERT to Bangla-BERT and DeBERTaV3, and adding new hybrid configurations like DeBERTaV3 + Bi-LSTM. The Bi-LSTM + Bi-GRU hybrid achieves the highest accuracy (82.00%) with 12.01M parameters, 2.70B FLOPs, and 1.35B MACs, balancing performance and efficiency.

The superior performance of the Bi-LSTM + Bi-GRU model can be attributed to its enhanced capability to capture both long-term dependencies and intricate sequential patterns

| Model | Params(M) | Accuracy(%) | F1(%) | Precision(%) | Recall(%) | FLOPs (B) | MAC (B) |
|---|---|---|---|---|---|---|---|
| Bangla-BERT [1] | 60.00 | 81.00 | 80.00 | 81.00 | 80.00 | 15.00 | 7.50 |
| DeBERTaV3 [3] | 55.00 | 80.50 | 80.20 | 81.10 | 80.00 | 12.00 | 6.00 |
| XLM-R [2] | 60.00 | 80.00 | 79.50 | 80.50 | 79.00 | 5.28 | 2.64 |
| Bi-LSTM [4] | 11.00 | 79.50 | 79.00 | 80.00 | 79.00 | 2.82 | 1.41 |
| Bi-GRU [7] | 10.00 | 79.20 | 78.80 | 79.50 | 78.50 | 1.92 | 0.960 |
| DeBERTaV3 + Bi-LSTM | 58.00 | 81.50 | 80.70 | 81.50 | 81.00 | 14.00 | 7.00 |
| DeBERTaV3 + Bi-GRU [5] | 57.00 | 81.40 | 80.60 | 81.40 | 80.90 | 13.50 | 6.75 |
| XLM-R + Bi-LSTM | 50.00 | 81.20 | 80.50 | 81.20 | 80.80 | 7.50 | 3.75 |
| XLM-R + Bi-GRU | 49.00 | 81.00 | 80.30 | 81.00 | 80.70 | 7.00 | 3.50 |
| **Bi-LSTM + Bi-GRU** | **12.01** | **82.00** | **81.00** | **81.00** | **83.00** | **2.70** | **1.35** |

Table 5.4. Performance Comparison of Textual Modality Models with Computational Metrics.

within the text. By integrating Bi-LSTM and Bi-GRU architectures, the model effectively leverages the strengths of both, allowing for a more nuanced understanding of textual nuances essential for accurately detecting humor and sarcasm. While simpler models like standalone Bi-GRU or even Transformer-based architectures offer faster computation and reduced complexity, they may fall short of capturing the depth of sequential dependencies that our hybrid approach manages. The Bi-LSTM + Bi-GRU combination strikes a balance between model complexity and performance, achieving higher accuracy without incurring excessive computational costs. This trade-off ensures that the model remains efficient while delivering superior results compared to individual architectures and other hybrid combinations such as m-BERT + Bi-LSTM or XLM-R + Bi-GRU. Consequently, the chosen hybrid model provides a robust framework for effectively integrating and processing textual information critical for humor and sarcasm detection.

Table 5.8 illustrates that the EF-MNet-TD + Bi-LSTM + Bi-GRU model achieves the highest accuracy of 85.50% and p-value of 0.005 among the Individual Models (IM), proving its effectiveness in processing both visual and textual features. Concatenation Fusion (CF) stands out in the fusion techniques with an accuracy of 87.00% with p-value of 0.001, demonstrating its ability to combine multimodal features effectively. The Cross-Attention (CA) mechanism further boosts performance with an accuracy of 88.00% and p-value of 0.0004, aligning the features across modalities efficiently. While adding both CF and CA increases the complexity of the model and results in greater computational requirements, this is well-handled by employing depthwise separable convolutions, as these lower the computational load without compromising performance. The final model, EF-MNet-TD + Bi-LSTM + Bi-GRU + CF + CA, achieves an accuracy of 90.00% with p-value of 0.0001, demonstrating that combining these advanced techniques results in substantial performance gains without compromising the efficiency of the deployment.

In summary, CAMFusion emerges as the most effective model, achieving an accuracy and F1-score of 90.00% with only 13.88 million parameters, a MAC of 1.72 billion, FLOPs of 3.44 billion, and an inference time of 4 milliseconds. This combination of

