Weighted Attention in Multimodal Transformers for the Detection of Questionable Content in Videos

By:

Arnold Morales Morales

Thesis submitted as a partial requirement to obtain the degree of:

MASTER OF SCIENCE IN THE AREA OF COMPUTER SCIENCE

in the

Instituto Nacional de Astrofísica

Óptica y Electrónica

March, 2024

Tonantzintla, Puebla

Directed by:

Dr. Hugo Jair Escalante Balderas

Table of Contents

List of Figures

List of Tables

Acknowledgments

This thesis owes its existence to the support extended by the Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT), under Grant No. 804969, and project CB-S-26314, which played an integral role in facilitating this thesis's execution.

Furthermore, my heartfelt appreciation extends to my advisor, Dr. Hugo J. Escalante Balderas, whose unwavering support, dedication, and expertise were instrumental in ensuring the successful completion of this work.

Gratitude is also extended to the entire team at the Instituto Nacional de Astrofísica Óptica y Electrónica, including its dedicated staff and esteemed professors, whose guidance and knowledge have been invaluable in shaping my academic journey.

Abstract

We address the problem of questionable content filtering in video platforms, with a specific focus on identifying and flagging comic mischief. These contents mix elements such as violence, adult content or sarcasm with humor, which complicates their detection. Current methodologies rely heavily on attention-based models, prominently featuring Hierarchical Cross-Attention (HCA) to fuse information across different modalities. While HCA has proven to be effective, its optimal applicability in this context remains uncertain. This work explores an innovative approach termed Parallel Cross-Attention (ParCA) as an alternative mechanism for enhancing model in identifying nuanced forms of comic mischief.

Furthermore, we advocate for the integration of Gated Multimodal Units (GMU) into the framework. GMUs offer a refined method for combining multiple attention mechanisms, surpassing the traditional concatenation approach by dynamically adjusting the importance of modalities at various stages of processing. This hybrid approach promises to improve the interpretability and performance of the model in discerning subtle comic elements amidst diverse multimedia content.

Our experimental results substantiate the efficacy of ParCA and GMU integration, revealing substantial performance enhancements compared to the HCA-based baseline. Specifically, our approach achieves notable improvements in F1-Score met-

ric, demonstrating its capacity to effectively filter and flag comic mischief in video content. This research underscores the importance of innovative model architectures and multimodal fusion techniques in advancing content filtering capabilities for evolving digital platforms.

Resumen

Abordamos el problema del filtrado de contenido cuestionable en plataformas de video, con un enfoque específico en la identificación y señalización de diferentes tipos de comedia. Las metodologías actuales dependen en gran medida de modelos basados en atención, destacando el uso de la Atención Cruzada Jerárquica (HCA) para fusionar información a través de diferentes modalidades. Aunque HCA ha demostrado ser eficaz, su aplicabilidad óptima en este contexto sigue siendo incierta. Este trabajo explora un enfoque innovador denominado Atención Cruzada Paralela (ParCA) como un mecanismo alternativo para mejorar el modelo en la identificación de formas sutiles de travesuras cómicas.

Además, abogamos por la integración de Unidades Multimodales por Compuertas (GMU) en el marco de trabajo. Las GMU ofrecen un método refinado para combinar múltiples mecanismos de atención, superando el enfoque tradicional de concatenación al ajustar dinámicamente la importancia de las modalidades en diversas etapas del procesamiento. Este enfoque híbrido promete mejorar la interpretabilidad y el rendimiento del modelo en la detección de elementos cómicos sutiles en medio de contenido multimedia diverso.

Los resultados experimentales obtenidos confirman la eficacia de la integración de ParCA y GMU, revelando mejoras importantes en el rendimiento en compara-

ción con el modelo original basado en HCA. Específicamente, nuestro enfoque logra mejoras notables en F1-Score, demostrando su capacidad para filtrar y señalar efectivamente los tipos de comedia en contenido de video. Este trabajo enfatiza la importancia de arquitecturas de modelos innovadoras y técnicas de fusión multimodal en el avance de las capacidades de filtrado de contenido para plataformas digitales en evolución.

CHAPTER 1

INTRODUCTION

The issue of detecting objectionable content in videos arises in the context of the exponential growth in the generation and consumption of multimedia content on online platforms. With the ease of video production and sharing, there has been growing concern about the presence of inappropriate, violent and hateful content that can negatively affect users (Huesmann, 2007).

Detecting questionable content in videos has become crucial to ensure the safety and positive experience of users on online platforms. This type of content can include violent material, hate speech, extremist propaganda and other content that violates community policies and standards. Exposure to this type of content can cause psychological harm, foster misinformation, contribute to the spread of hate speech, and undermine trust in digital platforms (Solorio et al., 2021).

The problem presents unique challenges that cannot be effectively addressed with existing methods that focus solely on one type of modality, either text analysis or image processing. The complex and dynamic nature of videos requires a multimodal approach that combines information from multiple sources, such as visual content, audio content, and textual context. For example, the exclusive use of text analysis may miss key visual or auditory cues for the detection of scenes with explicit violence or hate speech, because avoidance techniques may be employed using

figurative language, slang, or coded expressions that are difficult to detect using a single modality.

By combining multiple modalities in a multimodal approach, complementary signals can be captured and used to improve the detection of questionable content in videos. Joint analysis of visual and auditory content enables a more complete and accurate understanding of video context and content. For example, detecting violent gestures or movements in visual content, combined with hate speech in audio content, can provide a stronger indication of the presence of questionable content.

In this work, we approach the problem of detecting questionable content in video with Multimodal Attention Based Models (MABMs). Identifying content with clear labels offers a more flexible solution than traditional age classification systems, which commonly use age-based rating systems, such as those by the Motion Picture Association of America (Baharlouei and Solorio, 2024). This is because people's tolerance for questionable content varies based on their age, life experience, socio-cultural values, and cognitive abilities (Anderson et al., 2003). This work focuses on the detection of comic mischief content in videos, a subset of questionable content. In these videos, the problematic content (such as violence, adult material, or sarcasm) is presented in a humorous context, making it even more disturbing. Baharlouei and Solorio (2024) have recently shown that MABMs comprise an effective solution to this task. In particular, the authors showed that a single-head Hierarchical-Cross-Attention (HCA) based model effectively leverages multimodal information for predicting comic mischief. While effective, it is unclear whether fusion mechanisms alternative to HCA can perform better for this task. Likewise, the use of multiple heads in such MABM has not been explored, this method is ideal for capturing complex multimodal dependencies. Using multiple heads allows the model to pay attention to different aspects of the inputs simultaneously, improving the understanding of interactions between the different modalities.

Due to the nature of the task, humor categories are not mutually exclusive. The same fragment may contain sarcasm and mature humor, which requires models capable of dealing with the information and that a category may be treated better with one modality than with the other. Accordingly, in this paper, we study the effectiveness of a novel *parallel cross-attention* (ParCA) mechanism to combine multimodal information in MAMBs. Additionally, we explore a new way to merge multiple multimodal care mechanisms using multimodal triggered units (GMU) (Arevalo et al., 2020), exploiting this fusion method at the attention module level to capture the dependencies between modalities and obtain a better representation of each. Moreover, the performance of these models is explored with single-headed and multi-headed variants of MABMs, to explore the different representaions as mentioned before.

Summarizing, the main contributions of this work are as follows:

- A comparative study of HCA and the proposed ParCA mechanisms for learning multimodal cross-attention for comic mischief detection.
- A new way to combine information from multiple multimodal attention mechanisms based on GMU for comic mischief detection.
- An experimental evaluation of several variants of MAMBs in terms of attent, number of heads, and fusion schemas.
- A new way to explore different fusion types for multimodal information using transformer-based model.
- A comparative study with different tasks an datasets with multimodal learning for questionable content detection.

1.1 Problem Statement

Developing accurate models for identifying questionable content presents a substantial challenge in machine learning. Challenges arise from both the complexity of the task and the scarcity of labeled data available, compounded by the subjective nature of such content and its cultural diversity. Despite the growing availability of multimodal datasets, many lack the volume necessary to adequately train transformer models, which have shown potential in diverse applications (Lin et al., 2021). Previous research employing transformer architectures in this domain has not thoroughly explored effective methods for integrating different data types.

This thesis aims to develop a tailored machine learning model for detecting questionable content, addressing the following issues:

- Detecting questionable content in videos can be especially challenging due to factors like sarcasm, use of language with multiple meanings, and subtle hints, which traditional methods may struggle to identify. These intricacies demand a deeper comprehension of context and semantics. Hence, instead of analyzing modalities separately, integrating them could enhance the capacity of the model to recognize and classify such content accurately.

- Traditional transformer-based models and attention-based models typically use a common cross-modal attention mechanism to capture relationships between different modalities. This approach is effective for understanding connections and correlations across the data from different sources, such as text and images. However, most of these models are limited to considering only two modalities. This limitation can pose challenges when dealing with more than two modalities, as the models may struggle to integrate information from multiple sources effectively.

Comparing various forms of questionable content through Transformer archi-

tectures may provide valuable insights into the influence of different modalities on such tasks, particularly in scenarios where data availability is limited. Exploring these comparisons could shed light on how factors like image, text, and audio contribute to the overall understanding and detection of questionable content. Furthermore, understanding the interplay between these modalities can offer crucial guidance for developing more robust and effective detection systems, particularly in contexts where data resources are constrained.

1.2 Motivation

Facing the task of questionable content detection using transformer architectures presents an exciting opportunity to delve into the complexities of content detection in the digital age. This effort is not only intellectually stimulating but also holds immense practical significance in enhancing the accuracy and reliability of content moderation systems across diverse platforms.

Moreover, by grappling with the scarcity of data in this domain, we are compelled to innovate and devise novel methodologies that push the boundaries of machine learning research. Tackling this task requires creativity, perseverance, and a willingness to confront the inherent uncertainties and ambiguities associated with detecting nuanced forms of questionable content.

In this context, the use of Gated Multimodal Units (GMU) and multi-headed architectures can be decisive for the detection of questionable content, since GMUs efficiently integrate different data modalities by prioritizing the relevant information in each case, while multiple heads allow capturing different aspects of the content simultaneously, thus improving the accuracy and the model's ability to recognize complex patterns in data-limited contexts.

1.3 Objectives

In this section we present the general and specific objectives of this thesis.

1.3.1 General Objective

The goal of this work is to develop and evaluate a transformer-based model for detecting questionable content in videos. The aim is to enhance the precision and efficiency of content moderation processes.

1.3.2 Specific Objectives

The specific objectives are as follows:

- Selecting a multimodal transformer model that uses a cross-attention mechanism to integrate more than two modalities, enhancing the ability of the model to combine and understand diverse data types.
- Extracting and encoding essential features, and using representation learning to capture valuable representations of different modalities.
- Designing and developing a weighting or fusion scheme within existing attention mechanisms, optimizing the model's performance by appropriately combining attention weights from different modalities.

1.4 Scope and Limitations

This thesis aims to introduce new approaches for improving the performance of existing models in questionable content detection. The focus is on creating an innovative

model for questionable content detection and to present new insights and suggestions for refining the performance of current models. By addressing these issues, this study aims to advance the field of machine learning and develop more reliable and efficient models for questionable content detection.

It is important to acknowledge that this study will face certain limitations due to factors such as cultural differences, subjective interpretation of questionable content, and a lack of data, which is a significant challenge. Moreover, the effectiveness of the proposed methods and strategies may be affected by the volume of available data. Another limitation arises from potential biases in the training and evaluation data. Therefore, it is essential to thoroughly evaluate the performance of the model across various datasets and consider the all possible contexts in which they will be applied.

1.5 Published Articles

The publications derived from this thesis are listed below.

- Morales, A., Baharlouei, E., Solorio, T., & Escalante, H. J. (2024, June). Multimodal-Attention Fusion for the Detection of Questionable Content in Videos. In *Mexican Conference on Pattern Recognition* (pp. 188-199).
- Morales, A., Baharlouei, E., Solorio, T., & Escalante, H. J. (2024, June). On the use of Multimodal Attention for Questionable Content Detection in Videos. In *LXAI NAACL Workshop of 2024*.

1.6 Thesis Organization

This thesis is organized as follows:

- Chapter 2: Background: This chapter introduces some definition of transformers and key concepts related to this topic, to aid in comprehending the proposed solution.

- Chapter 3: Related Work: In this chapter existing state-of-art related to questionable content detection, and fusing modalities are reviewed. Including an overview of different approaches employed.

- Chapter 4: Methodology: In this chapter, the proposed method for the detection of questionable content in presented in detail. The structure of the model and the dataset we used for classification.

- Chapter 5: Experiments: Here, we explain the expreminents we carried out to evaluate the performance of the model, an analysis of the results and the experimental setup.

- Chapter 6: Conclusions and Future Work: This chapter details and summarizes the main contributions of the presented work, the limitations of the presented approach and what could be the future research.

CHAPTER 2

BACKGROUND

The aim of this chapter is to provide a concise overview of the key concepts essential for understanding the ensuing discussions. First, Section 2.1 deals with "questionable content" and clarifies the various forms and characteristics that define it as such. Section 2.2 reviews transformers. A central point is the explanation of how transformers work. Section 2.3 addresses different ways of handling multimodal inputs, exploring various strategies for effectively merging these inputs.

2.1 Questionable Content

This section aims to explain the concept of questionable content, delving into its definition and scope. The goal is to assist readers in comprehending the significance of these terms within their respective domains.

2.1.1 What is Questionable Content?

There is not a single, universally accepted formal definition of “questionable content” provided by a specific institution. The interpretation of what constitutes questionable content often varies across different platforms, cultures, and societies. Ques-

tionable content typically refers to material, such as text, images, or videos, that raises doubts about its appropriateness, accuracy, or ethical standards (Weidinger et al., 2021). It can encompass a wide range of content that may be considered controversial, offensive, or unreliable. The evaluation of what is questionable often depends on cultural, social, and individual perspectives.

Questionable content encompasses politically or ideologically motivated online disinformation, fake news, hate speech, online misinformation, misreporting, and misconstrued satire, potentially influencing individuals and the population collectively by shaping consumer attitudes (Chang et al., 2021).

2.1.2 Questionable Content in Videos

With the widespread availability of online platforms, individuals across various age groups increasingly engage in online content consumption as a prevalent form of entertainment. The content featured on these platforms may frequently contain material that parents deem inappropriate for their children. Furthermore, researchers in psychology have conducted studies revealing the potential adverse impacts of certain online videos on young viewers (Wilson, 2008; Chang and Bushman, 2019; Bridges et al., 2010; Bushman and Anderson, 2009; Dillon and Bushman, 2017).

The prevalent types of content often categorized as "questionable" within these formats encompass *Violence, Hate Speech*, and *Sexual Themes*—whether implicit or explicit. Nevertheless, it's essential to recognize that a diverse array of content falls under the umbrella of "questionable" across the various modalities offered by videos (Solorio et al., 2021). This broad classification underscores the importance of considering a spectrum of factors when evaluating content, including cultural nuances, context, and evolving societal standards, Figure 2.1 shows an example of this content.

Figure 2.1: A scene from YouTube Kids video showing Mickey Mouse in a pool of blood while Minnie Mouse looks on, an example of the type of implied violence content (Maheshwari, 2017).

2.1.3 Comic Mischief

Comic mischief content can be difficult to define clearly due to its ambiguous nature. What is considered mischievous and humorous in one context may be seen as inappropriate or offensive in another. The subjectivity of humor and cultural context play a crucial role in interpretation, complicating the creation of accurate models. To address the subjectivity of the task, each instance was reviewed by three evaluators and the final label assigned to each segment is determined by the majority of votes among the annotators. To measure the quality of the annotations, the Inter-Annotator Agreement (IAA) using Cohen's Kappa (κ) was calculated, comparing the annotations of each evaluator with the majority vote. According to the IAA values obtained, substantial agreement was found ($\kappa = 0.70$).

Detecting the appropriate context in which the comic mischief content occurs is essential for models, especially as they struggle to differentiate between serious and funny content. In addition, videos contain images, sound and text, so it is necessary

Figure 2.2: A scene from an animated program, in which the duck is blown up to get a laugh and win the show.

to pay attention to all these modalities to capture the mood (Yang et al., 2023). Figure 2.2 shows an example of this type of content.

Providing accurate classification and clear labels allows users to choose content that aligns with their personal preferences, thus enhancing their experience in virtual environments. Ensuring content that does not cause harm or discomfort to your users. Detecting and correctly classifying comic mischief content ensures that the sensitivities of different audiences are properly managed.

2.2 Transformers

This section gives a look at how transformers are set up in the common encoder and decoder architectures and how the attention mechanism works.

2.2.1 Attention Mechanism

The first attention mechanism introduced by Bahdanau et al. (2014), revolutionized neural machine translation by introducing a dynamic attentional mechanism. Traditional sequence-to-sequence models often struggled with fixed-size context vectors, especially when dealing with lengthy input sequences. Bahdanau attention addressed this limitation by allowing the model to adaptively focus on different parts of the in-

put sequence during decoding. At each step, attention weights are computed based on the alignment between the current decoding position and each element in the input sequence. In this context, the attention mechanism gained notable attention for its effectiveness in addressing alignment problems and handling variable-length input and output sequences.

2.2.2 Attention in Transformers

The attention mechanism in transformers was introduced by Vaswani et al. (2017). The key idea is to allow the model to weigh the importance of different parts of the input sequence when producing each element of the output sequence as shown in Figure 2.3. This attention mechanism is self-attention or scaled dot-product attention.

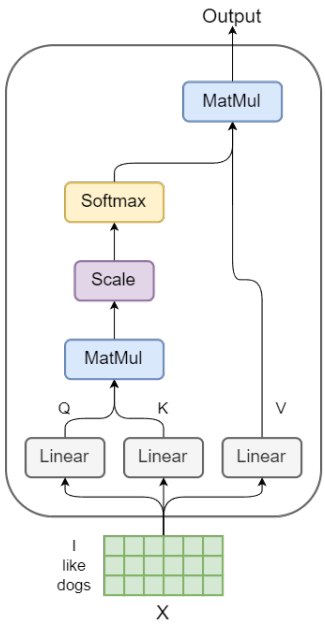


Figure 2.3: Scaled Dot-Product attention from Vaswani et al. (2017) transformer architecture.