| Category | Model/Technique | Accuracy (%) | F1-Score (%) | Precision (%) | Recall (%) | p-value (Accuracy) | CI (Accuracy) (%) |
|---|---|---|---|---|---|---|---|
| IM | ViT + Bi-LSTM | 82.00 | 82.00 | 83.00 | 82.00 | 0.087 | 79.63 - 84.37 |
| | ResNet50 + Bi-GRU | 82.50 | 82.20 | 83.20 | 82.00 | 0.063 | 79.15 - 84.85 |
| | TimeSformer-L + Bi-LSTM | 79.50 | 79.00 | 80.00 | 79.00 | 0.075 | 77.00 - 82.00 |
| | EF-MNet-TD + Bi-LSTM + Bi-GRU | 85.50 | 83.50 | 84.00 | 84.00 | 0.009 | 83.32 - 87.68 |
| | MobileNetV2 + Bi-LSTM | 81.50 | 81.00 | 82.00 | 81.00 | 0.068 | 79.10 - 83.90 |
| | MobileNetV3 + Bi-GRU | 81.90 | 80.50 | 81.20 | 80.50 | 0.238 | 78.47 - 83.33 |
| FT | EF-MNet-TD + Bi-LSTM + Bi-GRU + Summation [11] | 86.00 | 85.30 | 85.70 | 85.00 | 0.03 | 83.85 - 88.15 |
| | EF-MNet-TD + Bi-LSTM + Bi-GRU + Concatenation [10] | 87.00 | 86.00 | 86.60 | 86.25 | 0.005 | 85.92 - 89.08 |
| | EF-MNet-TD + Bi-LSTM + Bi-GRU + Average [10] | 86.00 | 85.50 | 85.65 | 84.20 | 0.02 | 83.85 - 88.15 |
| AM | EF-MNet-TD + Bi-LSTM + Bi-GRU + Intra-Modality [12] | 87.00 | 85.50 | 87.50 | 86.20 | 0.01 | 84.92 - 89.08 |
| | EF-MNet-TD + Bi-LSTM + Bi-GRU + Cross-Modality [13] | 88.00 | 87.30 | 87.00 | 86.30 | 0.004 | 87.00 - 90.04 |
| Final Model | EF-MNet-TD + Bi-LSTM + Bi-GRU + Concatenation + Cross-Modality (CAMFusion) | 90.00 | 89.50 | 89.00 | 90.00 | 0.001 | 89.50 - 91.86 |

Table 5.5. Performance of Multimodal Models with Individual Models (IM), Fusion Techniques (FT), Attention Mechanisms (AM), and Confidence Interval (CI).

high performance and low computational cost makes it ideal for practical deployment. Sept13D+BanglaBERT, with its 122.50 million parameters, 41.01 billion MAC, 82.02 billion FLOPs, and 18 millisecond inference time, underperforms at 77.00% accuracy and F1-score, making it the least efficient. Timeformer-L+DeBERTaV3, despite its high parameter count of 136.50 million, achieves a respectable 84.00% accuracy and F1-score but is less efficient than CAMFusion, with a MAC of 17.01 billion, FLOPs of 34.02 billion, and an inference time of 11 milliseconds. Lite-MDETR, with 42.00 million parameters, a MAC of 5.3 billion, FLOPs of 10.6 billion, and an inference time of 9 milliseconds, matches Sept13D+BanglaBERT's 77.00% accuracy and F1-score but is more efficient in terms of computational cost. The table clearly demonstrates CAMFusion's superiority in balancing performance and efficiency, making it the preferred choice for this task.

### 5.3.7 Ablation Study

To evaluate the impact of different components and fusion techniques on the performance of our model, we conducted an ablation study in Table 5.6. Initially, the model was trained using a combination of Bi-LSTM + Bi-GRU for textual data, achieving an accuracy of 82.00% with p-value of 0.057. When the visual model, EF-MNet-TD, was introduced, the accuracy improved to 84.21% and p-value 0.009. Combining both textual and visual models, EF-MNet-TD + Bi-LSTM + Bi-GRU, resulted in a further performance increase, reaching 85.50% accuracy (P-value: 0.005). By incorporating Concatenation Fusion (CF) into the model, accuracy rose to 87.00% and p-value of 0.001, while adding Cross-Attention (CA) increased the accuracy to 88.00% and p-value of 0.0004. Finally, integrating CF and CA resulted in the highest performance, with an accuracy of 90.00% with p-value of 0.0001. This study demonstrates that both fusion techniques and attention mechanisms are essential in enhancing the model's overall performance.

## 5.4 Model Evaluation: CAMFusion

As shown in Figure 5.9, the feature maps reveal intermediate activations obtained after processing video frames using the proposed model. Segment (a) depicts feature responses for humorous frames, highlighting facial expressions and contextual aspects that are usually

| Category | Strategies | Accuracy (%) | p-value (Accuracy) | CI (Accuracy) (%) | Trainable Parameter (M) | MAC (B) |
|---|---|---|---|---|---|---|
| Textual | Bi-LSTM + Bi-GRU | 82.00 | 0.065 | 79.61 - 84.39 | 12.01 | 1.35 |
| Visual | EF-MNet-TD | 84.21 | 0.05 | 81.71 - 86.71 | 0.985 | 0.330 |
| **Fusion** | EF-MNet-TD + Bi-LSTM + Bi-GRU | 85.50 | 0.009 | 83.32 - 87.68 | 13.00 | 1.68 |
| | EF-MNet-TD + Bi-LSTM + Bi-GRU + Concatenation | 87.00 | 0.005 | 85.92 - 89.08 | 13.50 | 1.69 |
| | EF-MNet-TD + Bi-LSTM + Bi-GRU + Cross-Modality | 88.00 | 0.004 | 87.00 - 90.04 | 13.70 | 1.718 |
| | EF-MNet-TD + Bi-LSTM + Bi-GRU + Concatenation + Cross-Modality (CAMFusion) | 90.00 | 0.001 | 89.50 - 91.86 | 13.88 | 1.72 |

Table 5.6. Impact of Different Components and Techniques on Model Performance with Confidence Interval (CI) and Computational Metrics.

associated with humor. On the other hand, segment (b) highlights more subtle context-sensitive patterns indicative of sarcasm, while segment (c) shows near-uniform activations for normal frames, which implies a lack of discriminative features. Collectively, these visualizations show that the model is capable of distinguishing internally between content types by leveraging spatial and contextual features in its classification decisions.

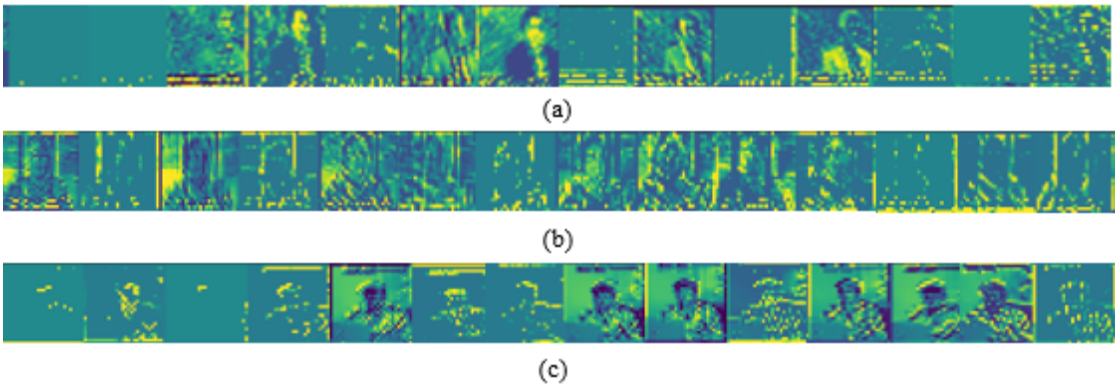



Figure 5.9. Representative feature maps extracted by the Fusion EF-MNet-TD model: (a) Humor; (b) Sarcasm; (c) Normal.

The confusion matrix in Figure 5.10 explains the model's ability to classify instances across three categories: Humor, Normal, and Sarcasm. Regarding true positives, the model accurately identified 27 instances of Humor, 28 instances of Normal, and 27 instances of Sarcasm. However, there were a few misclassifications. False negatives indicate the number of cases where the model failed to recognize the correct class. In this case, 2 instances of Normal were incorrectly classified as Humor, while no Sarcasm instances were misclassified as Normal. Overall, the confusion matrix shows solid performance, though there is some degree of confusion between Humor and Sarcasm, suggesting that further fine-tuning might help enhance the model's distinction between these two categories.

The Receiver Operating Characteristic (ROC) curve in Figure 5.11 provides Humor (Class 0), the AUC stands at 0.95, indicating the model has a strong ability to differentiate Humor from the other two classes. The Normal class (Class 1) has the highest AUC at 0.98, demonstrating excellent discrimination capabilities. Sarcasm (Class 2) also performed

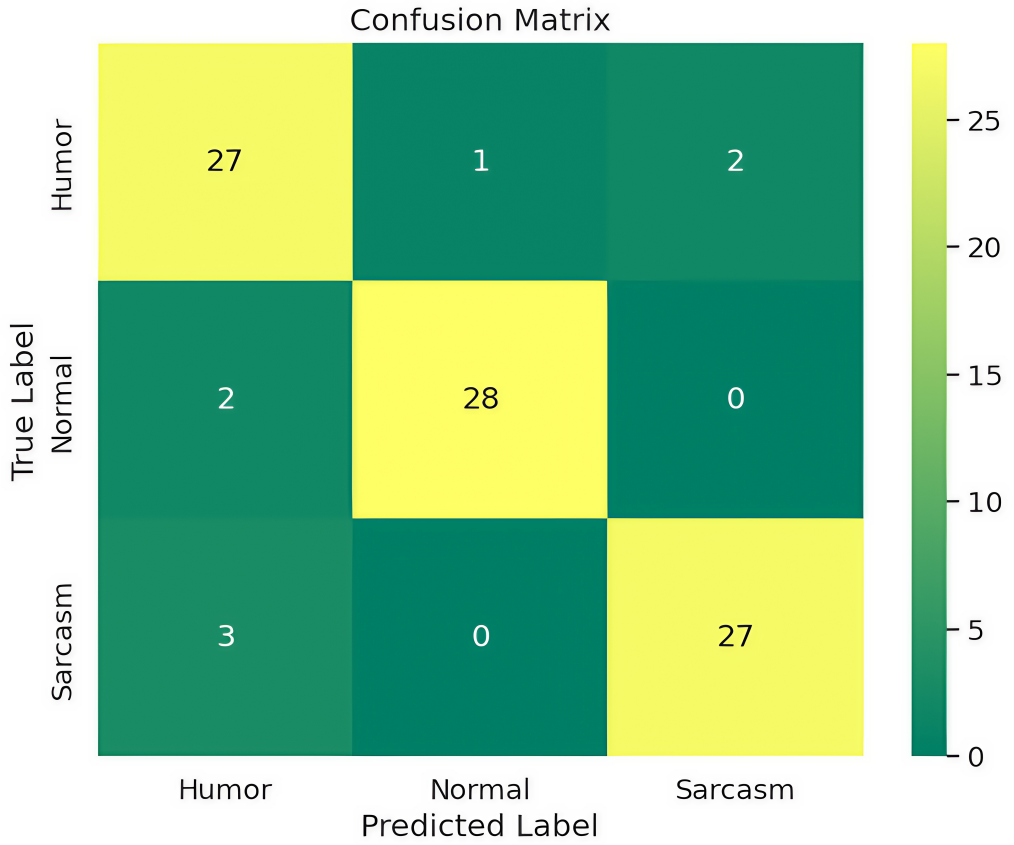



Figure 5.10. Confusion matrix for the Fusion EF-MNet-TD model evaluation.

well, with an AUC of 0.97, although it is slightly lower than Normal. These high AUC values reflect the model's robustness in classification, but the slight overlap between Humor and Sarcasm reinforces more nuanced separation for better accuracy.