The scaled dot-product attention (see Figure 2.3) uses three type of vectors as an input. The queries, Q, the keys, K, and the values, V. Each of them are linear

projections, Queries are used to determine how much attention each element in the sequence should pay to the others. In the context of transformers, each element in the input sequence is associated with a query, key and value vector; Keys are used to represent the relationships between different elements. The attention mechanism computes the similarity or affinity between the queries and keys to determine the attention weights; Values are the vectors that will be combined based on the attention weights to produce the output. The scaled dot-product attention is explained in detail below.

1. Given an input sequence $X \coloneqq [x_1, \ldots, x_N]^\top \in \mathbb{R}^{N \times D_x}$ of N feature vectors, is transformed into Q, K and V matrices via linear transformations, Eq 2.1.

$$Q = XW_Q^\top;\ K = XW_K^\top;\ V = XW_V^\top \tag{2.1}$$

where $W_Q,\ W_K \in \mathbb{R}^{D_k \times D_x}$, and $W_V \in \mathbb{R}^{D_v \times D_x}$ are the weight matrices. Denoting $Q \coloneqq [q_1, \ldots, q_N]^\top$, $K \coloneqq [k_1, \ldots, k_N]^\top$, $V \coloneqq [v_1, \ldots, v_N]^\top$ and $q_i,\ k_i,\ v_i$ are the query, key and value vectors, respectively, see Figure 2.4a for an illustrative form.

2. The attention mechanism calculates attention scores, Equation 2.2, by taking the dot product of Q with $K^\top$. Each element of the resulting matrix represents the attention score between a pair of elements in the sequence.

$$\text{Attention Scores} = QK^\top \tag{2.2}$$

3. To stabilize the learning process, the attention scores in Equation 2.2 are scaled by the square root of the dimension of the key vectors, Equation 2.3. This scaling factor, often denoted as $\frac{1}{\sqrt{D_k}}$, is the dimension of the key vectors, prevents the gradients from becoming too small or too large during training.

$$\text{Scaled Attention Scores} = \frac{QK^\top}{\sqrt{D_k}} \tag{2.3}$$

4. The scaled attention scores in Equation 2.3 are passed through the softmax function to obtain normalized weights, see 2.4. The softmax operation converts the

scores into probabilities, ensuring that the weights of each vector sum to 1. Finally, the attention weights are then used to compute a weighted sum of the value vectors (V) as in Equation 2.5 and Figure 2.4b. This weighted sum represents the context vector for each position in the sequence.

$$\text{Attention Weights} = \text{softmax}\left(\frac{QK^\top}{\sqrt{D_k}}\right) \tag{2.4}$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{D_k}}\right)V \tag{2.5}$$

In the vector form, Equation 2.5 can be written as follows:

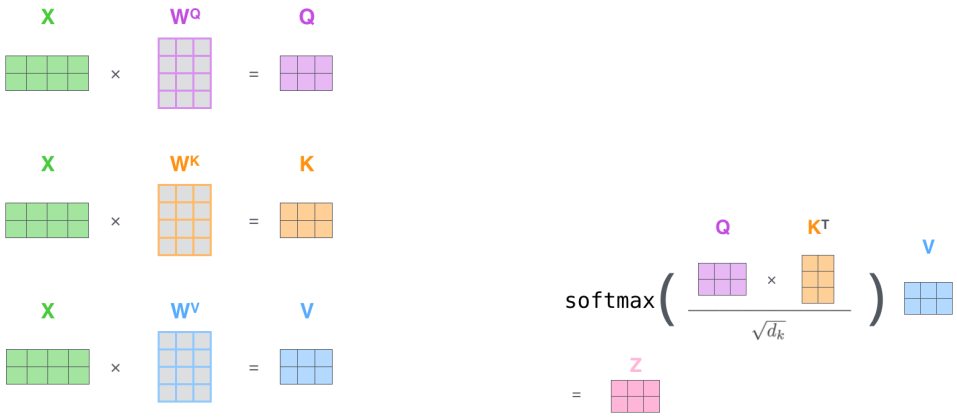
$$\text{Attention}_i(Q, K, V) = \sum_{j=1}^{N} \text{softmax}\left(\frac{q_i \cdot k_j}{\sqrt{D_k}}\right) v_j \tag{2.6}$$

The key innovation here is the dynamic nature of attention. Instead of relying on a fixed context for the entire decoding process, the model can selectively attend to different parts of the input sequence based on their relevance to the current decoding step. This flexibility allows the model to handle long-range dependencies more effectively, making it particularly beneficial for tasks like machine translation where the alignment between words in the source and target languages is crucial.

Scaled Dot-Product Attention is pivotal in neural network architectures, particularly exemplified in models like transformers, for several compelling reasons. Its ability to capture long-range dependencies stands out, allowing the model to discern relationships between elements across the entire sequence(Bahdanau et al., 2014; Vaswani et al., 2017; Kim et al., 2016), a feat challenging for traditional sequential architectures.

2.2.3 Different types of Attention

There are several types of attention mechanisms used in the models. The objective of these attention mechanisms cater to different requirements in different tasks and



(a) Calculation of Query, Key and Value matrices from step 1.

(b) Scaled Dot-Product Attention calculation in matrix form.

Figure 2.4: The Scaled Dot-Product Attention, an illustrative way to understand it. (Alammar, 2018b).

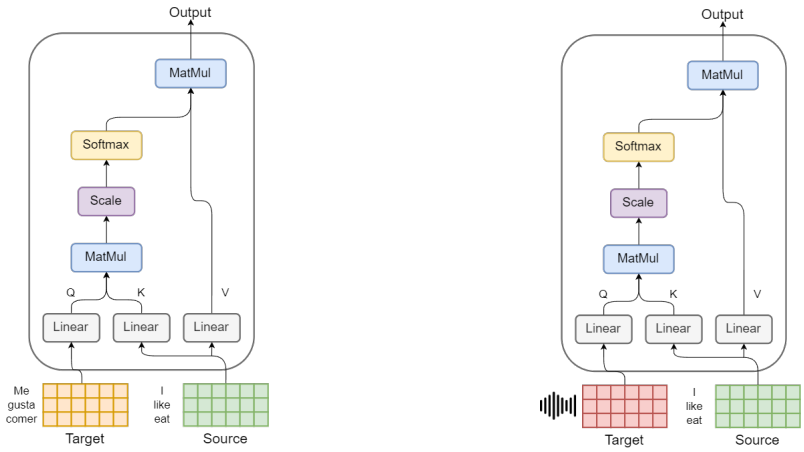
contribute to the adaptability and effectiveness of transformers models in capturing dependencies within sequential data. There are three which are the most common explained below.

Scaled Dot-Product Attention

This method, also called Self-Attention, as explained in Sec. 2.2.2, takes the same source vectors X and transform them into queries, keys and values matrices. The key idea behind the attention mechanism in transformers is to allow the model to weigh the importance of different parts of the input sequence when producing each element of the output sequence, see Figure 2.3. Self-attention mechanisms have proven crucial in various natural language processing (NLP) and sequence-based tasks such as text summarization task (See et al., 2017), question answering (Kenton and Toutanova, 2019), language modelling (OpenAI et al., 2023; Kenton and Toutanova, 2019) and machine translation (Lewis et al., 2020; Vaswani et al., 2018)

Cross-Attention

Cross-attention is a specialized mechanism employed in sequence-to-sequence models, notably in tasks like machine translation but recently also used in multimodal tasks. Unlike self-attention where each position in the sequence attends to itself, cross-attention enables the model to selectively focus on relevant segments of the source sequence while generating individual elements of the target sequence, Figure 2.5 shows a scheme of this mechanism. This mechanism enhances the capacity of the model to align and capture dependencies between the input and output sequences. As the decoder progresses in generating the target sequence, it employs both self-attention to consider previously generated elements and cross-attention to attend to distinct parts of the source sequence. This dynamic attention to relevant information during the decoding process contributes to the improved performance of the model in tasks requiring a comprehensive understanding of the global context, such as machine translation (Vaswani et al., 2017) but also in multimodal tasks (Sun et al., 2019; Tan and Bansal, 2019; Li et al., 2019; Lu et al., 2019)



(a) Cross-Attention calculation for the original machine translation task.

(b) Cross-Attention calculation for multimodal tasks (audio and text).

Figure 2.5: Cross-Attention mechanism for translation task and for multimodal tasks.

Multihead-Attention

Multi-head attention, as shown in Figure 2.6, is a variant of the attention mechanism in transformer architectures that enhances the ability of the model to capture diverse relationships within input sequences simultaneously. In this mechanism, input vectors, including queries, keys, and values, undergo multiple linear projections using different learned weight matrices for each attention head (Equation 2.5). Each attention head independently computes attention scores, applies softmax, and generates context vectors. The resulting context vectors from all attention heads are concatenated and linearly projected to produce the final output, see Equation 2.7. By allowing the model to attend to different parts of the input sequence in parallel, multi-head attention promotes the learning of nuanced patterns and dependencies, contributing to the expressive power and effectiveness of transformer models in various natural language processing tasks.

$$\text{MultiHead}\left(\{Q, K, V\}_{i=1}^{H}\right) = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_H\right) W^O \tag{2.7}$$

Where H is the number of heads, $W^O \in \mathbb{R}^{HD_v \times HD_v}$ is the projection matrix and $\text{head}_i = \text{Attention}(Q, K, V)$ is the self-attention.

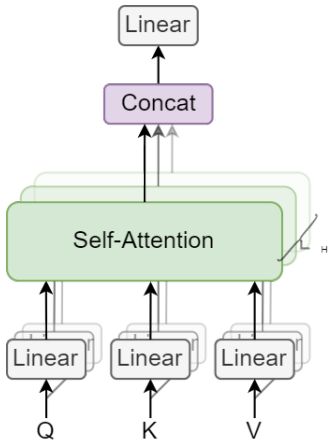


Figure 2.6: An illustration of the multihead-attention mechanism.

2.2.4 Transformer Architecture

The Transformer architecture, first introduced by Vaswani et al. (2017), revolutionized neural network-based sequence-to-sequence tasks. Its core innovation lies in the self-attention mechanism (Sec. 2.2.2), which enables the model to weigh different elements of an input sequence dynamically, capturing long-range dependencies efficiently. Comprising an encoder-decoder structure, the Transformer employs multiple layers, each featuring multi-head self-attention and feedforward sub-layers. The encoder processes input sequences in parallel, while the decoder, with additional masked self-attention and encoder-decoder attention mechanisms, generates output sequences. Residual connections and layer normalization stabilize training. Widely adopted in natural language processing and beyond, the Transformer architecture's parallelization and ability to capture intricate relationships between sequence elements have made it a cornerstone in various machine learning applications. Even though the Vanswani transformer is based on encoder-decoder architecture, there are some that are based only in encoder or decoder architectures. Each one of them explained below.

Transformer Encoder

The encoder, Figure 2.7, is responsible for processing the input sequence and extracting relevant features, which are then used by the decoder for generating the output sequence. The encoder consists of multiple identical layers, and each layer has two main sub-layers: the multi-head self-attention (MHSA) mechanism and the position-wise feedforward network (PFFN).

To account for the sequential nature of the input, positional encodings are added to the input embeddings before feeding them into the encoder. These encodings provide information about the position of each token in the sequence, allowing the model to understand the order of the elements. There are many choices of

positional encodings, learned and fixed (Gehring et al., 2017). Equation 2.8 and Equation 2.9 shows the fixed positional encodings.

$$PE(\text{pos}, 2i) = \sin\left(\frac{\text{pos}}{1000^{\frac{2i}{d_{\text{model}}}}}\right) \tag{2.8}$$

$$PE(\text{pos}, 2i+1) = \cos\left(\frac{\text{pos}}{1000^{\frac{2i}{d_{\text{model}}}}}\right) \tag{2.9}$$

where "pos" is the position, i is the dimension and $d_{\text{model}} = D_x$ is the same dimension of the source embeddings.

The next step after positional encoding is to apply multi-head self-attention sub-layer, as explained in Sec. 2.2.3, this sub-layer involves running the self-attention mechanism multiple times in parallel, each with different learned linear projections. After this, the ouput passes through a position-wise feedforward network. This network consists of fully connected layers applied independently to each position in the sequence. It helps capture complex, non-linear relationships between elements in the sequence.

Layer normalization (Ba et al., 2016) is applied after each sub-layer, and a residual connection (He et al., 2016) is used to add the input of the sub-layer to its output. This helps stabilize training by preventing the vanishing gradient problem and allows the model to learn identity mappings, facilitating the flow of information through the network.

Transformer Decoder

The decoder is responsible for generating an output sequence. It also consists of multiple identical layers, each with three main sub-layers: the masked multi-head self-attention mechanism, the multi-head encoder-decoder attention mechanism, and the position-wise feedforward network.

Similar to the encoder, the decoder employs a masked MHSA mechanism, it

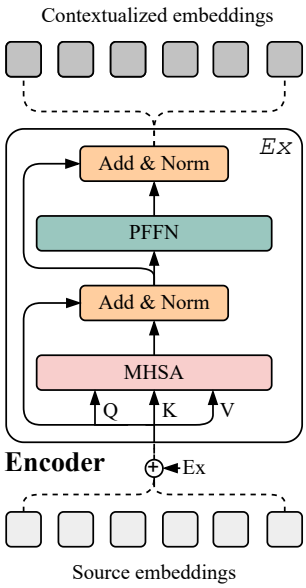


Figure 2.7: Encoder transformer used in the original transformer architecture, (Vaswani et al., 2017).

uses a mask to prevent attending to future positions in the sequence. This ensures that each position can only attend to its preceding positions, avoiding information leakage from the future. The latter two-sublayers are the same for both, encoder and decoder mechanisms, and layer normalization and residual connections are applied after each sub-layer. Figure 2.8 shows a visual representation of decoder mechanism.

2.2.5 Main appplications

Architectures that only use encoders or decoders typically serve different purposes and are designed for specific types of tasks.

Encoder-only architectures are often used for tasks where the input is processed, and the goal is to extract meaningful representations or embeddings of the input data. These representations can be used for downstream tasks such as classification, clustering, or feature extraction. While Decoder-only architectures are less

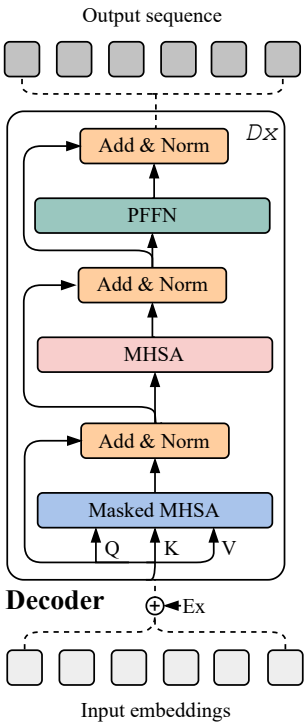


Figure 2.8: Transformer decoder architecture featuring a Masked MHSA module.

common but can be found in certain generative models. The primary purpose is to generate output sequences or samples based on learned representations or conditions. Autoregressive models, such as language models like GPT (Generative Pre-trained Transformer) (Brown et al., 2020; OpenAI et al., 2023), primarily use decoders. They generate sequences one element at a time, with each element depending on the previously generated ones.

The principal tasks of these two architectures are image or speech recognition and text classification for Encoder-only, where meaningful representations are crucial. Decoder-only architectures are prevalent in tasks like text generation, machine translation, and image synthesis.

It is worth noting that many successful models, especially in natural language processing, use both encoders and decoders in an encoder-decoder architecture. This design is powerful for tasks that involve both understanding and generating se-

quences, such as machine translation or summarization.

2.3 Multimodal Learning

In recent times, numerous transformers have undergone thorough examination for diverse multimodal tasks, demonstrating compatibility with various modalities in both discriminative and generative tasks. This section will delve into an exploration of the fundamental techniques and designs employed in existing multimodal transformer models, specifically focusing on self-attention variants.

2.3.1 Gated Multimodal Unit (GMU)

The GMU module, see Figure 2.9, proposed by Arevalo et al. (2020) allows the model to learn different representation by combining different inputs or modalities, where the module learn to decide the contribution of each input, focusing only in the most relevant aspects.

This method, inspired by recurrent units such as LSTM and GRU (Gated Recurrent Unit), employs gating mechanisms to control the flow of information, and it measures the activation function for constructing the output.

The general formulation for the model is as shown in Equation 2.10 and in Figure 2.9a. W_i and Y_i are learnable parameters, x_i is the feature vector modality i, σ is the sigmoid activation function and $[.,.]$ stands for concatenation.

$$\begin{aligned} h_i &= \tanh\left(W_i \cdot x_i\right) \\ z_i &= \sigma\left(Y_i \cdot [x_1, \ldots, x_k]\right) \\ h &= \sum_{i=1}^{k} z_i * h_i \end{aligned} \tag{2.10}$$

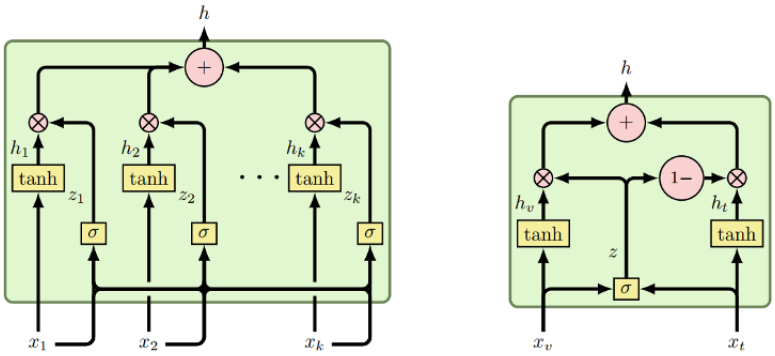


Figure 2.9: (a) Gated Multimodal Unit (GMU) for more than two modalities. (b) Simplification for bimodal approach (Arevalo et al., 2020).