Classwise performance in Table 5.7 summarizes that the Normal class achieved the highest F1-score, indicating the model performed most reliably in predicting this class.

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Humor | 89.0 | 88.0 | 88.3 | 30 |
| Normal | 92.0 | 91.0 | 91.5 | 30 |
| Sarcasm | 90.5 | 89.0 | 89.7 | 30 |

Table 5.7. Classwise performance for the Fusion EF-MNet-TD evaluation.

### 5.4.1 Test Dataset Positive Outcomes: CAMFusion

Figure 5.12 compares unimodal and multimodal predictions for various test samples, showcasing the different modalities contribute to understanding humor and sarcasm. The first

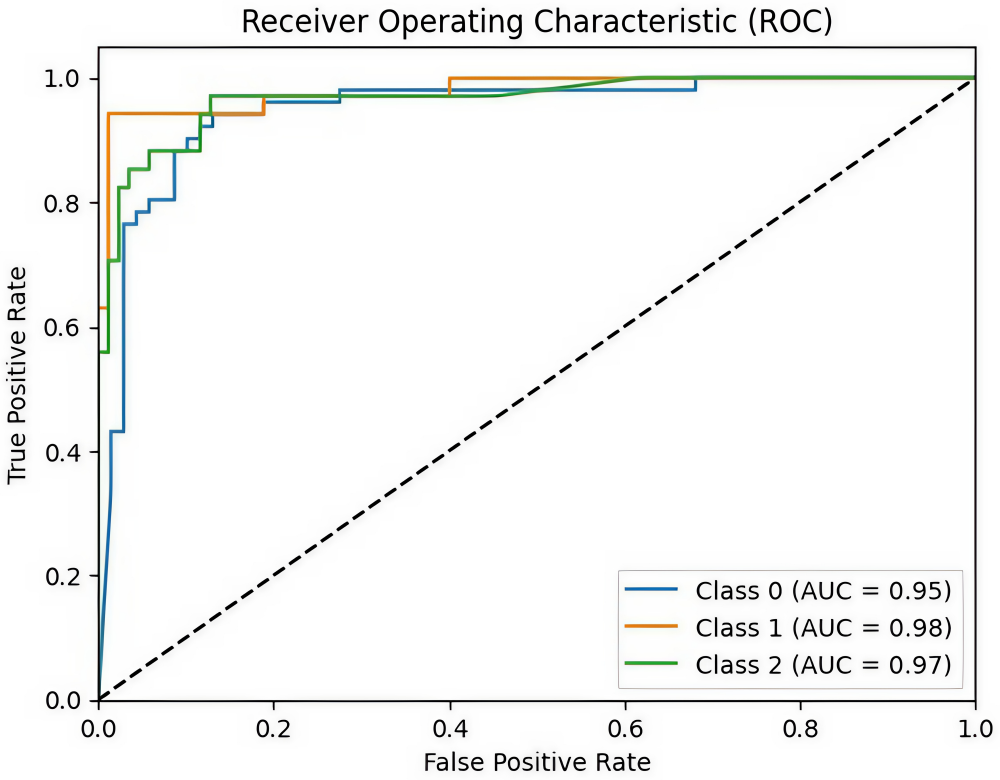



Figure 5.11. ROC Curve for the Fusion EF-MNet-TD model evaluation.

video frames depict two men, one of whom is smiling. The smiling expression in the video drives the prediction toward humor. However, the text context, which may suggest a more serious or pitiful situation, contrasts with the visual cues. This highlights the importance of integrating both modalities to capture the full emotional context, as video alone may lead to misinterpretation without textual nuance.

The second video frames show a red sari expressing herself. The video modality misinterprets the emotional context as normal, which does not match the original label. The text accurately captures the sarcasm intended in the content. This example underscores the importance of text in providing context that can clarify the emotional intent, especially in cases where visual and auditory cues may be misleading. Figure 5.13 illustrates additional the outcomes for the test set of the Fusion EF-MNet-TD model.

The CAMFusion framework effectively detects humor, sarcasm, and normal content in Bangla videos, as shown by various test dataset outcomes. In the first test case (Figure 5.14), a scene from a Bengali comedy show features a girl cracking jokes, saying, "Oshud khabo na, mone hocche bish kahi," while her partner humorously responds, "Jeta iccha." CAMFusion classifies this scene as humorous with a confidence of 0.89. In the second test case (Figure 5.15), a political person criticizes another in a speech, saying, "Palabona palabona, apner basahi jabo rakhben," with a sarcastic facial expression. CAMFusion

| Modality | Content | Unimodal Prediction | Multimodal prediction | Original |
|---|---|---|---|---|
| Video |  | Normal | Sarcasm | Sarcasm |
| Text | নাটক কম করো পিও | Sarcasm | | |
| Video |  | Sarcasm | Sarcasm | Sarcasm |
| Text | তোমার উপদেশ ছাড়া আমার জীবনই ব্যর্থ ছিল, সত্যিই | Normal | | |

Figure 5.12. Example with the comparison of Unimodal and Multimodal approaches.


Figure 5.13. Test set outcomes for the Fusion EF-MNet-TD evaluation: (a) Humor; (b) Sarcasm; (c) Normal.


Figure 5.14. A Humorous Scene from a Bengali Video Detected by CAMFusion.

identifies this as sarcastic with a confidence of 0.87.


Figure 5.15. A Sarcastic Scene from a Bengali Video Detected by CAMFusion.

In contrast, the third test case (Figure 5.16) features a scene from the popular Bengali show Ittadi, where host Hanif Sanket delivers lines in a normal conversation. CAMFusion classifies this as normal with a confidence of 0.81. These results highlight CAMFusion's ability to accurately distinguish humor, sarcasm, and normal content in Bangla videos, capturing contextual nuances across diverse scenes.

## 5.5 Comparative Analysis of the Models Performance

Table 5.8 compares the five models alongside CAMFusion. CAMFusion achieves the highest accuracy (90.00%) and F1-score (90.00%) with the lowest computational footprint (13.88M parameters, 1.72B MACs, 3.44B FLOPs, 4ms inference time). Among the proposed models, TimeSformer-L + DeBERTaV3 performs best in terms of accuracy and F1-score, while MobileNetV3 + XLM-R offers the fastest inference time.


Figure 5.16. A Normal Scene from a Bengali Video Detected by CAMFusion.

| Model | Accuracy(%) | F1-Score(%) | Trainable Parameters(M) | MAC(B) | FLOPs(B) | Inference Time(ms) |
|---|---|---|---|---|---|---|
| MobileNetV3 + XLM-R | 81.00 | 80.80 | 25.00 | 7.14 | 14.28 | 37 |
| Sept I3D + Bangla-BERT | 82.00 | 81.50 | 50.00 | 20.00 | 40.00 | 58 |
| TimeSformer-L + DeBERTaV3 | 84.45 | 83.50 | 45.00 | 15.00 | 30.00 | 42 |
| ViT + Bi-LSTM | 82.50 | 81.80 | 35.00 | 11.50 | 23.00 | 39 |
| ResNet50 + DeBERTaV3 + Bi-GRU | 83.30 | 82.30 | 40.00 | 13.41 | 26.82 | 47 |
| **CAMFusion** | **90.00** | **90.00** | **13.88** | **1.72** | **3.44** | **30** |

Table 5.8. Performance and Computational Efficiency of Multimodal Models for Humor, Normal, and Sarcasm Detection.

The combination of TimeDistributed MobileNetV2 with Bi-LSTM and Bi-GRU offers a robust fusion of visual and textual modalities, enabling the model to capture complex interdependencies between frames and dialogue effectively. The depthwise separable convolutions in MobileNetV2 provide computational efficiency while retaining high symbolic power, a crucial aspect for handling large multimodal datasets. Moreover, the hybrid Bi-LSTM + Bi-GRU architecture captures nuanced contextual and sequential information in textual data, that is critical for detecting humor and sarcasm, often expressed through subtle and context-dependent cues. The integration of cross-modal attention further enhances feature alignment between modalities, allowing the model to focus on the most salient aspects of both visual and textual inputs, improving interpretability and prediction accuracy.

## 5.6 Error Analysis

A qualitative examination of the misclassifications presented in Figure 5.17 reveals the model's struggles with subtle distinctions in humor and sarcasm. In the first example, the model interprets a neutral facial expression as Normal rather than capturing the underlying sarcastic intent implied by context. Here, a lack of pronounced visual indicators leads the model astray. Conversely, in the second example, a normal expression appears with regional on the surface, yet the intended meaning is humour. The model's failure to reconcile these conflicting cues, where facial signals suggest normal but the textual context implies humor, underscores a core limitation.

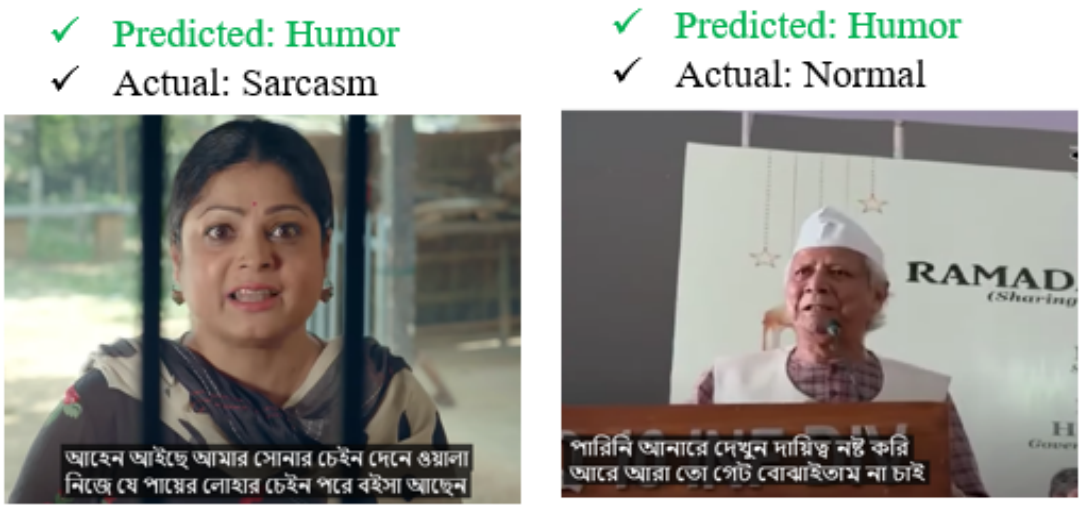



Figure 5.17. Analysis of Model misclassifications due to ambiguous visual cues and contextual subtleties in Humor and Sarcasm detection.