In the bimodal approach, the formulation is shown in Equation 2.11, where W_1, W_2 and W_z are learnable parameters, x_1 and x_2 are modality feature vectors, σ is the sigmoid activation function and $[.,.]$ stands for concatenation.

$$\begin{aligned} h_1 = \tanh\left(W_1 \cdot x_1\right), \qquad h_2 = \tanh\left(W_2 \cdot x_2\right) \\ z = \sigma\left(W_z \cdot [x_1, x_2]\right) \\ h = z * h_1 + (1 - z) * h_2 \end{aligned} \tag{2.11}$$

2.3.2 Self-Attention in Multimodal Context

Multimodal fusion is a key aspect of these transformers, referring to the methods employed to combine information from different modalities. Several fusion techniques have been developed to facilitate effective integration of modalities, each with its unique approach, including (1) early summation, (2) early concatenation, (3) hierarchical attention (Xu et al., 2023). Thus, we will review these main modelling practices of transformers.

Early Summation

Early summation (Gavrilyuk et al., 2020; Xu and Zhu, 2021) takes the information from different modalities and combine them by sum at each token position, illustrated in Figure 2.10a, often before feeding the data into the neural network model. One of its main advantages is that id does not increase computational complexity, however, it assumes that all modalities contribute equally to the task, and it might not capture complex relationships between modalities as effectively as more sophisticated fusion methods.

Early concatenation

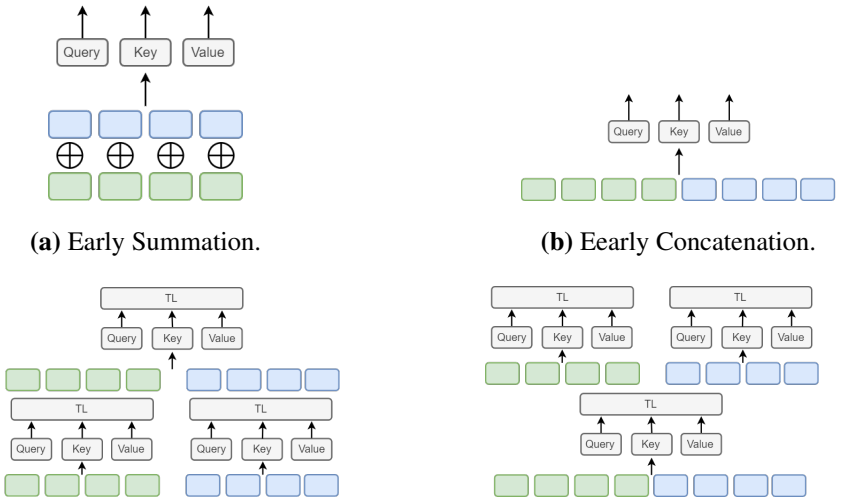
Concatenation (Guo et al., 2020; Sun et al., 2019; Shi et al., 2021; Zheng et al., 2021), creates a single unified representation that encompasses information from all modalities, as seen in Figure 2.10b. The combined representation is then fed into subsequent layers, allowing the model to learn and capture the interdependencies between the different modalities. During the training process, the model refines the weights assigned to each modality, adapting to the task at hand. Early concatenation is also termed "all-attention" or "Co-Transformer" (Zhan et al., 2021; Xu et al., 2023)

Hierarchical Attention

In this type of fusion there are two principal ways to apply hierarchical attention, *multi-stream to one-stream* and *one-stream to multi-stream*.

Multi-Stream to One-Stream. A normal way to use hierarchical attention is that all modalities inputs are encoded by an independent layer and their outputs are concatenated and fused by another transformer, such as Li et al. (2021) did, depicted in Figure 2.10c.

One-Stream to Multi-Stream. Unlike the Multi to One-Stream, this practice first applies concatenation, as shown in Figure 2.10d, the output is encoded by a single-stream transformer, followed by two separated streams maintaining the uni-modal representations with the advantage of taking global information. InterBERT (Lin et al., 2020) is one example of this attention.



(a) Early Summation.

(b) Eearly Concatenation.

(c) Hierarchical Attention

(d) Hierarchical Attention

Figure 2.10: Transformer-based cross-modal interactions: (a) Early Summation, (b) Early Concatenation, (c) Hierarchical Attention (multi-stream to one-stream), (d) Hierarchical Attention (one-stream to multi-stream). "Q": Query embedding; "K": Key embedding; "V": Value embedding. "TL": Transformer Layer.

2.4 Discussion

Multimodal transformers offer a promising approach to questionable content classification due to their ability to integrate and analyze multiple data types simultaneously. However, it is crucial to address and mitigate shortcomings related to computational overhead, modality imbalance, inherent biases, lack of interpretability, and scalability challenges. As research and technology advance, it is expected

that these challenges will be addressed, enabling more effective and responsible use of multimodal transformers in questionable content classification.

The above-mentioned self-attention variants for multimodal interactions are generic in modality and can be applied in flexible strategies for tasks of different levels of granularity. In particular, these interactions can be flexibly combined and nested. For example, multiple cross-flow attention streams are used in a hierarchical approach (from one-stream to multi-stream) as compared to a decoupled two-stream model. Furthermore, these variants can be extended to include more than three modalities (≥ 3). An example is TriBERT (Rahman et al., 2021), which implements a trimodal cross-attention (co-attention) for vision, pose, and audio, where the Query embedding is combined with the Key and Value embeddings of the other modalities. This type of cross-attention through concatenation is applied to three modalities (speech, video, and audio) by Tsai et al. (2019).

CHAPTER 3

RELATED WORK

This chapter focuses on the topic of questionable content detecion and explores various related works. Section 3.1 discusses some datasets that are collected for different tasks but focusing in multimodality, while Section 3.2 delves into the attempts at solving the problem using different Multimodal Attention Based Models (MABMs). Section 3.3 highlights the techniques that use Multimodal Transformers to better merge different modalities.

3.1 Multimodal Datasets for Video Analysis

Multimodal datasets represent a rich and diverse form of data that incorporates multiple modalities, such as text, images, audio, and video. These datasets are designed to capture the complexity and richness of real-world information, enabling a more comprehensive understanding of a given task or problem. By combining various modalities it is possible to leverage the strengths of each data type, leading to more robust models. The integration of multimodal information is particularly beneficial in fields like computer vision, speech recognition, and human-computer interaction, where a broader range of sensory inputs enhances the overall performance and applicability of machine learning algorithms.

At the inception of multimodal datasets, some were collected for different tasks, for instance, Castro et al. (2019) proposed MUStARD, which stands as a multimodal video corpus tailored for automated sarcasm discovery research. Comprising audio-visual utterances, MUStARD is uniquely annotated with sarcasm labels. Another is CMU-MOSI, collected by Zadeh et al. (2016), this dataset contains 2,199 opinion video clips, meticulously annotated with labels for subjectivity, sentiment intensity, per-frame and per-opinion visual features, and per-millisecond annotated audio features. CMU-MOSEI (Bagher Zadeh et al., 2018) encompass online videos for sentiment analysis and the identification of nine discrete emotions (angry, excited, fear, sad, surprised, frustrated, happy, disappointed, and neutral). Moreover, UR-FUNNY dataset (Hasan et al., 2019) was proposed and it is a multimodal dataset of humor detection in human speech involving the effective use of words (text), accompanying gestures (visual), and prosodic cues (acoustic). Arevalo et al. (2020) introduced a multimodal dataset for genre prediction on movies by collecting genre, poster, and plot information for each movie. Kinetics-400 (Kay et al., 2017) is a multimodal dataset that contains video clips covering a diverse range of 400 human action classes, with at least 400 video clips for each action.

3.1.1 Datasets for Questionable Content and Video Analysis

Multimodal datasets are essential for enhancing questionable content detection by providing comprehensive contextual information. By combining different modalities, models can discern subtleties and improve differentiation between different content types. Some datasets have been collected for this task, for example, XD-Violence (Wu et al., 2020) is a large-scale dataset that provides simultaneous visual and audio data capturing instances of violent events. It encompasses a total of 4757 videos, equivalent to 217 hours of content, covering six distinct types of violent events. Another instance is the UCF-Crime dataset proposed by Sultani et al.

(2018), it consists of long untrimmed surveillance videos which cover 13 real world anomalies, including Abuse, Arrest, Arson, Assault, Road Accident, Burglary, Explosion, Fighting, Robbery, Shooting, Stealing, Shoplifting, and Vandalism. These anomalies are selected because they have a significant impact on public safety. Concerned about the detection of fights in surveillance footage, Degardin and Proença (2020) mined 1,000 videos (collected from Youtube and LiveLeak): 784 containing normal events, and the remaining 216 containing some fight segment. This dataset, named UBI-Fights, was manually annotated at the frame-level. In the field of online content Baharlouei and Solorio (2024) introduced the Comic Mischief dataset. Such dataset contains 1-minute clips obtained from YouTube videos that were crawled, segmented, and manually labeled. By curating a diverse range of videos that encompass these distinct forms of comedic expression, the comic mischief dataset provides a valuable resource for studying and analyzing the multifaceted nature of humor in online content. Table 3.1 summarizes each dataset in the modalities they use and the number of instances, also the type of task.

Table 3.1: Multimodal datasets proposed in multimodal representation learning. The input modalities are ℓ: language, v: video, a: audio, i: image and o: optical flow.

Dataset	Modalities	Num. of Samples	Prediction Task
MUStArd	$\{\ell, v, a\}$	690	sarcasm
UBI-Fights	$\{v, a\}$	1000	violence
CMU-MOSI	$\{\ell, v, a\}$	2,199	sentiment
Comic Mischief	$\{\ell, v, a\}$	3,604	humor
XD-Violence	$\{v, a\}$	4,757	violence
UR-FUNNY	$\{\ell, v, a\}$	16,514	humor
CMU-MOSEI	$\{\ell, v, a\}$	22,777	sentiment, emotions
MM-IMDB	$\{\ell, i\}$	25,959	movie genre
Kinetics-400	$\{v, a, o\}$	306,245	sentiment, emotions

We focus specifically on Comic mischief detection, which is a significant challenge in the field of artificial intelligence and multimedia media processing for several reasons. Unlike datasets that explore humor tasks, this specific one encompasses two main tasks: a binary task and a multi-label classification task, in which four distinct categories of humor are identified: sarcasm, gory humor, slapstick humor, and mature humor. In addition, three input modalities are used: audio, text, and video, which adds an additional layer of complexity and richness to the task. This is why we focus primarily on the detection of this task and also on this dataset.

3.2 Questionable Content Detection

The detection of questionable content is vital to create a safe online environment, like YouTube in which more than 500 hours of video are uploaded per minute. Several studies have already addressed this issue, for instance, Shafaei et al. (2021) propose a scheme based on multimodal deep learning that addresses the problem of classifying questionable content in movie trailers utilizing LSTMs and contextual attention. A GMU was used to combine multimodal information from video clips. Also Rodríguez Bribiesca et al. (2021) use MABMs to learn information from different modalities for movie genre classification task, using GMU for final fusion. Similarly, Pang et al. (2021) focuses on violence detection in videos using visual and audio information. They use Multimodal Attention Based Models (MABMs) with standard cross-attention in pairs. Liu et al. (2023) propose a MABM for real-time anomaly detection in videos implementing a two-stage process. Only the video modality is considered in this work. Wei et al. (2022) proposes a MABM based on label refinement and multimodal fusion for violence detection in videos. Audio and video modalities were considered with a pairwise cross-attention mechanism. Xiao et al. (2023) focus on violence detection in videos, they employ a MABM to fuse RGB, optical flow and auditory features. Rendón-Segador et al. (2023) use a MABM

to detect violent activity using model based on vision transformer and neural structured learning. Yang et al. (2022) propose multimodal action recognition method using visual and audio information. Recently Baharlouei and Solorio (2024) describe a MABM implementing a three-modal hierarchical variant of cross-attention (HCA) for comic mischief detection. Authors show that multimodal information combined with HCA resulted in better performance than baselines, and state-of-the-art models including recent MABMS/Transformers.

Discussion: These works have highlighted the importance of detecting different types of online content, especially videos, using multimodal approaches. Among the advantages of these approaches is the ability to fuse diverse sources of information (such as visual, auditory, textual, optical flow among others) using multimodal attention models (MABMs), which has been shown to significantly improve detection performance compared to other methods. However, these models present considerable challenges, such as computational complexity, for example, Rodríguez Bribiesca et al. (2021) take into account two transformers per modality applying a cross-attention module, increasing the computational cost. In addition, some of the work has focused on the use of bimodal models, which may limit the use of the available information. Therefore, there is still work to be done in broadening the spectrum of questionable content types that these models can detect, as well as in optimizing the models for use in real time and in scenarios with limited computational resources.

3.3 Multimodal Transformers

This section aims to explore different techniques for fusing multimodal information, as well as those models that focus on the analysis of attention head weighting.

3.3.1 Multimodal cross-attention

Multimodal representation learning and fusion aim to generate a unified representation of multiple modalities that facilitates automatic analysis tasks by constructing classifiers or other models. In the context of attention-based models, multimodal attention mechanisms associated to different modalities are combined, expecting that the fusion captures information about the interaction of modalities. A basic approach is to concatenate individual representations features to obtain a final representation (Anwar et al., 2022; Kiela and Bottou, 2014; Pei et al., 2013; Suk and Shen, 2013). Although this is a straightforward strategy, given that the nature of the data for each modality is different, their statistical properties are generally not shared across modalities (Srivastava and Salakhutdinov, 2012), requiring the predictor to model complex interactions between them.

Instead, other leverage on cross-attention to have a contextualized representation of each modality given the information of the others. Zaidi et al. (2023) proposed a multimodal transformer with dual attention, where they used co-attention to capture complex dependencies across different modalities. In the same way, Yoon et al. (2022) use a co-attention module to capture the relationship across modalities, and generate a more comprehensive representation. Moreover, the hierarchical cross-attention has been explored (Chen et al., 2022, 2021; Dutta and Ganapathy, 2023; Zhang et al., 2022) in order to capture hierarchical intra- and inter-modal correlation. Despite the effectiveness of these methods, they consider only two modalities, which are not appropriate tasks involving more modalities.

To the best of our knowledge, the only work considering cross-attention of more than two modalities is that of Baharlouei and Solorio (2024). There, authors apply three times HCA that is later combined via concatenation. While effective, it is not clear if HCA is the best way to combine multimodal information, as the modalities are processed sequentially, meaning that information from one modality

is incorporated before the next modality is processed. This can result in significant information loss. In addition, since each cross-attention layer relies on the outputs of previous layers, any errors or noise in one modality can propagate and amplify through subsequent layers. Likewise, it remains unexplored the use of alternative ways to combine the outputs of these attention mechanisms.

3.3.2 Multimodal Attention-Head Fusion

Multi-head attention was shown to make more efficient use of the Transformer capacity. While the model has gained widespread acceptance and recent attempts to investigate the kinds of information learned by attention heads (Raganato and Tiedemann, 2018), the analysis of multi-head attention is challenging. Previous studies examining the formulation of representations by multi-head attention mechanism concentrated on either the mean or the peak attention weights across all heads (Voita et al., 2018; Tang et al., 2018).

In this way, some works have studied gated attention which allows the model to selectively emphasize or de-emphasize different parts of the input sequence when computing the self-attention scores. Huang et al. (2019) propose attention on attention model, which extends the conventional attention mechanisms to determine the relevance between attention results and queries. Ahmed et al. (2017) propose weighted transformer, which replace the multi-head attention by multiple self-attention branches that the model learns to combine during the training process. Transformer with weighted forced attention is proposed by Okamoto et al. (2020), where each modality is weighted during training. Kim et al. (2020) introduce T-SGA, whose attention weights are attenuated according to the distance between target and context symbols.

Recently, it has been shown that those attention matrices lie on a low-dimensional manifold and, thus, are redundant (Bhojanapalli et al., 2021). Thus, Nguyen et al.

(2022) introduce FishFormer, a class of efficient and flexible transformers that allow the sharing of attention matrices between attention heads. Voita et al. (2019) focus their studies on the importance of each attention head by applying a soft pruning. Instead, Michel et al. (2019) follow the same principle, but applying hard pruning, Table 3.2 summarizes the previous works.

Table 3.2: Self-attention variants for multimodal interaction/fusion. α, β, w and g denote weightings. C: Concatenation. L: Linear transform. G: Gaussian matrix. S: Score matrix. Q: Query matrix. K: Key matrix. V: Value matrix. σ: Sigmoid activation function.

Method	Type of attention	Level	Formulations
AoANet (Huang et al., 2019)	Gated	SA	$\mathbf{Z} \leftarrow \sigma\left(L_1\left(\mathbf{Z}_{(\mathbf{A})}\right)\right) \odot \left(L_2\left(\mathbf{Z}_{(\mathbf{A})}\right)\right)$
T-SGA (Kim et al., 2020)	Gated/Kernel	SA	$\mathbf{Z} \leftarrow \text{SoftMax}\left(G \odot S\right) V$
FishFormer(Nguyen et al., 2022)	Regression	SA	
MBA (Ahmed et al., 2017)	Weighted	MHSA	$\mathbf{Z} \leftarrow \sum_{i=1}^{n}\left(\alpha_i \cdot \mathbf{Z}_{(i)} \cdot \beta_i\right)$
Forced Attention (Okamoto et al., 2020)	Weighted	MHSA	$\begin{cases} \mathbf{Z}_{(\mathbf{A})} \leftarrow (1-w)\, MHSA\left(\mathbf{Q_B}, \mathbf{K_A}, \mathbf{V_A}\right) \\ \mathbf{Z}_{(\mathbf{B})} \leftarrow (1-w)\, MHSA\left(\mathbf{Q_A}, \mathbf{K_B}, \mathbf{V_B}\right) \end{cases}$
Soft Pruning (Voita et al., 2019)	Weighted	MHSA	$\mathbf{Z} \leftarrow C\left(g \cdot \mathbf{Z}_{(i=1,\ldots,n)}\right)$
Hard Pruning (Michel et al., 2019)	Weighted	MHSA	$\mathbf{Z} \leftarrow C\left(\{0,1\} \cdot \mathbf{Z}_{(i=1,\ldots,n)}\right)$

Discussion: The aforementioned works propose different approaches to improve attention mechanisms. Methods such as gated attention and weighted transformers seek to improve accuracy and efficiency by emphasizing relevant parts of the input sequence or by combining self-attention branches. However, some of these appear to be equivalent, as is the case with MBA and pruning methods.

In addition, as Voita et al. (2019) demonstrated, some attention matrices learn redundant information, which has led to the development of pruning techniques that remove less relevant heads, and models such as FishFormer, which optimize efficiency by sharing attention matrices among heads. Although these methods achieve greater efficiency, they may oversimplify the model in some contexts. In this sense, the proposal of a new GMU-based method may be justified, as it offers a more effective combination of multimodal information.

3.4 Discussion

In this chapter, we review various approaches to deal with classification tasks using multimodal information. So far no other model has tested with only two modalities in the comic mischief dataset, however, Baharlouei and Solorio (2024) performed bimodal experiments, showing that the use of three modalities leads to better results. Although some of previous methodologies have proven to be effective, their limitation to the use of only two modalities may not be sufficient in certain cases. For example, the T-SGA model has shown effectiveness and its concept can be applied to several tasks, but it is restricted to a single modality. Or some other methods that use a single weighing for the different representations at both the multi-head and self-attention level.

We address these problems by using more than two modalities for classification problems, and exploit the correlation between them by means of cross-attention modules. In addition, we propose a weighting method for each modality to obtain a better representation, focusing on the most relevant aspects of each modality for the classification task.

As for the datasets, although some offer three modalities, which is ideal for our proposal, they tend to focus on tasks such as sentiment analysis or emotion classification. Therefore, we decided to focus on the comic mischief dataset, which, as mentioned in Section 3.1.1, provides three modalities and focuses on detecting questionable content, specifically detecting comic mischief in videos. This dataset presents a classification challenge due to its multi-label nature and the inherent subjectivity in its classification, leading to further challenges, such as the fact that a modality may better classify a category, giving it greater weight or relevance.

CHAPTER 4

PROPOSED METHOD

This chapter introduces a transformer-based model that utilizes a novel approach for combining different modalities, both at the self-attention level and at the multi-head attention level. It is organized into three sections, each concentrating on a particular element. The first section outlines the proposed method. The second section explains the feature extraction process for each of the three modalities involved in the task. Finally, the third section presents our strategy for detecting questionable content, detailing its general description and implementation specifics.

4.1 Overview

The widespread of multimodal information throughout diverse platforms and apps has boosted the amounts of multimodal data being generated. This represents a potential risk to users which can be exposed to inappropriate or harmful content. Therefore, methods for the identification of this type of content are highly needed. The proposed method addresses the challenge of detecting comic mischief using a transformer-based model. It leverages the sequential nature of inputs, focusing on three modalities: text, audio, and video. Pretrained models are also used to extract relevant features from the data.

This study introduces a novel approach to integrate the three modalities to achieve a contextualized representation. The comic mischief detection task comprehends both binary and multi-task aspects, with the latter involving five distinct target values, each representing a different class of comic mischief. This approach has several benefits, such as better generalization to different types of data and better adaptation to content variability. However, it also faces limitations, such as additional computational complexity and the need for large amounts of labeled data to efficiently train the models.

4.2 Feature Extraction

Pre-trained models have become a popular tool for feature extraction in machine learning and data analysis tasks. There are several reasons why these models are preferred for this task. First, pre-trained models offer low computational cost. Training a model from scratch can require a significant amount of computational resources and time, especially when dealing with deep neural networks. By using a pre-trained model, already optimized models are leveraged, thus reducing the computational cost associated with feature extraction.

Another reason is the size of the data sets, pre-trained models have often been trained on large datasets, which allows them to learn robust and general representations of the data. These models can capture essential features that are useful in a variety of tasks, even when the input data are limited or come from different domains than the original dataset. In addition, pre-trained models are available in many machine learning libraries and platforms, making them easy to implement and use.

As for the ways to do feature extraction, these can vary according to the needs and available data. A common way is the direct use of a pre-trained model. In this

approach, features are extracted directly from the intermediate layers of the model, allowing to capture information from different levels of abstraction in the data. Another option is fine-tuning, where the pre-trained model is tuned on the specific new dataset, allowing the model to learn features more relevant to the particular task and thus improving performance.

The following are the pre-trained models used in this work for feature coding.

4.2.1 Textual Feature Extraction

Within the domain of textual data, we used BERT (Kenton and Toutanova, 2019) model, Figure 4.1, to extract textual embeddings. This model is a natural language processing architecture that uses a bidirectional approach to understand the context of words in a sentence, enabling better understanding of meaning compared to unidirectional models. It leverages deep learning techniques and transformers to pre-train on large amounts of text and then adjust to specific tasks such as classification, translation and question answering, achieving outstanding results in a variety of natural language processing applications.

The use of BERT for textual feature extraction is very beneficial. As mentioned, BERT is bidirectional, which means that it can analyze the full context of a word in a sentence, thus capturing the contextualized meaning of each word and providing richer and more accurate features.

In addition, BERT has pre-training on large text corpora, such as Wikipedia and books. This allows it to learn complex language patterns and high-level semantic representations, making it capable of capturing features from a wide variety of texts. In addition, BERT is versatile and can be adapted to different NLP tasks, such as text classification, named entity recognition and sentiment analysis, making it a popular choice for many natural language processing projects.

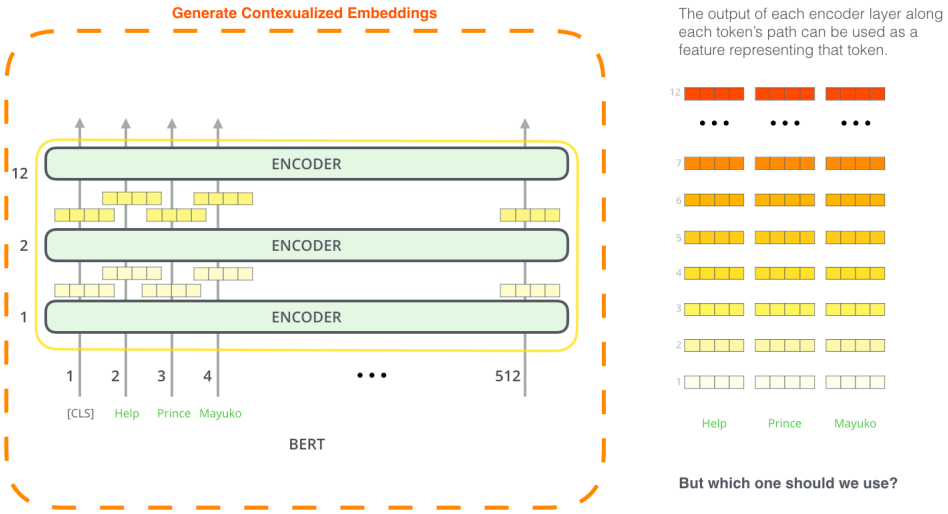


Figure 4.1: Feature extraction from BERT. The selection of which one we should use depends on the task. (Alammar, 2018a).

The feature extraction process starts with the preparation of the input data. Texts are tokenized using BERT's specific tokenizer and special tokens are added, such as '[CLS]' at the beginning and '[SEP]' at the end of each input. Then, a pre-trained version of BERT is loaded, which can be the base model or a specific variant depending on the needs of the task. The tokenized texts are passed through the BERT model, which processes the text sequences and produces a feature output for each token.

Another advantage is the use of the WordPiece algorithm, which splits words into sub-words for an even more detailed representation. This is useful for capturing words that are not in the vocabulary.

4.2.2 Audio Feature Extraction

Audio feature extraction consists of converting an audio signal into a more manageable representation containing information relevant to the desired purpose. Common techniques for extracting audio features include converting the signal into a spec-

trogram, which shows the amplitude of frequencies over time, and calculating mel frequency cepstral coefficients (MFCCs), which capture information about the timbre of the sound based on the mel scale, a perceptual scale of tones.

Other important features include RMS (Root Mean Square) energy and Zero Crossing Ratio (ZCR), which provide information about the loudness and frequency of the audio, respectively. Harmonic features such as chromatic features, which represent how musical notes are present in an audio file, can also be extracted. In addition, temporal features such as signal energy over time are considered.

In our particular case, as we want to process sound effects, ambient sounds (e.g. explosions) and dialogues (speech) we use the pretrained VGGish network (Hershey et al., 2017), because it is useful for extracting generalized audio features, such as music, sound effects, and ambient noise, etc. as opposed to other models that focus on specific tasks such as speech recognition. Morover, as the video instances have a long duration, they were divided into samples of 60 seconds to facilitate the work. VGGish network is one of the most widely used for extracting audio features because it combines the robustness of convolution, pooling, and fully connected layers to capture high and low (MFCCs) frequency patterns in audio data. Thanks to its design, VGGish can extract audio features from different levels of abstraction, making it versatile for different audio processing tasks.

Furtheremore, VGGish has a pre-trained model on large audio datasets, giving a solid foundation for feature recognition on different types of audio and saving time and resources on training models from scratch.

4.2.3 Video Feature Extraction

Video feature extraction is an essential process for visual content analysis, as it allows to identify and understand various aspects of videos such as objects, movements,

actions and contexts. There are several ways to extract video features, among them:

Frame-based extraction processes video frames individually, similar to how still images are processed. Computer vision techniques such as object detection, face recognition and image classification are used to extract relevant features from each frame. However, this method may not capture the continuity and temporality of the video.

Another technique is optical flow-based extraction, which analyzes motion between consecutive frames of a video. This analysis allows the extraction of motion features in the video, which is useful for tasks such as action detection and activity recognition.

Recurrent neural networks (RNNs) are capable of processing temporal sequences of data, such as video, by maintaining an internal state that captures temporal dependencies. However, RNNs can have efficiency problems and limited capture of visual information.

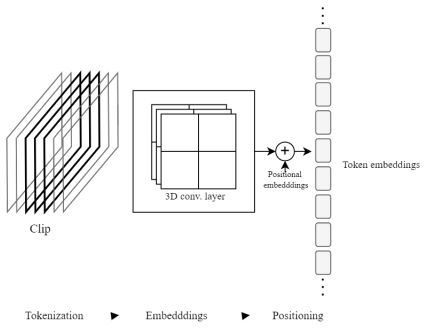


Figure 4.2: Overview of the input pre-processing step, showing tokenization and embedding strategy.

In our case, as mentioned previously, we divided video segments into intervals of 60 seconds, so each interval can be processed. In this way 3D convolutional models apply 3D convolutional neural networks (CNNs) to data cubes instead of individual images, see Figure 4.2. These models process video as a 3D volume, considering both spatial and temporal dimensions, to extract features more accurately.

The i3D (Inflated 3D Convolutional Network) (Carreira and Zisserman, 2017) architecture is one of the most widely used for video feature extraction due to its efficient and effective design. i3D converts 2D operations, such as convolutions, into 3D to process the video in its entirety (both spatially and temporally). In addition, it can pre-train on large 2D image datasets, such as ImageNet, and then adapt the model to 3D for videos, which has been shown to improve performance on video tasks.

i3D combines the ability to capture complex visual features of 2D CNN models with the temporal analysis of 3D models. This architecture has proven useful in a variety of applications such as action recognition, event detection and scene understanding in videos, and is a popular choice for video feature extraction.

4.3 Transformer for Questionable Content Detection

This section describes the proposed cross-attention mechanisms and the usage of a GMU for combining them. Before that, we describe the base multimodal attention-based model (MABM) for comic mischief detection that we consider.

4.3.1 Reference model

As a base model, we consider a simplified version of the model proposed by Baharlouei and Solorio (2024), a generic diagram is shown in Figure 4.3. We describe this simplified model with both approaches, binary classification task, and multi-task model.

The so-called, HIerarchical Cross-attention model with CAPtions (HICCAP) implements a hierarchical cross-attention (HCA) to combine embeddings of multiple modalities. It is divided in several stages that are described next.

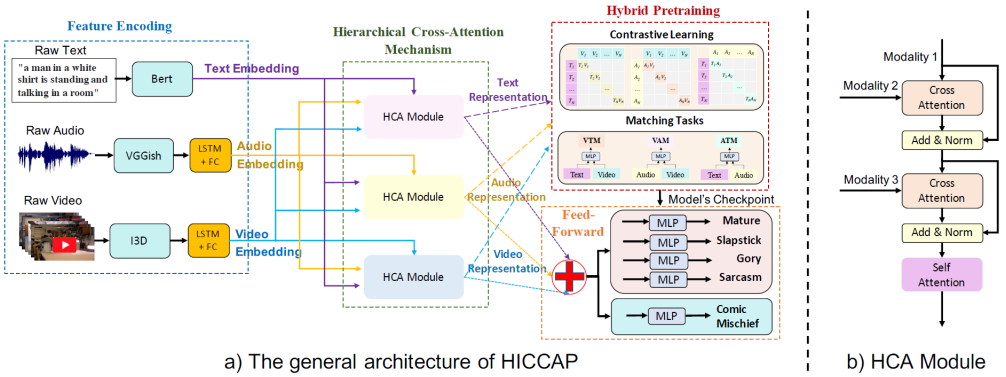


Figure 4.3: (a) HICCAP general architecture. (b) The hierarchical attention model implemented by Baharlouei and Solorio (2024)

Hierachical Cross-Attention (HCA) mechanism

Once the involved modalities have been encoded with descriptors, a hierarchical cross-attention (HCA) mechanism is adopted (Figure 4.3b). Three HCA modules are incorporated into HICCAP, each performing cross-attention at multiple levels to harness the attention across all three modalities, rather than solely focusing on pairwise attention. HCA facilitates the exploration of complex relationships and dependencies within the multimodal data, ultimately enhancing the overall fusion and understanding of the combined modalities. The model concatenates the contextualized outputs of the three HCA mechanisms before classification.

Pretraining and classification

For the classification stage, following the original method, two tasks are considered: binary and multi-label classification. In the binary task, the objective is to determine whether a video clip contains comic mischief or not. To accomplish this, a multilayer perceptron (MLP) model is adopted. On the other hand, the multi-label classification aims to classify clips into four distinct categories of comic mischief. To

tackle this task, a separate MLP is employed for each class, implementing a multi-task learning approach (Crawshaw, 2020). This allows the model to simultaneously learn and classify the different categories of comic mischief, leveraging the shared information across tasks to enhance the overall performance.

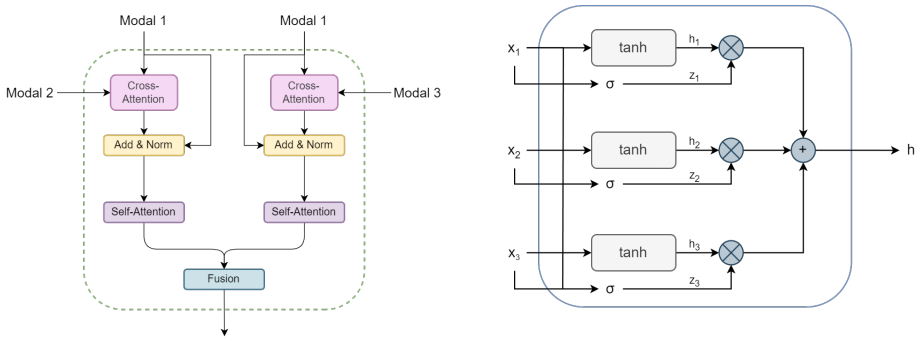
The HICCAP was pretrained using contrastive learning and multimodal matching tasks. For this work we decided to evaluate the performance of the model when trained from scratch, this is to reduce the number of factors that may have an impact on the modeling process.

4.3.2 Parallel cross-attention

The proposed ParCA mechanism, depicted in Figure 4.4a, replaces the HCA module and aims to enhance the representation of each modality concerning the two other modalities. This enhancement is performed with two multimodal cross-attention mechanisms that are then fused, see Figure 4.4a. ParCA comprises two sub-blocks: cross-attention and self-attention. Cross-attention calculates the attention in parallel for modality $\mathbf{m_1}$ taking modalities $\mathbf{m_2}$ and $\mathbf{m_3}$ (Equation 4.1).

$$\mathbf{x}^{\mathbf{2}}_{\mathbf{m_1}} = \text{softmax}\left(\frac{Q_{\mathbf{m_1}} K^{\top}_{\mathbf{m_2}}}{\sqrt{d_k}}\right) V_{\mathbf{m_2}}, \;\; \mathbf{x}^{\mathbf{3}}_{\mathbf{m_1}} = \text{softmax}\left(\frac{Q_{\mathbf{m_1}} K^{\top}_{\mathbf{m_3}}}{\sqrt{d_k}}\right) V_{\mathbf{m_3}} \tag{4.1}$$

In Equation 4.1, $\mathbf{x}^{\mathbf{2}}_{\mathbf{m_1}}$ is the representation of modality $\mathbf{m_1}$ based on $\mathbf{m_2}$, and $\mathbf{x}^{\mathbf{3}}_{\mathbf{m_1}}$ the representation based on $\mathbf{m_3}$. Then we employ a residual connection around the first sub-block, followed by layer normalization for both outputs. The last sub-block takes the outputs and passes them through standard self-attention to further enhance the representation of the modality and make it more suitable for classification. The outputs of these enhanced attention mechanisms are combined with a fusion technique.



(a) Parallel Cross-Attention **(b)** Gated Multimodal Unit (GMU)

Figure 4.4: a) The proposed ParCA mechanism, consists of two sub-blocks: cross-attention and self-attention. b) The model of GMU for more than two modalities.