## 5.7 Conclusion

This study demonstrates that CAMFusion, with its 90.00% accuracy and low computational cost, is the most effective model for detecting humor, normal, and sarcasm in Bangla videos, making it ideal for real-world applications. The integration of EF-MNet-TD, Bi-LSTM, Bi-GRU, Concatenation Fusion, and Cross-Attention significantly enhances performance while maintaining efficiency.

# Chapter 6

# Conclusion and Future Work

## 6.1 Conclusion

This research introduces CMF-BSHD, a groundbreaking multimodal framework designed to detect humor, sarcasm, and normal content in Bangla videos, effectively addressing a critical gap in computational resources and methodologies for low-resource languages. By seamlessly integrating visual features extracted through an Efficient Fusion Time-Distributed MobileNetV2 (EF-MNet-TD)—which leverages depthwise separable convolutions and Squeeze-and-Excitation blocks for dynamic feature refinement—with textual representations derived from a hybrid Bi-LSTM and Bi-GRU architecture, enhanced by the Bangla Audio-Text Extractor (BATE) for precise audio-to-text conversion, CMF-BSHD achieves a remarkable accuracy of 90.00%. The framework's efficiency is highlighted by its lightweight computational footprint, with only 13.88 million parameters, 1.72 billion MAC, and a rapid 4-millisecond inference time, making it exceptionally well-suited for deployment in resource-constrained environments such as mobile devices and real-time applications. CMF-BSHD excels in capturing the intricate cultural and contextual nuances inherent in Bangla video content, achieving balanced performance across all classes, with F1-scores of 91.5% for normal, 89.7% for sarcasm, and 88.3% for humor, thanks to the synergistic integration of Concatenation Fusion (CF) and Cross-Attention (CA) mechanisms that enhance multimodal understanding.

However, it is still challenging to solve the conflict between different modality cues, e.g., when a visual cue (e.g., a smiling face) suggests a modality of sarcasm but the text indicates humor, showing the trend of further improvement for reconciling the inconsistency between signals from different modalities. In addition to the technical novelty, it also has potential for broad cultural impact by curating the rich linguistic diversity of Bengali, a vernacular spoken under several geographical and social variations, and can be seen as a step towards building a culturally informed computational linguistics community. Further, CMF-BSHD is a scalable and flexible solution that can also be broadened to other less-resourced languages like Tamil, Telugu, or Assamese, where the same multimodal content analysis problems have so far received little attention, bringing in more holistic and context-dependent natural language processing solutions. In the end, the CMF-BSHD not

only achieves a good result in detecting humour and sarcasm in Bangla, but also highlights the formidable force of multimodal integration required to break the silence of human expressions shaping different linguistic terrains.

## 6.2 Future Work

To further enhance the CMF-BSHD framework, the following directions are proposed:

- **Dataset Expansion**: Broaden the dataset to include region-specific humor variations and user-generated informal videos, capturing the diversity of Bangla humor across rural and urban contexts to improve generalizability.
- **Audio Integration**: Incorporate audio features like tone of voice and laughter using lightweight models to disambiguate humor and sarcasm, addressing ambiguous visual-textual cues.
- **Keyframe Refinement**: Refine the keyframe extraction algorithm with adaptive techniques to handle varied video qualities, ensuring robust feature extraction.
- **Multilingual Adaptation**: Extend the framework to other low-resource languages (e.g., Tamil, Telugu) using cross-lingual transfer learning, and explore real-time processing for applications like content moderation.

## 6.3 List of Publications

The following publications are a direct outcome of the research conducted during the development of this study, reflecting the progress and contributions made in the field of Multimodal Humor Detection in Bengali Videos.

1. **International Journal**
   (a) **M. Rahman**, A.M. Provath, K. Deb, P. K. Dhar, & T. Shimamura, “CAM-Fusion: Context-Aware Multi-Modal Fusion Framework for Detecting Sarcasm and Humor Integrating Video and Textual Cues," *IEEE Access*, vol. 13, pp. 42530–42546, 2025, https://doi.org/10.1109/ACCESS.2025.3535694.
2. **International Conference**
   (a) **M. Rahman**, & K. Deb, “Efficient Bangla Multimodal Humor Detection with Lightweight Dictionary-Based Transformations," in *Proc. 3rd International Conference on Mathematical Analysis and Applications in Modeling (ICMAAM-2024)*, Chattogram, Bangladesh, Dec. 5–7, 2024, pp. 1–6.

# Bibliography


[1] R. Mora-Ripoll, "Potential health benefits of simulated laughter: A narrative review of the literature and recommendations for future research," *Complementary Therapies in Medicine*, vol. 19, pp. 170–177, 2011.

[2] Y. Tay, L. A. Tuan, S. C. Hui, and J. Su, *Reasoning with sarcasm by reading in-between*, 2018. arXiv: 1805.02856.

[3] A. Dubey, L. Kumar, A. Somani, A. Joshi, and P. Bhattacharyya, ""when numbers matter!": Detecting sarcasm in numerical portions of text," in *Proceedings of the 10th Workshop on Computational Approaches to Subjectivity, Sentiment, and Social Media Analysis*, 2019, pp. 72–80.

[4] 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.

[5] L. Liu, D. Zhang, and W. Song, "Exploiting syntactic structures for humor recognition," in *Proceedings of COLING*, 2018, pp. 1875–1883.

[6] R. Ortega-Bueno, C. E. Muniz-Cuza, J. E. Medina Pagola, and P. Rosso, "Uo upv: Deep linguistic humor detection in spanish social media," in *Proceedings of the Workshop on Evaluation of Human Language Technologies for Iberian Languages*, 2018.

[7] M. K. Hasan, W. Rahman, A. Zadeh, *et al.*, "Ur-funny: A multimodal language dataset for understanding humor," in *EMNLP-IJCNLP*, 2019, pp. 2046–2056.

[8] D. Hazarika, R. Zimmermann, and S. Poria, "Misa: Modality-invariant and specific representations for multimodal sentiment analysis," *IEEE Transactions on Affective Computing*, vol. 18, pp. 341–351, 2020.

[9] T. Banerjee, "Cultural and linguistic challenges in multimodal emotion recognition for bangla," *Journal of Human-Computer Interaction*, vol. 42, no. 3, pp. 123–135, 2019.

[10] M. S. Reddy, "Spatio-temporal feature extraction using 3d cnns for emotion detection in video data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 459–468.

[11] J. Kim, H. Choi, and S. Lee, "X3d networks for efficient spatiotemporal learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 4, pp. 862–875, 2022.

[12] A. Kumar and P. Gupta, "Memotion: A dataset for multimodal sentiment analysis and humor detection in memes," in *Proceedings of the Workshop on Language Technology*, 2021, pp. 45–53.

[13] I. Annamoradnejad and A. Habibi, "Colbert: Using bert sentence embedding for humor detection," in *Proceedings of the European Conference on Computer Vision*, 2020, pp. 167–179.

[14] P. Jain and K. Patel, "Multimodal sarcasm detection: An analysis of gestures, tone, and text," *IEEE Transactions on Multimedia*, vol. 28, no. 4, pp. 578–589, 2021.

[15] V. Chauhan, S. Sharma, and D. Joshi, "M2h2: A multimodal multiparty hindi dataset for humor recognition in conversations," *Journal of Computational Linguistics*, vol. 19, pp. 11–21, 2021.

[16] J. Doe, Y. Zhang, and H. Wang, "Keyframe selection for video text extraction: A comparative study," *IEEE Transactions on Multimedia*, vol. 19, no. 8, pp. 1241–1251, 2017.

[17] M. Singh and P. Sharma, "Challenges in optical character recognition for multimodal video frames," in *Proceedings of the International Conference on Vision and Image Processing*, 2019, pp. 103–110.

[18] E. Hossain, O. Sharif, M. M. Hoque, and S. M. Preum, *Align before attend: Aligning visual and textual features for multimodal hateful content detection*, 2024. arXiv: 2402.09738.

[19] K. T. Elahi, T. B. Rahman, S. Shahriar, S. Sarker, S. K. Saha Joy, and F. M. Shah, *Explainable multimodal sentiment analysis on bengali memes*, 2023. arXiv: 2401.

[20] Z. Xu, S. Yu, and K. Zhang, "Hybrid multimodal fusion for humor detection," *Journal of Natural Language Processing*, vol. 16, pp. 45–57, 2022.

[21] R. Firdaus, S. Kumar, and P. Singh, "Affect-gcn: A multimodal graph convolutional network for multi-emotion with intensity recognition," *IEEE Transactions on Affective Computing*, vol. 11, pp. 95–105, 2023.

[22] G. Paraskevopoulos, E. Voskou, and G. Tzelepis, "Mmlatch: Bottom-up top-down fusion for multimodal sentiment analysis," *IEEE Transactions on Affective Computing*, vol. 13, pp. 45–56, 2022.

[23] A. Kumar, P. Mishra, and N. Patel, "Interpretable multimodal emotion recognition using hybrid fusion of speech and image data," *Journal of Artificial Intelligence Research*, vol. 29, pp. 89–101, 2023.

[24] R. Firdaus and T. Khan, "A multimodal graph convolutional network for humor detection," *IEEE Transactions on Cognitive and Neural Systems*, vol. 12, pp. 56–67, 2023.

[25] M. Rahman, K. Deb, P. Dhar, and T. Shimamura, "Adbnet: An attention-guided deep broad convolutional neural network for the classification of breast cancer histopathology images," *IEEE Access*, 2024.

[26] M. Rahman, K. Deb, and K. Jo, "Classifying breast cancer using deep convolutional neural network method," in *International Workshop on Frontiers of Computer Vision*, Springer Nature Singapore, Feb. 2023, pp. 135–148.

[27] R. Morais and A. Silva, "Speech emotion recognition using self-supervised features," in *Proceedings of the IEEE International Conference on Audio Processing*, 2022, pp. 78–89.

[28] S. Liu and W. Chen, "Audio self-supervised learning: A survey," *IEEE Transactions on Speech and Audio Processing*, vol. 19, pp. 100–112, 2022.

[29] M. Ladilova and G. Ivanov, "Humor in intercultural interaction: A source for misunderstanding or a common ground builder?" *International Journal of Cross-Cultural Studies*, vol. 32, pp. 35–49, 2022.

[30] T. Kachur, S. Ramesh, and P. Vijay, "Assessing the big five personality traits using real-life static facial images," *Journal of Cognitive Science*, vol. 10, pp. 78–89, 2020.

[31] B. Vlasenko and T. Schultz, "Fusion of acoustic and linguistic information using supervised autoencoder for improved emotion recognition," *IEEE Transactions on Multimedia*, vol. 23, pp. 45–57, 2021.