For the fusion we adopted the classical concatenation and sum techniques. Additionally, we propose the use of a GMU (Arevalo et al., 2020) for learning the relevance from each of the paths of cross-attention (mote that at this stage, only $\mathbf{x}^{\mathbf{2}}_{\mathbf{m_1}}$ and $\mathbf{x}^{\mathbf{3}}_{\mathbf{m_1}}$ are merged to obtain a final representation of modality $\mathbf{m_1}$, since ParCA replaces HCA module). A GMU learns the importance of the information from each modality and aims to fuse only the most relevant aspects. Using a GMU involves the derivation of an intermediate representation by amalgamating data from diverse modalities. Figure 4.4b illustrates the architecture of a GMU, where each x_i denotes a feature vector linked to modality i. Each feature vector is input to a neuron with a tanh activation function, aiming to encode an internal representation feature specific to the modality. For every input modality, x_i, there exists a gate neuron (σ), responsible for regulating the impact of the feature computed from x_i (represented as z) on the overall output of the unit. Upon receiving a new sample, the gate neuron associated with modality i processes input feature vectors from all modalities to determine the contribution of the modality i to the internal encoding of the given input sample.

This weighting mechanism acknowledges that not all modalities contribute equally, as there may be instances where the audio does not correspond to the visual

elements or the dialogue in a video. Such discrepancies between modalities can lead to conflicts or inconsistencies in the information they provide. We repeat this process to obtain the representations for modalities $\mathbf{m_2}$ and $\mathbf{m_3}$, respectively, and concatenate them at the end, as in the reference model. ParCA is finally incorporated into the reference model replacing the HCA mechanism.

4.3.3 Transformer model

Different from the HICCAP base model, which contains only one depth level per modality, as part of our proposed method, we extend the model to a transformer-based one, increasing the number of both HCA and ParCA modules and structuring it around encoders, this can offer several significant advantages that improve its ability to handle and understand multimodal content.

First, transformers and their multiple attention heads are ideal for capturing complex multimodal dependencies. Using multiple attention heads allows the model to pay attention to different aspects of the inputs simultaneously, improving the understanding of interactions between text, audio and video. By increasing the number of modules, as seen in Figure4.5, this functionality can be emulated, allowing the model to process information in a more robust and diversified way.

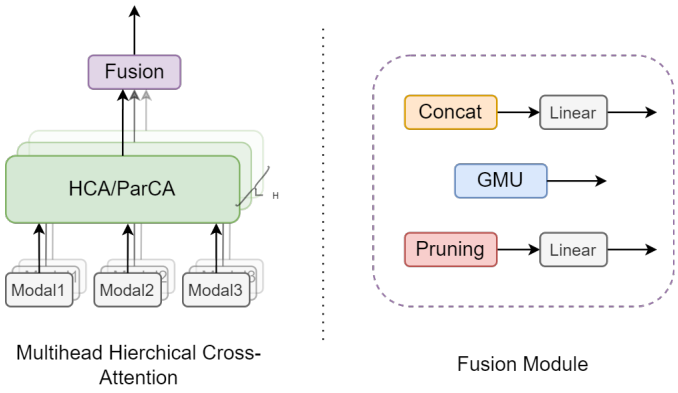


Figure 4.5: Multihead modules for three different modalities.

In our proposal, we address three different approaches for merging the output information from each head into a multi-head model. The first approach is concatenation, which is the classical and widely used method for combining the output information from each head.

The second approach is the use of Gated Multimodal Unit (GMU). Inspired by the ParCA module, we propose the implementation of GMU for merging the outputs of the heads. The GMU allows learning the contribution of each head adaptively, differentiating between those that contribute more significantly to the task at hand.

The third approach is head pruning. Following the ideas presented by Voita et al. (2019), we implemented a head pruning method to weight each head differently, where each head has a different weight along the different encoders, and each weight is learned during training. This approach allows discerning the relevance of each head through a selection process, in which less important heads are eliminated.

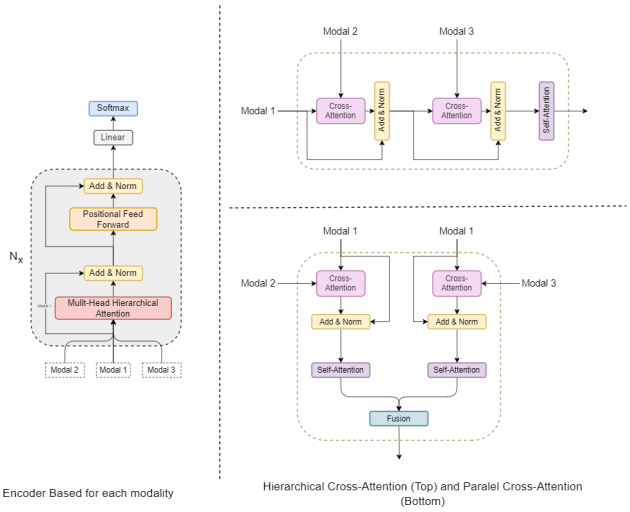


Figure 4.6: Transformer-based model for multimodality

Moreover, we increase the structure to one based on encoders, as in Figure4.6. This structure provides scalability and flexibility to the model. In this way, input sequences can be processed in parallel. In addition, the modularity of the encoders makes it easy to extend the model by adding more layers as needed to capture deeper

and more complex features.

CHAPTER 5

EXPERIMENTS

This section offers an intricate breakdown of the experiments carried out in the study, along with an analysis of the outcomes. It also outlines the setups employed, encompassing a depiction of the dataset and thorough information regarding the architectural elements of the models.

5.1 Comic Mischief Dataset

For our experimental evaluation we relied on a subset, due to privacy policies, of the comic mischief dataset introduced by Baharlouei and Solorio (2024). Such a dataset contains 1-minute clips obtained from YouTube videos that were crawled, segmented, and manually labeled. By curating a diverse range of videos that encompass these distinct forms of comedic expression, the comic mischief dataset provides a valuable resource for studying and analyzing the multifaceted nature of humor in online content. The dataset is labeled according to the following categories:

- **Gory humour:** it is centered around gruesome or macabre elements. It often includes exaggerated violence, blood, or graphic imagery for comedic effect.
- **Slapstick humour:** it is characterized by physical comedy, often involving ex-

(a) Gory humor

(b) Slapstick humor

(c) Mature humor

(d) Sarcasm humor

Figure 5.1: Examples of the considered comic mischief categories in cartoons

aggerated and humorous physical actions, gestures, or mishaps. It relies on visual gags, pratfalls, and absurd or exaggerated physical movements to generate laughter.

- **Mature humour:** it is comedy that contains content or themes intended for mature audiences. It often includes jokes or references that touch upon taboo subjects, such as sexuality, politics, social issues, or dark humor.

- **Sarcasm:** it is a form of humour that involves the use of irony, mocking, or taunting remarks to convey humor or to express a contradictory meaning. It relies on the delivery of statements that are opposite to what is actually meant, often with a dry or sharp tone.

Figure 5.1 shows screenshots from clips associated with the considered humor categories. The dataset is challenging for several reasons, including the multi-faceted nature of comedic expression across categories and the fact that different categories can be expressed/distinguished by different information modalities (e.g., for detecting *Sarcasm* and *Mature*, language, audio information is often more useful than the visual one; while for detecting *Slapstick* humor, visual information tend to be more useful). The working hypothesis of our work and previous approaches is that

by effectively leveraging multimodal information (image, audio, and text) one can develop competitive solutions for this task.

Table 5.1 shows the number of samples available for each of the categories and for different partitions for developing and evaluating our methods and Table 5.2 provides an overview of the class-level statistics for the video segments. Please note that this is a multi-label classification task, that is, each clip may contain humor from more than one category. Also, please note that in previous work the binary classification task of distinguishing a video containing any comic mischief category or not has been studied. Accordingly, in this work, we perform experiments for both classification tasks.

Table 5.1: Samples per partition and per category: Mature Humour (MH), Slapstick Humour (SH), Gory Humour (GH) and Sarcasm (S).

	MH	**SH**	**GH**	**S**	**None**	**All**
Train	222	166	86	374	307	1007
Validation	24	18	6	48	31	113
Test	35	19	11	41	30	113

Table 5.2: Statistics for video segments. C0 and C1 stand for class 0 and class 1, respectively.

	Max		Min		Avg		Med	
	C0	C1	C0	C1	C0	C1	C0	C1
# Words	259	266	0	0	106	118	111	125
V/A Length	64.9	71.9	0.1	9.4	54.7	58.6	60.1	60.5
# Frames	1836	2157	1	108	538	658	460	478

We used a subset of the Comic Mischief dataset. Unlike the original method, we split this subset into three partitions: train (80 %), validation (10 %) and test (10 %), due to the limitation of the data.

5.1.1 Experimental Setup

To ensure fair comparisons, we used the same metrics used in the reference work: F1-measure of the positive (comic mischief) in the binary task and macro F1-measure for the multilabel task.

The training of models involves the utilization of the Adam optimizer with a learning rate set at $2e-5$ and a batch size of 16. In this study, we employed 5 distinct random weight initialization choices for all experiments, and the mean performance based on these random initializations is presented in the results of all subsequent sections. The model was subjected to 25 epochs for binary tasks and 40 epochs for multi-task scenarios, as it was observed that the validation performance reaches saturation within these epoch limits.

5.2 Multihead Attention-based Model

Initially, we varied the number of heads from 2 to 12 for a 1-level modal, a range that is standard in traditional architectures, using the original HCA module. In Figure 5.2 we show the results for the binary task in three different types of combination, including traditional concatenation, pruning (Voita et al., 2019; Michel et al., 2019) and GMU, evidencing how the variation in the number of heads significantly affects the performance of the model. This analysis highlights the importance of optimizing the number of heads to better capture multimodal correlations.

From this figure we can observe that the concatenation method (green) appears to have a consistent and somewhat variable trend compared to the other two methods, as its shaded area is less pronounced. On the other hand, the pruning method (purple) shows greater variability across the different head number configurations, which is reflected in its wider shaded area. The GMU method (yellow) appears to have lower variability, with some fluctuations but not as pronounced as prunning or concatenation.

In terms of performance, at first glance there does not appear to be a drastic difference in the average performance of the three methods, as the lines of all three are relatively close to each other across the graph. However, in the range of 7 to 9

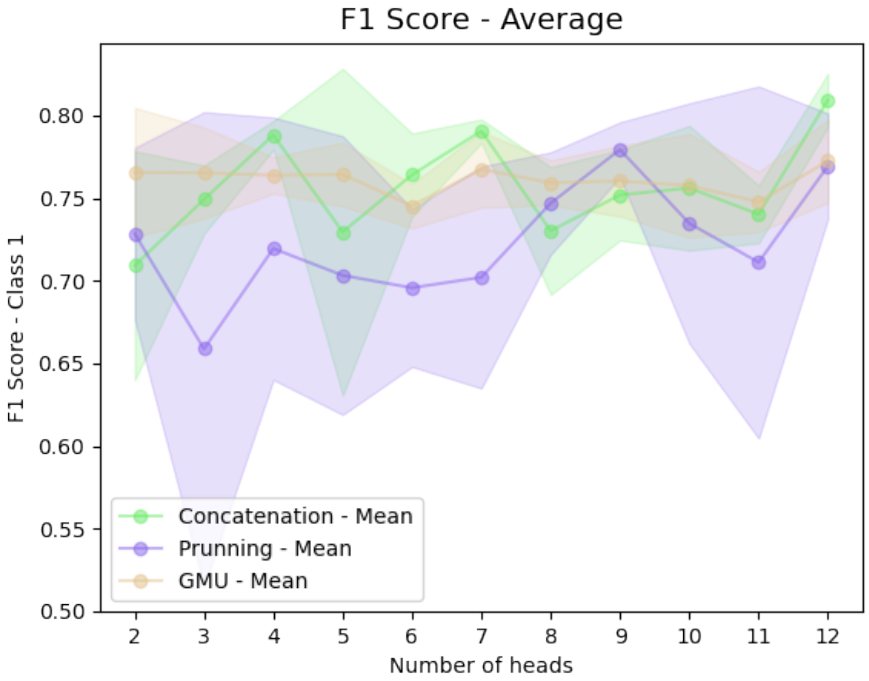


Figure 5.2: Results varying number of heads from 2 to 12 in Binary Task for each HCA module.

there is similar performance among the three and less variability, indicating that the appropriate number of heads may be the midpoint.

A similar analysis was conducted with multilabel task, Figure 5.3, using the same HCA module. Each graphic corresponds to one category in dataset, and the F1 result in that category varying the number of heads in same range. The same three methods of fusion are compared.

From this figure, we can observe that the upper left graph (Mature), concatenation and GMU maintains steady performance with little fluctuation, while pruning shows high variability. In the upper right graph (Gory), concatenation has notable variability with peaks, pruning shows high variability and GMU has significant peaks and valleys. In the lower left graph (Slpastick), concatenation shows stability in the central part, pruning is highly variable and GMU has lower variability than pruning but is not as stable. In the lower right graph (Sarcasm), concatenation is constant

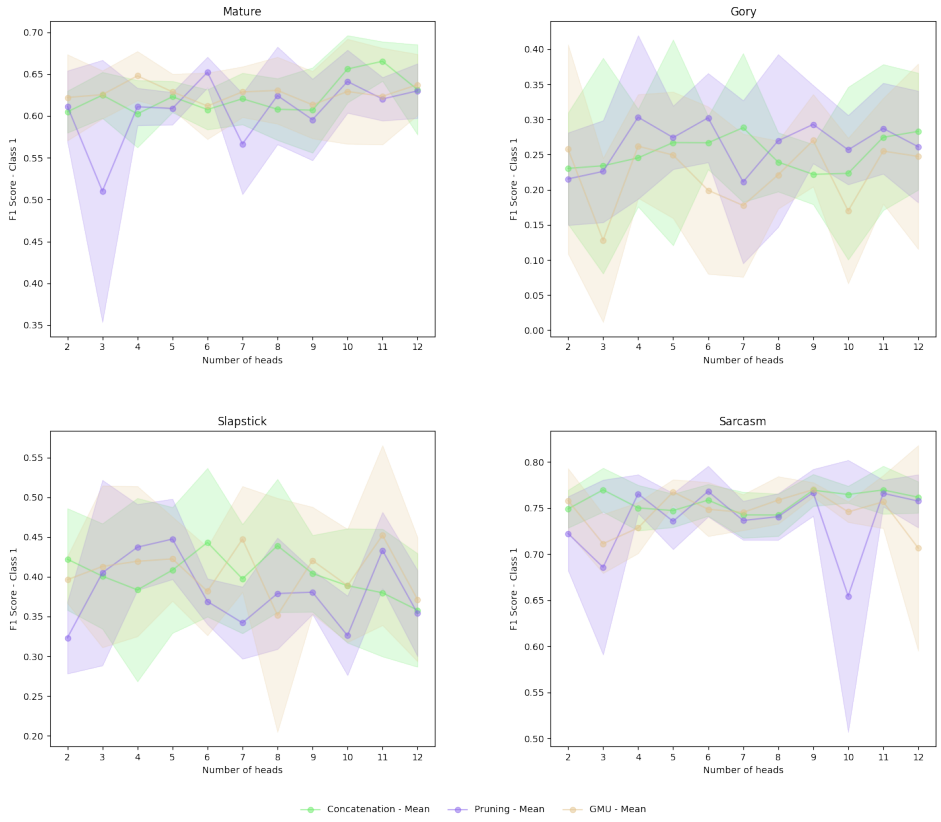


Figure 5.3: Results varying number of heads from 2 to 12 in Multi Task for each HCA module.

with lower variability, pruning shows high variability and GMU is more stable than pruning. Overall, concatenation appears to be the most consistent method across the different classes.

These results suggest that an intermediate number of heads, around 8 to 10, seems to offer a good balance between performance and stability. Again, the mid-point may be the right number of heads.

5.3 Evaluation of Detection Performance

Tables 5.3 and 5.4 show the results obtained by the different variants we tried for the binary and multilabel tasks, respectively. In both tables we compare the performance of the reference model implementing the standard HCA with 1 (first row and baseline) and 8 (second row) heads with alternatives of the HICCAP model implementing our ParCA mechanism (rows 3 and on). The latter variants use different fusion strategies for ParCa, namely: concatenation (ParCACon), sum (ParCASum), and the proposed fusion based on GMU (ParCAGMU). Also, we report the performance of models using a single attention head as in the reference model and multi head models with 8 heads (we tried other numbers of heads but results did not vary considerably).

Table 5.3: Binary classification results. F1-Score is reported.

F1-Score Binary-Task		
Method	Num.Heads	**F1**
HICCAP (Baseline)	1	0.7978
HICCAP	8	0.8126
HICCAP - ParCACon	1	0.7874
HICCAP - ParCASum	1	0.8009
HICCAP - ParCAGMU	1	**0.8335**
HICCAP – ParCACon	8	0.8012
HICCAP – ParCASum	8	0.7898
HICCAP – ParCAGMU	8	0.8146

From Table 5.3, it can be seen that overall the reference model with ParCA mechanism outperformed the standard HICCAP (baseline) by 3.21% (absolute) in terms of F1 measure when using the GMU fusion in the binary task. This shows

the effectiveness of the proposed mechanism in modeling multimodal interaction. There are only two results out of 6 that did not improve the reference model. In terms of the fusion strategy, GMU obtained consistently better results than sum and concatenation. Interestingly, adding multiple attention heads into the baseline model improved its performance by almost 2%, but adding more layers to the models based on ParCA did not result in a consistent improvement.

Table 5.4: Multi-class classification results. F1-Score for each class and Average Macro-F1 across all classes are reported.

F1-Score Multi-Task						
Method	**Num. Heads**	**Mature**	**Gory**	**Slasptick**	**Sarcasm**	**Macro**
HICCAP (Baseline)	1	0.6191	0.1769	0.3853	0.7411	0.4806
HICCAP	8	0.5910	0.1831	**0.4859**	0.7388	0.4997
HICCAP - ParCACon	1	0.6197	0.1917	0.3957	0.7455	0.4882
HICCAP - ParCASum	1	**0.6480**	0.2814	0.1118	0.1400	0.2953
HICCAP - ParCAGMU	1	0.5790	0.2829	0.3152	**0.7486**	0.4814
HICCAP – ParCACon	8	0.5717	0.2942	0.3611	0.7391	0.4915
HICCAP – ParCASum	8	0.6250	0.2095	0.3828	0.7395	0.4892
HICCAP – ParCAGMU	8	0.5677	**0.3190**	0.4797	0.7147	**0.5203**

Regarding the multilabel task, Table 5.4 shows that this time better results in Macro F1 were obtained with the multi-head version of the HICCAP with ParCA mechanism. This result suggests that multimodal interactions are more complex for this problem, requiring of more attention heads to better model the problem. Also, please note that different variants obtained the best results in each class.