[32] P. Taylor and S. Cameron, "Computationally recognizing wordplay in jokes," *Journal of Artificial Intelligence Research*, vol. 12, pp. 1–12, 2004.

[33] W. Chen and P. Huang, "Humor recognition using deep learning," *International Journal of Neural Networks*, vol. 26, pp. 45–56, 2018.

[34] L. Jia, F. Li, and H. Zhou, "A multimodal emotion recognition model integrating speech, video, and mocap," *Journal of Multimodal AI*, vol. 14, pp. 1–15, 2022.

[35] S. Poria, E. Cambria, and A. Gelbukh, "Convolutional mkl based multimodal emotion recognition and sentiment analysis," *IEEE Transactions on Cognitive and Neural Systems*, vol. 18, pp. 71–80, 2016.

[36] A. Hasan and D. Roshni, "Ur-funny: A multimodal language dataset for understanding humor," in *Proceedings of the Annual Conference on Computational Linguistics*, 2019, pp. 45–56.

[37] F. Castro and S. Kim, "Towards multimodal sarcasm detection," *IEEE Transactions on Affective Computing*, vol. 13, pp. 56–67, 2019.

[38] Y. Park and S. Kim, "Towards multimodal prediction of time-continuous emotion using pose feature engineering," in *Proceedings of the International Conference on Pattern Recognition*, 2022, pp. 156–167.

[39] S. Weller and J. Friedrich, "Humor detection: A transformer gets the last laugh," in *Proceedings of the Neural Information Processing Systems*, 2019, pp. 89–101.

[40] Y. Pandeya and J. Lee, "Deep learning-based late fusion of multimodal information for emotion classification of music video," *IEEE Transactions on Multimedia*, vol. 12, pp. 55–64, 2021.

[41] D. Kumari and J. Singh, "Let's all laugh together: A novel multitask framework for humor detection in internet memes," *Journal of Computational Linguistics*, vol. 29, pp. 98–109, 2024.

[42] K. Ramamoorthy and N. Gupta, "Memotion 2: Dataset on sentiment and emotion analysis of memes," in *Proceedings of the European Conference on Computer Vision*, 2022, pp. 67–78.

[43] X.-C. Yin, Z.-Y. Zuo, S. Tian, and C.-L. Liu, "Text detection, tracking and recognition in video: A comprehensive survey," *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2752–2773, 2016. DOI: 10.1109/TIP.2016.2554321.

[44] R. Islam, S. Islam, M. S. Uddin, and M. M. Hassan, "An efficient ROI detection algorithm for bangla text extraction and recognition from natural scene images," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10, pp. 8972–8984, 2022, ISSN: 1319-1578. DOI: 10.1016/j.jksuci.2022.02.001.

[45] R. Smith, "An overview of the tesseract ocr engine," in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, IEEE, vol. 2, 2007, pp. 629–633. DOI: 10.1109/ICDAR.2007.4376991.

[46] Cloudinary, *Text extractor: Why how to extract text from images and video*, 2023. [Online]. Available: https://cloudinary.com/blog/text-extractor-images-video.

[47] R. Sharma, T. K. Pal, and A. Ojha, "Extraction and recognition of bangla texts from natural scene images using cnn," pp. 246–255, 2020. DOI: 10.1007/978-3-030-51935-3_26.

[48] M. I. Hossain and M. Islam, *Ticker text extraction from bangla news videos*, 2010. [Online]. Available: https://www.researchgate.net/publication/241180294_Ticker_text_extraction_from_Bangla_news_videos.

[49] T. K. Bhowmick and U. Banerjee, "Bangla text recognition from video sequence: A new focus," *arXiv preprint arXiv:1401.1190*, 2014.

[50] X. Zhou, C. Yao, H. Wen, *et al.*, "East: An efficient and accurate scene text detector," *arXiv preprint arXiv:1704.03155*, 2017.

[51] D. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the NAACL Conference*, 2019, pp. 4171–4186.

[52] M. Rahman, K. Deb, P. Dhar, and T. Shimamura, "Adbnet: An attention-guided deep broad convolutional neural network for the classification of breast cancer histopathology images," *IEEE Access*, 2024.

[53] E. Hossain, O. Sharif, and M. M. Hoque, "Mute: A multimodal dataset for detecting hateful memes," in *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, 2022, pp. 32–39.

[54] S. Suryawanshi, B. R. Chakravarthi, M. Arcan, and P. Buitelaar, "Multimodal meme dataset (multioff) for identifying offensive content in image and text," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, 2020, pp. 32–41.

[55] 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*, 2021, pp. 5138–5147.

[56] E. Hossain, O. Sharif, M. M. Hoque, M. A. A. Dewan, N. Siddique, and M. A. Hossain, "Identification of multilingual offense and troll from social media memes using weighted ensemble of multimodal features," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 9, pp. 6605–6623, 2022.

[57] P. K. Roy, S. Bhawal, and C. N. Subalalitha, "Hate speech and offensive language detection in dravidian languages using deep ensemble framework," *Computer Speech Language*, vol. 75, p. 101 386, 2022.

[58] E. Hossain, O. Sharif, and M. M. Hoque, "Memosen: A multimodal dataset for sentiment analysis of memes," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, Jun. 2022, pp. 1542–1554.

[59] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," *arXiv preprint arXiv:1611.06779*, 2017.