5.3.1 Fusion Heads

In this section, we evaluate the effectiveness of the head weighting methods presented in Section 4.3.3 (Figure 4.5). Three different methods were implemented for the combination of attention heads: Concatenation, Pruning, and GMU. Each modality was tested with the above methods: HICCAP - HCA[1], HICCAP - ParCACon, HICCAP - ParCASum, and HICCAP - ParCAGMU.

Table 5.5 shows the F1 scores in binary task for three different ways of weighting heads, with the number of heads set to 8. The table compares the performance of different methods (HICCAP - HCA, HICCAP - ParCACon, HICCAP - ParCASum, HICCAP - ParCAGMU) under three head fusion approaches: Concatenation, Pruning and GMU.

With the concatenation method, it is observed that the HICCAP - ParCAGMU method obtains the highest F1 score. This suggests that, for the concatenation technique, the ParCAGMU approach has a slight superior performance compared to the other methods.

In the GMU technique, it stands out that the HICCAP - ParCASum method achieves the highest F1 score in the whole table. The pure HICCAP method has the lowest score in this section. This suggests that, under the GMU technique, the ParCA-based methods significantly outperform the unmodified HICCAP method.

However, the use of pruning is where the lowest results are obtained in general, highlighting the results obtained in the original study (Voita et al., 2019), where it is found that the method is more effective when training with all the heads and in the fine-tuning process it is better to pruning.

Although the effectiveness of the methods varies considerably depending on the

[1]It is important to mention that HICCAP - HCA is using the baseline module HCA, not the model, the module was increased by eight times for this experiments. Also the ParCA modules.

Table 5.5: F1-Score in binary task for three different ways of weighting heads. Number of heads was set to 8.

Fusion Heads	Method	F1
Concatenation	HICCAP	0.8126
	HICCAP - ParCACon	0.8012
	HICCAP - ParCASum	0.7898
	HICCAP - ParCAGMU	0.8146
Pruning	HICCAP	0.7720
	HICCAP - ParCACon	0.7629
	HICCAP - ParCASum	0.7645
	HICCAP - ParCAGMU	0.6948
GMU	HICCAP	0.7937
	HICCAP - ParCACon	0.8247
	HICCAP - ParCASum	**0.8286**
	HICCAP - ParCAGMU	0.8228

head fusion technique used. The GMU technique appears to be the most promising in terms of achieving the highest F1 score, especially when combined with the ParCASum method.

Table 5.6 shows the F1 scores in multi-task task for the same three ways of weighting heads, with the number of heads set to 8. The methods compared are the same as in the previous task evaluated in four categories (Mature, Gory, Slapstick, Sarcasm) and an overall Macro-F1.

In the concatenation method, the HICCAP - ParCASum method excels in the Mature category, while HICCAP - ParCAGMU leads in Gory. HICCAP - ParCAGMU also obtains the best score in Slapstick. In terms of the Macro-F1

Table 5.6: F1-Score in multi-task for three different ways of weighting heads. Number of heads was set to 8.

Fusion Heads	Method	Mature	Gory	Slapstick	Sarcasm	Macro-F1
Concatenation	HICCAP	0.5910	0.1831	**0.4859**	0.7388	0.4997
	HICCAP - ParCACon	0.5717	0.2942	0.3611	0.7391	0.4915
	HICCAP - ParCASum	**0.6520**	0.2095	0.3828	0.7395	0.4892
	HICCAP - ParCAGMU	0.5677	**0.3190**	0.4797	0.7147	**0.5203**
Pruning	HICCAP	0.6215	0.1467	0.3787	0.7290	0.4690
	HICCAP - ParCACon	0.5128	0.1478	0.3347	0.6665	0.4155
	HICCAP - ParCASum	0.5369	0.2345	0.4128	0.6985	0.4708
	HICCAP - ParCAGMU	0.4450	0.0814	0.2882	0.6262	0.3602
GMU	HICCAP	0.5718	0.2701	0.4331	**0.7429**	0.5045
	HICCAP - ParCACon	0.2726	0.1746	0.2410	0.4551	0.2858
	HICCAP - ParCASum	0.4705	0.1895	0.3312	0.6321	0.4058
	HICCAP - ParCAGMU	0.5457	0.2838	0.4578	0.7036	0.4977

score, HICCAP - ParCAGMU is the highest, suggesting that for the concatenation technique, this method is the most effective overall.

From this table we can see the trend of pruning not having an improvement over the other fusion methods, however, it obtains results not so far from the baseline.

In the GMU technique, HICCAP scores the highest in the Sarcasm category. This suggests that the pure HICCAP method is quite robust in the GMU technique, especially in categories where sarcasm and slapstick are prominent.

Analyzing the data globally, it is observed that the performance of HICCAP - ParCAGMU is superior in the concatenation technique against the GMU technique, showing the best overall Macro-F1. This may be due to different reasons, our hypothesis is that when implementing the ParCAGMU module it already contains the necessary information and, not being a deep model, concatenation offers a simple

fusion unlike GMU which may have an overload of parameters, leading to lower performance.

5.3.2 Transformer-based Models

On the other hand, we performed experiments on a deep model, following the architecture of Section 4.3.3 (Figure 4.6). The experiments were performed on the same dataset and with the same evaluation metrics in both tasks. The evaluation was performed using a configuration with 8 attention heads as well as 1 and 5 encoders, ensuring good performance at different depth levels while varying head weighting techniques.

Table 5.7 presents the F1-Score results on the binary task using an encoder-based model with the three head weighting methods: Concatenation, Pruning and GMU.

In the Concatenation method, it is observed that the HICCAP - ParCA-Con method achieves the highest F1-Score, outperforming both HICCAP and the other ParCA variants. While in the Pruning and GMU methods the HICCAP - ParCAGMU method achieves the best result in both cases. This suggests that the ParCAGMU method is more effective in combining heads compared to the other two methods.

Overall, the Pruning with HICCAP - ParCAGMU strategy emerges as the most effective, this can be attributed to the reduction of redundancy and an effective combination of selected heads of care.

In Table 5.8 we can see the results using 5 encoders, with the same head merging techniques. From this table we can see that the results vary from the previous table.

Mainly, the HICCAP - ParCAGMU method is the one that obtains worse

Table 5.7: F1-Score in binary task for three different ways of weighting heads using encoder-based model

Fusion Heads	Method	Num. Heads/Encs	F1
Concatenation	HICCAP	8/1	0.8069
	HICCAP - ParCACon	8/1	0.8152
	HICCAP - ParCASum	8/1	0.8124
	HICCAP - ParCAGMU	8/1	0.7634
Pruning	HICCAP	8/1	0.8140
	HICCAP - ParCACon	8/1	0.8114
	HICCAP - ParCASum	8/1	0.8009
	HICCAP - ParCAGMU	8/1	**0.8187**
GMU	HICCAP	8/1	0.8102
	HICCAP - ParCACon	8/1	0.8093
	HICCAP - ParCASum	8/1	0.7853
	HICCAP - ParCAGMU	8/1	0.8096

results in Pruning and GMU techniques, while with the HICCAP - ParCASum Concatenation technique is the one that obtained the best performance in general.

This improvement can be attributed to a number of factors including a higher representation capacity thanks to the use of 5 encoders, a deeper and more diversified attention, the effectiveness of the concatenation fusion technique, and the specific improvements introduced by the HICCAP - ParCASum method. While the decline of the HICCAP - ParCAGMU method may be due to a combination of increased complexity, incompatibility of GMU with complex environments, negative impacts of pruning on feature representation, and challenges in multimodal information fusion.

Regarding the Multi-Task classification, Tables 5.9 and 5.10 present the F1-

Table 5.8: F1-Score in binary task for three different ways of weighting heads using encoder-based model

Fusion Heads	Method	Num. Heads/Encs	F1
Concatenation	HICCAP	8/5	0.8037
	HICCAP - ParCACon	8/5	0.8274
	HICCAP - ParCASum	8/5	**0.8453**
	HICCAP - ParCAGMU	8/5	0.7302
Pruning	HICCAP	8/5	—-
	HICCAP - ParCACon	8/5	0.8158
	HICCAP - ParCASum	8/5	0.6299
	HICCAP - ParCAGMU	8/5	0.5918
GMU	HICCAP	8/5	0.8045
	HICCAP - ParCACon	8/5	0.8078
	HICCAP - ParCASum	8/5	0.8056
	HICCAP - ParCAGMU	8/5	0.7985

Score results using an encoder-based model. Table 5.9 shows the results obtained with a single encoder, while Table 5.10 presents the results with five encoders. The categories evaluated are Mature, Gory, Slapstick, Sarcasm, as well as Macro-F1, and different methods and configurations are compared.

From Table 5.9, we can see that there is no one method that is better in all categories, or at least in most of them, however, in the Pruning technique it is the HICCAP - ParCACon method that obtains the best average results (Macro-F1). Although this method did not obtain the best results per class, it is competitive with respect to the others, suggesting that the pruning method may be effective in deep models.

Table 5.9: F1-Score in multi-task for three different ways of weighting heads using encoder-based model.

Fusion Heads	Method	Num. Heads/Encs	Mature	Gory	Slapstick	Sarcasm	Macro-F1
Concatenation	HICCAP	8/1	**0.6317**	0.1073	0.4508	0.7405	0.4828
	HICCAP - ParCACon	8/1	0.6038	0.2407	0.3621	0.7548	0.4904
	HICCAP - ParCASum	8/1	0.5921	0.2550	0.4226	0.7467	0.5041
	HICCAP - ParCAGMU	8/1	0.5988	0.1615	0.3914	0.7222	0.4685
Pruning	HICCAP	8/1	0.6105	0.1450	0.4030	0.7291	0.4719
	HICCAP - ParCACon	8/1	0.6168	**0.2627**	0.4479	0.7539	**0.5203**
	HICCAP - ParCASum	8/1	0.5988	0.2484	0.4567	**0.7559**	0.5124
	HICCAP - ParCAGMU	8/1	0.6055	0.2418	0.3279	0.7209	0.4740
GMU	HICCAP	8/1	0.6230	0.1888	0.4102	0.7262	0.4871
	HICCAP - ParCACon	8/1	0.6260	0.1433	0.3838	0.7217	0.4687
	HICCAP - ParCASum	8/1	0.5986	0.2126	**0.4878**	0.7136	0.5032
	HICCAP - ParCAGMU	8/1	0.6088	0.1901	0.4148	0.7323	0.4865

In Table 5.10, with five encoders, the results show improvements in a couple of categories over the previous table.

For example, an improvement is obtained in Gory using the ParCASum method with the GMU fusion method, while in Mature an improvement was obtained using the HCA method and the GMU technique. The rest did not obtain improvements but remain close to the previous results. It is noteworthy that the improvement was obtained using the GMU fusion technique with the ParCASum method.

In general, the results indicate that the use of multiple encoders tends to improve performance in some of the categories, obtaining a better result on average. However, the encoder-based model significantly increases the computational cost, increasing the processing time by up to 8 hours compared to the base model that only uses a single layer. For example, when using 8 heads compared to the base model, the processing time increases by approximately four hours. Despite this considerable increase in time and resources required, the cost/benefit is favorable. This

Table 5.10: F1-Score in multi-task for three different ways of weighting heads using encoder-based model.

Fusion Heads	Method	Num. Heads/Encs	Mature	Gory	Slapstick	Sarcasm	Macro-F1
Concatenation	HICCAP	8/5	0.3648	0.0500	0.1719	0.3528	0.2349
	HICCAP - ParCACon	8/5	0.5973	0.2587	0.3489	**0.7525**	0.4893
	HICCAP - ParCASum	8/5	0.6022	0.3546	0.3435	0.7178	0.5030
	HICCAP - ParCAGMU	8/5	0.5717	0.3540	0.3569	0.6747	0.4893
Pruning	HICCAP	8/5	0.3648	0.0500	0.1719	0.3528	0.2349
	HICCAP - ParCACon	8/5	0.5701	0.2054	0.3506	0.7512	0.4693
	HICCAP - ParCASum	8/5	0.5714	0.1961	**0.3846**	0.7097	0.4654
	HICCAP - ParCAGMU	8/5	0.5580	0.3091	0.3679	0.6048	0.4825
GMU	HICCAP	8/5	**0.6416**	0.1519	0.3474	0.7454	0.4716
	HICCAP - ParCACon	8/5	0.5712	0.1884	0.3434	0.7295	0.4581
	HICCAP - ParCASum	8/5	0.6167	**0.4443**	0.3762	0.6944	**0.5342**
	HICCAP - ParCAGMU	8/5	0.5907	0.3201	0.3746	0.7089	0.5074

increase in computational cost translates into more accurate performance, improving the model's ability to generalize and adapt to different contexts and data.

5.3.3 Analysis of results

In this section, we present a detailed analysis of the results obtained by applying different configurations and head fusion techniques for a binary and multi-task. The results varied significantly depending on the number of modules, the use of encoders and the fusion techniques employed. Also, as comic mischief dataset. Furthermore, as the comic mischief dataset has only been tested with the HICCAP model we limited ourselves to direct comparison, although in the original method they were compared with other multimodal approaches, HICCAP was able to improve performance over the rest.

Binary Task:

Table 5.11 shows the summarized results for the binary task of all the previous experiments. We can notice that the use of a single ParCAGMU module without encoders shows a high F1-Score. The simplicity of this configuration can be beneficial to avoid overcomplication of the model, which can sometimes lead to better performance on specific tasks.

Table 5.11: The best results for each of the different models for binary task. *None* is specified when that attribute does not apply.

Method	Multi-head Fusion Type	Num. Heads/Encs.	F1
HICCAP - ParCAGMU	*None*	1/*None*	**0.8335**
HICCAP - ParCASum	GMU	8/*None*	0.8286
HICCAP - ParCAGMU	Pruning	8/1	0.8187
HICCAP - ParCASum	Concatenation	8/5	0.8334

Using multiple heads and different fusion techniques did not improve the result significantly compared to using a single head. This could be because simply adding more heads without encoders does not provide a much richer rendering capability, and the concatenation technique may not be the most effective technique for combining information from these multiple heads.

Despite this, when using the encoder-based models, a better result was obtained in the deeper model using the ParCASum module with the GMU fusion technique. The use of this configuration probably allowed a better representation and capture of the relevant features in the data.

Multi-Task:

In multi-tasking classification, we summarize the results of previous experiments in Table 5.12. Here, the Macro-F1 results when using different configurations and head fusion techniques show variations that merit in-depth analysis.

The use of the ParCAGMU module augmented up to eight times obtained equal performance (on average) to the use of the ParCACon module with the encoder-based model using the Pruning fusion method. While these results are equal, we can note that for the former method, although concatenation can combine information from multiple heads, it does not necessarily optimize the combination of features to improve overall performance on all tasks.

Table 5.12: The best results in the different models for multi task. *None* is specified when that attribute does not apply.

Method	Multi-head Fusion Type	Num. Heads/Encs.	Mature	Gory	Slapstick	Sarcasm	Macro
HICCAP - ParCAGMU	Concatenation	8/*None*	0.5677	0.3190	0.4797	0.7147	0.5203
HICCAP - ParCACon	Pruning	8/1	0.6168	0.2627	0.4479	0.7549	0.5203
HICCAP - ParCASum	GMU	8/5	0.6167	0.4443	0.3762	0.6944	**0.5342***[2]

While in the second configuration, the use of a single encoder, together with pruning, may not provide a significant advantage in head fusion for this specific configuration. It is possible that a single encoder may not be sufficient to effectively capture and combine the relevant information from all heads.

Finally, the GMU head fusion method implemented to the encoder-based model performed the best of all, as shown in Table 5.12, this can be attributed to better feature representation and that the GMU technique provides a more sophisticated and dynamic combination of information from multiple heads.

[2]Using the Friedman test to evaluate whether there were statistically significant differences between the models. No significant differences were found, suggesting that the performance of the three models is comparable.

Overall, the results indicate that the use of GMU really works and is particularly effective in binary and multi-task classification, because of its ability to combine different representations in a weighted and dynamic way makes this technique effective. Not only at the multi-head level, but also within the ParCA module.

5.3.4 Effect of GMU

Here we will perform a detailed analysis on the use of GMU in the classification stage for the different categories using various modules. This analysis aims to evaluate the relevance of each of the final representations per modality in the classification of each category.

In Table 5.13, the average values of the σ gates per modality and category are presented. These results allow us to observe how GMU assigns different weights to the modalities depending on the method used. For example, in the "Gory" and "Slapstick" categories, the HCA module gives less importance to the visual part. This is somewhat counterintuitive, as the samples in both categories are predominantly visual. This undervaluation of the visual modality could be the cause of the poor classification performance of these categories when using the HCA module.

In contrast, the ParCACon module gives greater weight to the visual modality for the classification of these specific classes. However, for the rest of the categories, ParCACon maintains a more balanced approach, assigning a more balanced weight to each modality.

The ParCAGMU module, which obtained the best overall classification results, shows an interesting trend. For the "Mature" category, ParCAGMU assigns greater weight to Audio and Image information. This makes sense, since the topics covered in this class (drugs, sexual topics, alcoholism, etc.) are associated with facial expressions and tone of voice.

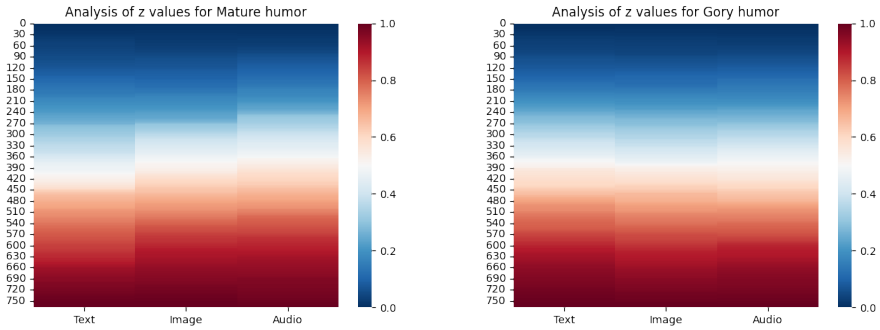
Table 5.13: Analysis of the use of GMU module for each category at the classification stage using the four different modules. Best results per category are in bold and best results per row are in italic. 0.0 and 1.0 values means the lowest and the highest importance for each category, respectively.

Method	**Modalities**	**Value of z**			
		Mature	Gory	Slapstick	Sarcasm
HCA	**Text**	***1.0***	***1.0***	0.6209	***1.0***
	Image	0.0	0.0	0.0	*0.6599*
	Audio	0.8067	0.2508	***1.0***	0.0
ParCACon	**Text**	***1.0***	0.0	0.0	0.8347
	Image	0.0	***1.0***	***1.0***	***1.0***
	Audio	0.1286	*0.6565*	0.004	0.0
ParCASum	**Text**	0.7112	***1.0***	0.8350	0.0
	Image	***1.0***	0.4118	***1.0***	***1.0***
	Audio	0.0	0.0	0.0	*0.9784*
ParCAGMU	**Text**	0.0	***1.0***	0.0	0.4207
	Image	0.5794	0.0	***1.0***	***1.0***
	Audio	***1.0***	0.4356	0.9455	0.0

On the other hand, in the "Gory" and "Slapstick" categories, ParCAGMU assigns nearly equal weights to each modality. Although this might seem unfavorable, given that these classes might be dominated by a single modality, this balanced approach can be beneficial. By avoiding bias toward a single modality, GMU may be compensating to improve overall model performance across all classifications.

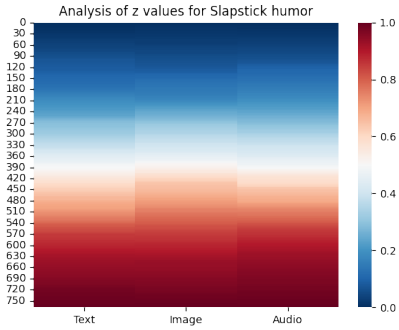
Figure 5.4 shows the heatmaps by category of the ParCAGMU module. These heat maps provide a clearer view of how GMU determines the relevance of each

modality in the classification process.

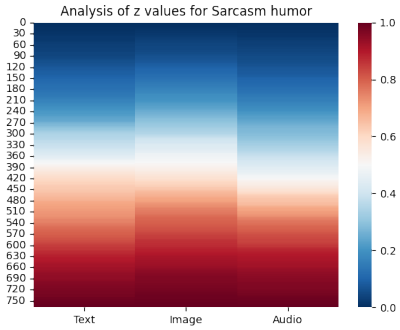


(a) Mature humor

(b) Gory humor



(c) Slapstick humor



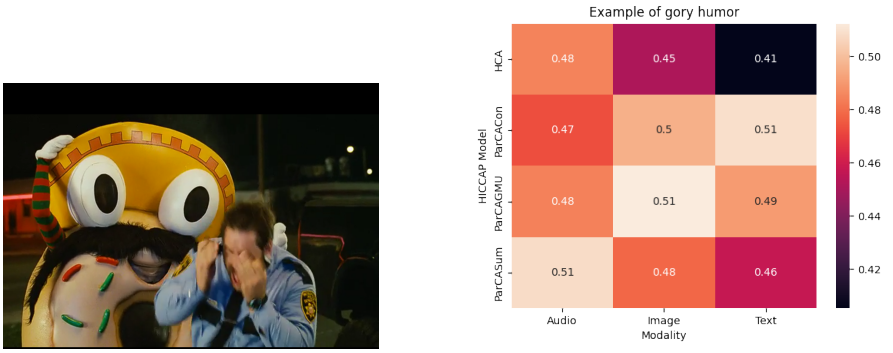
(d) Sarcasm humor

Figure 5.4: Heat-maps of the relevance of each modality in the classification stage for each category using the GMU fusion in Parallel Cross-Attention module (ParCAGMU). Each map has values from 0 to 756 which is the dimension of each input vector.

By looking at these heat maps, we can notice the different values that each σ input takes on and how these values influence the importance that GMU assigns to each modality. The heat maps reveal specific patterns, highlighting the most essential parts for each category. This allows us to better understand how GMU tailors its approach depending on the category and modality, optimizing the ranking process by focusing on the most relevant elements.

Example of z values

Here, we present a specific example involving gory and slapstick humor that is in comic mischief dataset. Figure 5.5a shows a frame extracted from a clip showing kicks or punches, and the frame shown is a policeman sprayed with pepper spray.



(a) An example of gory humor in comic mischief dataset.

(b) Heatmap of each module in the HICCAP model for this specific example.

Figure 5.5: Heatmap of the relevance of each modality in the classification stage for each module in HICCAP model.

Figure 5.5b shows σ values for each modulea at the classification stage. We can notices that these σ values vary significantly between different modules and modalities, and as this is clip contains more visual elements image modality could have more weight. For instance, with ParCASum module audio modality has more weight than the others, and GMU is able to dynamically adjust the importance of each modality according to the unique characteristics of this instance, as GMU works at an instance level.

In summary, these results demonstrate ability of GMU to dynamically adapt to the relevance of each modality according to the specific needs of each category, thus optimizing the multimodal classification process.

5.4 Results in Additional Datasets

In this section we will analyze the effectiveness of using different evaluated modules on different datasets, as well as on various tasks. It is important to examine how these modules behave in different scenarios to understand their versatility and robustness. In addition, these datasets contain multimodal information, which is the central purpose of this work.

For this study we used a tool called MultiBench. MultiBench offers features of multimodal datasets previously collected and pre-processed from different datasets to be able to work with them independently or with vanilla models that it also offers. In our case, we specifically worked with the CMU-MOSI and CMU-MOSEI datasets as they offer text, audio and video modalities.

These datasets are benchmarks commonly used to evaluate various methods. They consist of opinion videos extracted from YouTube, and the main task associated with this data is sentiment analysis. Each video is evaluated on a scale of [-3, 3], where each value indicates a different emotion. As baselines we used the model proposed by Hazarika et al. (2020) (MISA) and the model proposed by Tsai et al. (2019) (MulT).

Both MOSI and MOSEI are primarily regression tasks using the *mean absolute error* (MAE). In addition, the benchmark also includes classification scores covering *seven-class accuracy (Acc-7)* with a range of -3 to 3, *binary accuracy (Acc-2)* and *F-Score*. For binary accuracy scores, two different approaches have been considered in the past. The first is *negative/non-negative* classification, where labels for non-negatives are based on scores ≥ 0 (Zadeh et al., 2018a). In recent work, binary accuracy is calculated using a more precise *negative/positive* class formulation, where negative and positive classes are assigned for sentiment scores < 0 and > 0, respectively (Tsai et al., 2019). We report results on both metrics using the seg-

mentation marker -/-, where the score on the left-hand side is for the *neg./non-neg.* classification, while the score on the right-hand side is for the *neg./pos.* classification.

In addition, a pre-training was performed on other MultiBench datasets, UR-FUNNY (Hasan et al., 2019) and MUStARD (Castro et al., 2019), using the matching tasks method in Figure 4.3. In these results, pre-training was performed on both datasets and subsequently fine-tuning was performed on the corresponding dataset.

Table 5.14 presents the results obtained for the CMU-MOSI dataset. In this particular case, we employed an encoder-based model with an 8-head configuration and a single encoder. In addition, we used the GMU technique for the fusion of the different modules. Due to the number of instances contained in the CMU-MOSI dataset, we considered that this configuration would be the most suitable to handle the volume of data.

Table 5.14: Results for CMU-MOSI dataset using Mean Absolute Error (MAE), Accuracy top 2, Accuracy top 7 and F1 Score metrics. ⊗ from Tsai et al. (2019).

Fusion Heads	Method	Num. Heads/Encs.	Pre-Trained	MAE (↓)	Acc-2 (↑)	F1 (↑)	Acc-7 (↑)
GMU	HCA	8/1	No	0.8154	0.8000 / 0.8174	0.7999 / 0.8180	0.4084
	HCA	8/1	Yes	**0.8065**	0.7991 / 0.8159	0.7992 / 0.8165	0.4131
	ParCACon	8/1	No	0.8093	0.8020 / 0.8216	0.8019 / 0.8222	0.4087
	ParCACon	8/1	Yes	0.8223	0.7950 / 0.8116	0.7949 / 0.8121	0.4050
	ParCASum	8/1	No	0.8084	**0.8085 / 0.8259**	**0.8083 / 0.8264**	**0.4134**
	ParCASum	8/1	Yes	0.8166	0.7986 / 0.8152	0.7984 / 0.8157	0.4114
	ParCAGMU	8/1	No	0.8090	0.8067 / 0.8238	0.8067 / 0.8243	0.4076
	ParCAGMU	8/1	Yes	0.8245	0.8038 / 0.8219	0.8035 / 0.8223	0.4044
	MFM⊗ (Tsai et al., 2018)			0.951	0.7810 / -	0.7810 / -	0.3620
	RAVEN⊗ (Wang et al., 2019)			0.9150	0.780 / -	0.7660 / -	0.3220
	RMFN⊗ (Liang et al., 2018)			0.9220	0.7840 / -	0.780 / -	0.3830
	MCTN⊗ (Pham et al., 2019)			0.9090	0.7930 / -	0.7910 / -	0.3560
	MulT (Tsai et al., 2019)			0.8710	- / 0.830	- / 0.8280	0.40
	MISA (Hazarika et al., 2020)			0.7830	0.8180 / 0.8340	0.8170 / 0.8360	0.4230

From the results obtained, we can observe that the use of the pre-trained model

actually led to worse results compared to the results obtained by training the model from scratch. This suggests that, in this particular case, the pre-trained model did not provide significant advantages and, in fact, may have limited the performance of the system.

However, when analyzing the results of the ParCASum module, we can see that this module obtains good results when trained from scratch, outperforming the rest of the methods evaluated in this study. This indicates that ParCASum has significant potential when allowed to learn and adapt from scratch.

Table 5.15 presents the results obtained for the CMU-MOSEI dataset with a similar configuration as in the previous table, just changing the number of ecnoders from one to five for this dataset due to the number of instances in it.

Table 5.15: Results for CMU-MOSEI dataset using Mean Absolute Error (MAE), Accuracy top 2, Accuracy top 7 and F1 Score metrics. ⊗ from Tsai et al. (2019).

Fusion Heads	Method	Num. Heads/Encs.	Pre-Trained	MAE (↓)	Acc-2 (↑)	F1 (↑)	Acc-7 (↑)
GMU	HCA	8/5	No	0.5476	0.8099 / 0.8533	0.8159 / 0.8536	0.5215
	HCA	8/5	Yes	**0.5425**	**0.8131 / 0.8577**	**0.8190 / 0.8580**	0.5240
	ParCACon	8/5	No	0.5446	0.8075 / 0.8517	0.8138 / 0.8522	0.5206
	ParCACon	8/5	Yes	0.5439	0.8075 / 0.8526	0.8138 / 0.8531	0.5231
	ParCASum	8/5	No	0.5432	0.8072 / 0.8518	0.8135 / 0.8523	0.5215
	ParCASum	8/5	Yes	0.5434	0.8092 / 0.8534	0.8154 / 0.8538	0.5201
	ParCAGMU	8/5	No	0.5430	0.8112 / 0.8546	0.8172 / 0.8549	**0.5261**
	ParCAGMU	8/5	Yes	0.5479	0.8084 / 0.8510	0.8145 / 0.8514	0.5197
	RAVEN⊗ (Wang et al., 2019)			0.6140	0.7910 / -	0.7950 / -	0.50
	MCTN⊗ (Pham et al., 2019)			0.6090	0.7980 / -	0.8060 / -	0.4960
	Graph-MFN⊗ (Zadeh et al., 2018b)			0.710	0.7690 / -	0.770 / -	0.450
	MulT (Tsai et al., 2019)			0.580	- / 0.8250	- / 0.8230	0.5180
	MISA (Hazarika et al., 2020)			0.5550	0.8360 / 0.8550	0.8380 / 0.8530	0.5220

From the results obtained, we can observe that, in this particular case, the use of the pre-trained model led to good results compared to the results obtained by training the model from scratch. This improvement is especially noticeable when

using the HCA module. The results suggest that the pre-trained model offers significant advantages in terms of performance when integrated with this specific module.

In fact, when comparing the results with the baseline MISA, a noticeable improvement is observed in the MSE and Acc-7 metrics. In addition, when analyzing the Acc-7 and F1 Score metrics in the binary classification of positive/negative (right side), the best improvements were also obtained. These improvements indicate that the use of the pre-trained model has not only optimized the accuracy of the system, but also increased its ability to correctly classify in more challenging and diverse scenarios.

On the other hand, these similar results between MISA and HICCAP-HCA models can be attributed to the way both of them process the information. While MISA takes modality-invariant and -specific sub-spaces to process the modalities and to obtain six different representations (two per modality), HICCAP-HCA focus on process each modality jointly with the other two obtaining one final representation per modality, leading to a less complex sub-spaces.

Despite these results, it is important to note that this is not completely conclusive. There are methods in the state of the art that still show superior performance, such as the baseline MISA method for the CMU-MOSI dataset. Nevertheless, our model stills competitive and open to exploration and optimization to reach or even surpass the performance levels of the different methods in the state of the art.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

In this study, we addressed the problem of detecting questionable content in videos using multimodal information. We focused specifically on the detection of humorous pranks, a crucial task because these jokes may cross the line into inappropriate or dangerous behaviors. The diverse and complex nature of questionable content requires a detailed analysis of multiple modalities such as audio, text and images, allowing us a more complete and accurate representation of the videos. For this, we developed an innovative module called Parallel Cross-Attention (ParCA), which simultaneously handles three modalities and allows for parallel and equal integration, capturing details that could be missed if analyzed in isolation.

Our proposal represents a significant advance over previous methods such as Hierarchical Cross-Attention (HCA), as ParCA operates in a more coordinated and simultaneous manner. This improves efficiency and accuracy in detecting questionable content by reducing information loss and improving consistency in content interpretation. In addition, we introduced a new way of combining multimodal information, tailored to improve this detection by dynamically adjusting the weights of each modality according to the context of the video. This approach not only improves the identification of comical pranks, but also of other types of questionable content, providing a useful tool for video moderation on various platforms.

Moreover, we explored transformer-based architectures to evaluate several variants of Multihead Attention-Based Models (MAMBs) in terms of depth, number of heads, and fusion schemes. The results revealed that ParCA significantly outperformed HCA in detection, highlighting its ability to capture complex interactions between modalities. GMU adoption also proved to be more effective than conventional multimodal fusion approaches.

The encoder-based model demonstrated general effectiveness, although its performance may vary depending on the depth and specific architecture required for each task. We evaluated our approach on several MultiBench datasets, identifying both strengths and potential areas for improvement over the current state of the art.

The conclusions of this contribution are as follows:

- ParCA outperformed HCA in the detection of comic mischief, suggesting our proposed mechanism better captures the interaction across modalities.
- The use of a GMU outperformed the standard concatenation for the fusion of multimodal attention mechanisms.
- The ParCA module adapts well to different tasks and datasets with competitive results.
- The GMU module actually captures different dependencies from different modalities.
- The usefulness of adding multiple encoders into the reference model was not clear in the different datasets.
- The usefulness of pre-training approach is not clear for encoder-based models.
- The relevance of each modality is important when classifying the type of humor, since in certain instances one modality may predominate more than another.

As future work, we plan to extend our testing to datasets that address other types of questionable content, such as hate speech and violence detection, in both binary classification and multi-task classification tasks. We also intend to conduct a more detailed comparison with leading state-of-the-art models that handle more than two modalities, trying to further improve our approach to achieve even better and generalizable performance. In addition, we would like to develop explainable models for the detection of different types of questionable content.

References

Ahmed, K., Keskar, N. S., and Socher, R. (2017). Weighted transformer network for machine translation. *arXiv preprint arXiv:1711.02132*.

Alammar, J. (2018a). The illustrated bert, elmo and co. (how nlp cracked transfer learning) [blog post]. Retrieved from https://jalammar.github.io/illustrated-bert/. Accessed: 2023-01-23.

Alammar, J. (2018b). The illustrated transformer [blog post]. Retrieved from https://jalammar.github.io/illustrated-transformer/. Accessed: 2023-01-23.

Anderson, C. A., Berkowitz, L., Donnerstein, E., Huesmann, L. R., Johnson, J. D., Linz, D., Malamuth, N. M., and Wartella, E. (2003). The influence of media violence on youth. *Psychological science in the public interest*, 4(3):81–110.

Anwar, A., Kanjo, E., and Anderez, D. O. (2022). Deepsafety: Multi-level audio-text feature extraction and fusion approach for violence detection in conversations. *arXiv preprint arXiv:2206.11822*.

Arevalo, J., Solorio, T., Montes-y Gomez, M., and González, F. A. (2020). Gated multimodal networks. *Neural Computing and Applications*, 32:10209–10228.

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

Bagher Zadeh, A., Liang, P. P., Poria, S., Cambria, E., and Morency, L.-P. (2018). Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In Gurevych, I. and Miyao, Y., editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2236–2246, Melbourne, Australia. Association for Computational Linguistics.

Baharlouei, E. and Solorio, T. (2024). Labeling comic mischief content in online videos. In *LREC-COLING*, volume In press.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bhojanapalli, S., Chakrabarti, A., Jain, H., Kumar, S., Lukasik, M., and Veit, A. (2021). Eigen analysis of self-attention and its reconstruction from partial computation. *arXiv preprint arXiv:2106.08823*.

Bridges, A. J., Wosnitzer, R., Scharrer, E., Sun, C., and Liberman, R. (2010). Aggression and sexual behavior in best-selling pornography videos: A content analysis update. *Violence against women*, 16(10):1065–1085.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Bushman, B. J. and Anderson, C. A. (2009). Comfortably numb: Desensitizing effects of violent media on helping others. *Psychological science*, 20(3):273–277.

Carreira, J. and Zisserman, A. (2017). Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308.

Castro, S., Hazarika, D., Pérez-Rosas, V., Zimmermann, R., Mihalcea, R., and Poria, S. (2019). Towards multimodal sarcasm detection (an _Obviously_ perfect paper). In Korhonen, A., Traum, D., and Màrquez, L., editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4619–4629, Florence, Italy. Association for Computational Linguistics.

Chang, J. H. and Bushman, B. J. (2019). Effect of exposure to gun violence in video games on children's dangerous behavior with real guns: a randomized clinical trial. *JAMA network open*, 2(5):e194319–e194319.

Chang, L. Y., Mukherjee, S., and Coppel, N. (2021). We are all victims: Questionable content and collective victimisation in the digital age. *Asian journal of criminology*, 16(1):37–50.

Chen, X., Kang, B., Wang, D., Li, D., and Lu, H. (2022). Efficient visual tracking via hierarchical cross-attention transformer. In *European Conference on Computer Vision*, pages 461–477. Springer.

Chen, X., Yan, B., Zhu, J., Wang, D., Yang, X., and Lu, H. (2021). Transformer tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8126–8135.

Crawshaw, M. (2020). Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*.

Degardin, B. and Proença, H. (2020). Human activity analysis: Iterative weak/self-supervised learning frameworks for detecting abnormal events. In *2020 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–7. IEEE.

Dillon, K. P. and Bushman, B. J. (2017). Effects of exposure to gun violence in movies on children's interest in real guns. *JAMA pediatrics*, 171(11):1057–1062.

Dutta, S. and Ganapathy, S. (2023). Hcam–hierarchical cross attention model for multi-modal emotion recognition. *arXiv preprint arXiv:2304.06910*.

Gavrilyuk, K., Sanford, R., Javan, M., and Snoek, C. G. (2020). Actor-transformers for group activity recognition. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 839–848.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. In *International conference on machine learning*, pages 1243–1252. PMLR.

Guo, D., Ren, S., Lu, S., Feng, Z., Tang, D., Shujie, L., Zhou, L., Duan, N., Svyatkovskiy, A., Fu, S., , Tufano, M., Deng, S. K., Clement, C., Drain, D., Sundaresan, N., Yin, J., Jiang, D., and Zhou, M. (2020). Graphcodebert: Pre-training code representations with data flow. In *International Conference on Learning Representations*.

Hasan, M. K., Rahman, W., Bagher Zadeh, A., Zhong, J., Tanveer, M. I., Morency, L.-P., and Hoque, M. E. (2019). UR-FUNNY: A multimodal language dataset for understanding humor. In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2046–2056, Hong Kong, China. Association for Computational Linguistics.

Hazarika, D., Zimmermann, R., and Poria, S. (2020). Misa: Modality-invariant and-specific representations for multimodal sentiment analysis. In *Proceedings of the 28th ACM international conference on multimedia*, pages 1122–1131.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, R. C., Plakal, M., Platt, D., Saurous, R. A., Seybold, B., et al. (2017). Cnn architectures

for large-scale audio classification. In *ICASSP 2017-2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 131–135. IEEE.

Huang, L., Wang, W., Chen, J., and Wei, X.-Y. (2019). Attention on attention for image captioning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4634–4643.

Huesmann, L. R. (2007). The impact of electronic media violence: Scientific theory and research. *Journal of Adolescent health*, 41(6):S6–S13.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al. (2017). The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

Kenton, J. D. M.-W. C. and Toutanova, L. K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Kiela, D. and Bottou, L. (2014). Learning image embeddings using convolutional neural networks for improved multi-modal semantics. *Proceedings of the conference on empirical methods in natural languaje processng (EMNLP-14)*, pages 36–45.

Kim, J., El-Khamy, M., and Lee, J. (2020). T-gsa: Transformer with gaussian-weighted self-attention for speech enhancement. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6649–6653. IEEE.

Kim, Y., Denton, C., Hoang, L., and Rush, A. M. (2016). Structured attention networks. In *International Conference on Learning Representations*.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2020). BART: Denoising sequence-to-sequence

pre-training for natural language generation, translation, and comprehension. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J., editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Li, L. H., Yatskar, M., Yin, D., Hsieh, C.-J., and Chang, K.-W. (2019). Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Li, R., Yang, S., Ross, D. A., and Kanazawa, A. (2021). Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13401–13412.

Liang, P. P., Liu, Z., Zadeh, A., and Morency, L.-P. (2018). Multimodal language analysis with recurrent multistage fusion. *arXiv preprint arXiv:1808.03920*.

Lin, J., Yang, A., Zhang, Y., Liu, J., Zhou, J., and Yang, H. (2020). Interbert: Vision-and-language interaction for multi-modal pretraining. *arXiv preprint arXiv:2003.13198*.

Lin, T., Wang, Y., Liu, X., and Qiu, X. (2021). A survey of transformers. *arXiv preprint arXiv:2106.04554*.

Liu, T., Zhang, C., Lam, K.-M., and Kong, J. (2023). Decouple and resolve: Transformer-based models for online anomaly detection from weakly labeled videos. *IEEE Transactions on Information Forensics and Security*, 18:15–28.

Lu, J., Batra, D., Parikh, D., and Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.

Maheshwari, S. (2017). On youtube kids, startling videos slip past filters. Retrieved from https://www.nytimes.com/2017/11/04/business/media/youtube-kids-paw-patrol.html. Accessed: 2024-08-23.

Michel, P., Levy, O., and Neubig, G. (2019). Are sixteen heads really better than one? *Advances in neural information processing systems*, 32.

Nguyen, T., Nguyen, T., Do, H., Nguyen, K., Saragadam, V., Pham, M., Nguyen, K. D., Ho, N., and Osher, S. (2022). Improving transformer with an admixture of attention heads. *Advances in neural information processing systems*, 35:27937–27952.

Okamoto, T., Toda, T., Shiga, Y., and Kawai, H. (2020). Transformer-based text-to-speech with weighted forced attention. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6729–6733. IEEE.

OpenAI, :, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., Berdine, J., Bernadett-Shapiro, G., Berner, C., Bogdonoff, L., Boiko, O., Boyd, M., Brakman, A.-L., Brockman, G., Brooks, T., Brundage, M., Button, K., Cai, T., Campbell, R., Cann, A., Carey, B., Carlson, C., Carmichael, R., Chan, B., Chang, C., Chantzis, F., Chen, D., Chen, S., Chen, R., Chen, J., Chen, M., Chess, B., Cho, C., Chu, C., Chung, H. W., Cummings, D., Currier, J., Dai, Y., Decareaux, C., Degry, T., Deutsch, N., Deville, D., Dhar, A., Dohan, D., Dowling, S., Dunning, S., Ecoffet, A., Eleti, A., Eloundou, T., Farhi, D., Fedus, L., Felix, N., Fishman, S. P., Forte, J., Fulford, I., Gao, L., Georges, E., Gibson, C., Goel, V., Gogineni, T., Goh, G., Gontijo-Lopes, R., Gordon, J., Grafstein, M., Gray, S., Greene, R., Gross, J., Gu, S. S., Guo, Y., Hallacy, C., Han, J., Harris, J., He, Y., Heaton, M., Heidecke, J., Hesse, C., Hickey, A., Hickey, W., Hoeschele, P., Houghton, B., Hsu, K., Hu, S., Hu, X., Huizinga, J., Jain, S., Jain, S., Jang, J., Jiang, A., Jiang, R., Jin, H., Jin, D., Jomoto, S., Jonn, B., Jun, H., Kaftan, T., Łukasz Kaiser, Kamali, A., Kanitscheider, I., Keskar, N. S., Khan,

T., Kilpatrick, L., Kim, J. W., Kim, C., Kim, Y., Kirchner, H., Kiros, J., Knight, M., Kokotajlo, D., Łukasz Kondraciuk, Kondrich, A., Konstantinidis, A., Kosic, K., Krueger, G., Kuo, V., Lampe, M., Lan, I., Lee, T., Leike, J., Leung, J., Levy, D., Li, C. M., Lim, R., Lin, M., Lin, S., Litwin, M., Lopez, T., Lowe, R., Lue, P., Makanju, A., Malfacini, K., Manning, S., Markov, T., Markovski, Y., Martin, B., Mayer, K., Mayne, A., McGrew, B., McKinney, S. M., McLeavey, C., McMillan, P., McNeil, J., Medina, D., Mehta, A., Menick, J., Metz, L., Mishchenko, A., Mishkin, P., Monaco, V., Morikawa, E., Mossing, D., Mu, T., Murati, M., Murk, O., Mély, D., Nair, A., Nakano, R., Nayak, R., Neelakantan, A., Ngo, R., Noh, H., Ouyang, L., O'Keefe, C., Pachocki, J., Paino, A., Palermo, J., Pantuliano, A., Parascandolo, G., Parish, J., Parparita, E., Passos, A., Pavlov, M., Peng, A., Perelman, A., de Avila Belbute Peres, F., Petrov, M., de Oliveira Pinto, H. P., Michael, Pokorny, Pokrass, M., Pong, V., Powell, T., Power, A., Power, B., Proehl, E., Puri, R., Radford, A., Rae, J., Ramesh, A., Raymond, C., Real, F., Rimbach, K., Ross, C., Rotsted, B., Roussez, H., Ryder, N., Saltarelli, M., Sanders, T., Santurkar, S., Sastry, G., Schmidt, H., Schnurr, D., Schulman, J., Selsam, D., Sheppard, K., Sherbakov, T., Shieh, J., Shoker, S., Shyam, P., Sidor, S., Sigler, E., Simens, M., Sitkin, J., Slama, K., Sohl, I., Sokolowsky, B., Song, Y., Staudacher, N., Such, F. P., Summers, N., Sutskever, I., Tang, J., Tezak, N., Thompson, M., Tillet, P., Tootoonchian, A., Tseng, E., Tuggle, P., Turley, N., Tworek, J., Uribe, J. F. C., Vallone, A., Vijayvergiya, A., Voss, C., Wainwright, C., Wang, J. J., Wang, A., Wang, B., Ward, J., Wei, J., Weinmann, C., Welihinda, A., Welinder, P., Weng, J., Weng, L., Wiethoff, M., Willner, D., Winter, C., Wolrich, S., Wong, H., Workman, L., Wu, S., Wu, J., Wu, M., Xiao, K., Xu, T., Yoo, S., Yu, K., Yuan, Q., Zaremba, W., Zellers, R., Zhang, C., Zhang, M., Zhao, S., Zheng, T., Zhuang, J., Zhuk, W., and Zoph, B. (2023). Gpt-4 technical report.

Pang, W.-F., He, Q.-H., Hu, Y.-j., and Li, Y.-X. (2021). Violence detection in videos based on fusing visual and audio information. In *ICASSP 2021 - 2021 IEEE*

International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2260–2264.

Pei, D., Liu, H., Liu, Y., and Sun, F. (2013). Unsupervised multimodal feature learning for semantic image segmentation. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE.

Pham, H., Liang, P. P., Manzini, T., Morency, L.-P., and Póczos, B. (2019). Found in translation: Learning robust joint representations by cyclic translations between modalities. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6892–6899.

Raganato, A. and Tiedemann, J. (2018). An analysis of encoder representations in transformer-based machine translation. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: analyzing and interpreting neural networks for NLP*, pages 287–297.

Rahman, T., Yang, M., and Sigal, L. (2021). Tribert: Human-centric audio-visual representation learning. *Advances in Neural Information Processing Systems*, 34:9774–9787.

Rendón-Segador, F. J., Álvarez-García, J. A., Salazar-González, J. L., and Tommasi, T. (2023). Crimenet: neural structured learning using vision transformer for violence detection. *Neural networks*, 161:318–329.

Rodríguez Bribiesca, I., López Monroy, A. P., and Montes-y Gómez, M. (2021). Multimodal weighted fusion of transformers for movie genre classification. In Zadeh, A., Morency, L.-P., Liang, P. P., Ross, C., Salakhutdinov, R., Poria, S., Cambria, E., and Shi, K., editors, *Proceedings of the Third Workshop on Multimodal Artificial Intelligence*, pages 1–5, Mexico City, Mexico. Association for Computational Linguistics.

See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Shafaei, M., Smailis, C., Kakadiaris, I., and Solorio, T. (2021). A case study of deep learning-based multi-modal methods for labeling the presence of questionable content in movie trailers. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1297–1307, Held Online. INCOMA Ltd.

Shi, B., Hsu, W.-N., Lakhotia, K., and Mohamed, A. (2021). Learning audio-visual speech representation by masked multimodal cluster prediction. In *International Conference on Learning Representations*.

Solorio, T., Shafaei, M., Smailis, C., Diab, M., Giannakopoulos, T., Ji, H., Liu, Y., Mihalcea, R., Muresan, S., and Kakadiaris, I. (2021). White paper: Challenges and considerations for the creation of a large labelled repository of online videos with questionable content. *arXiv preprint arXiv:2101.10894*.

Srivastava, N. and Salakhutdinov, R. R. (2012). Multimodal learning with deep boltzmann machines. *Advances in neural information processing systems*, 25.

Suk, H.-I. and Shen, D. (2013). Deep learning-based feature representation for ad/mci classification. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013: 16th International Conference, Nagoya, Japan, September 22-26, 2013, Proceedings, Part II 16*, pages 583–590. Springer.

Sultani, W., Chen, C., and Shah, M. (2018). Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6479–6488.

Sun, C., Myers, A., Vondrick, C., Murphy, K., and Schmid, C. (2019). Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7464–7473.

Tan, H. and Bansal, M. (2019). Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111.

Tang, G., Sennrich, R., and Nivre, J. (2018). An analysis of attention mechanisms: The case of word sense disambiguation in neural machine translation. In Bojar, O., Chatterjee, R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Monz, C., Negri, M., Névéol, A., Neves, M., Post, M., Specia, L., Turchi, M., and Verspoor, K., editors, *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 26–35, Brussels, Belgium. Association for Computational Linguistics.

Tsai, Y.-H. H., Bai, S., Liang, P. P., Kolter, J. Z., Morency, L.-P., and Salakhutdinov, R. (2019). Multimodal transformer for unaligned multimodal language sequences. In *Proceedings of the conference. Association for computational linguistics. Meeting*, volume 2019, page 6558. NIH Public Access.

Tsai, Y.-H. H., Liang, P. P., Zadeh, A., Morency, L.-P., and Salakhutdinov, R. (2018). Learning factorized multimodal representations. *arXiv preprint arXiv:1806.06176*.

Vaswani, A., Bengio, S., Brevdo, E., Chollet, F., Gomez, A., Gouws, S., Jones, L., Kaiser, Ł., Kalchbrenner, N., Parmar, N., et al. (2018). Tensor2tensor for neural machine translation. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 193–199.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser,

Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Voita, E., Serdyukov, P., Sennrich, R., and Titov, I. (2018). Context-aware neural machine translation learns anaphora resolution. In Gurevych, I. and Miyao, Y., editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1264–1274, Melbourne, Australia. Association for Computational Linguistics.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., and Titov, I. (2019). Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Wang, Y., Shen, Y., Liu, Z., Liang, P. P., Zadeh, A., and Morency, L.-P. (2019). Words can shift: Dynamically adjusting word representations using nonverbal behaviors. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7216–7223.

Wei, D.-L., Liu, C.-G., Liu, Y., Liu, J., Zhu, X.-G., and Zeng, X.-H. (2022). Look, listen and pay more attention: Fusing multi-modal information for video violence detection. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1980–1984.

Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.-S., Cheng, M., Glaese, M., Balle, B., Kasirzadeh, A., et al. (2021). Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*.

Wilson, B. J. (2008). Media and children's aggression, fear, and altruism. *The future of children*, pages 87–118.

Wu, P., Liu, J., Shi, Y., Sun, Y., Shao, F., Wu, Z., and Yang, Z. (2020). Not only look, but also listen: Learning multimodal violence detection under weak supervi-

sion. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pages 322–339. Springer.

Xiao, Y., Wang, L., Wang, T., and Lai, H. (2023). Scoreformer: Score fusion-based transformers for weakly-supervised violence detection. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Xu, P. and Zhu, X. (2021). Deepchange: A large long-term person re-identification benchmark with clothes change. *arXiv preprint arXiv:2105.14685*.

Xu, P., Zhu, X., and Clifton, D. A. (2023). Multimodal learning with transformers: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yang, L., Wu, Z., Hong, J., and Long, J. (2022). Mcl: A contrastive learning method for multimodal data fusion in violence detection. *IEEE Signal Processing Letters*.

Yang, Z., Nakashima, Y., and Takemura, H. (2023). Multi-modal humor segment prediction in video. *Multimedia Systems*, 29(4):2389–2398.

Yoon, J., Kang, C., Kim, S., and Han, J. (2022). D-vlog: Multimodal vlog dataset for depression detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 12226–12234.

Zadeh, A., Liang, P. P., Poria, S., Vij, P., Cambria, E., and Morency, L.-P. (2018a). Multi-attention recurrent network for human communication comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Zadeh, A., Zellers, R., Pincus, E., and Morency, L.-P. (2016). Mosi: multimodal corpus of sentiment intensity and subjectivity analysis in online opinion videos. *arXiv preprint arXiv:1606.06259*.

Zadeh, A. B., Liang, P. P., Poria, S., Cambria, E., and Morency, L.-P. (2018b). Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable

dynamic fusion graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2236–2246.

Zaidi, S. A. M., Latif, S., and Qadi, J. (2023). Cross-language speech emotion recognition using multimodal dual attention transformers. *arXiv preprint arXiv:2306.13804*.

Zhan, X., Wu, Y., Dong, X., Wei, Y., Lu, M., Zhang, Y., Xu, H., and Liang, X. (2021). Product1m: Towards weakly supervised instance-level product retrieval via cross-modal pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11782–11791.

Zhang, L., Zhang, X., and Pan, J. (2022). Hierarchical cross-modality semantic correlation learning model for multimodal summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11676–11684.

Zheng, R., Chen, J., Ma, M., and Huang, L. (2021). Fused acoustic and text encoding for multimodal bilingual pretraining and speech translation. In *International Conference on Machine Learning*, pages 12736–12746. PMLR.