CIMAT

Centro de Investigación en Matemáticas, A.C.

# Biprojection Multimodal Transformer for Supervised Multimodal Data Classification

**T E S I S**

Que para obtener el grado de

**Maestro en Ciencias con Especialidad en Computación y Matemáticas Industriales**

**Presenta**

Diego Aarón Moreno Galván

**Director de Tesis:**

Dr. Adrián Pastor López Monroy
Centro de Investigación en Matemáticas (CIMAT)

**Codirector de Tesis:**

Dr. Luis Carlos González Gurrola
Universidad Autónoma de Chihuahua (UACH)

**Autorización de la versión final**

Guanajuato, Guanajuato, Junio 2022

# Acknowledgements

During the process of writing this thesis, I have received a lot of support and help.

First, I would like to thank my supervisors, Adrián Pastor López Monroy and Luis Carlos González Gurrola, whose invaluable expertise in the formulation of research questions and methods. Your insightful comments have motivated me to sharpen my thinking and take my work to the next level.

I would also like to thank my tutors, Dr. Oscar Dalmau and Dr. Pastor López, for their invaluable advice throughout my studies. You gave me the tools I needed to choose the right direction and complete my goals.

I would like to acknowledge CIMAT (Centro de Investigación en Matemáticas) and CONACyT for the support and for all of the opportunities I was given to perform my research.

I would also like to thank my parents for their wise advice. You are always there for me. Finally, I would not have been able to complete this thesis without the support of my partner and friends who provided stimulating discussions and enjoyable distractions to rest my mind outside of my research.

# Abstract

Analyzing, manipulating, and comprehending data from multiple sources (e.g., websites, software applications, files, or databases) has become increasingly important. For instance, many digital platforms, such as Netflix or YouTube, are interested in recommending appropriate and relevant digital material for us based on multimodal information, such as language turned into text, images, and audio. Current research centers on developing efficient strategies to automatically analyze and comprehend this type of multimodal content for classification. Furthermore, multimodal data is also used to detect emotions and sentiments in behavioral investigations. The movie genre categorization task is an exciting research case among the numerous problems in the modality domain. We take it in this work as a study case with two distinct datasets. Moreover, we also take the emotion recognition task as a study case and conduct several experiments with another two datasets.

A recently employed technique for multimodal categorization is the Multimodal Transformer (MulT) model, which is based on taking relevant information using attention matrices of each modality. Our research primarily addresses the automatic and efficient modality combining issue within the MulT model. The MulT has an excellent performance on many supervised multimodal datasets, but the information from various modalities is fused with a heavy transformer and is not intuitive. Furthermore, we focus on leveraging the GMU module within the architecture to efficiently and dynamically weigh modalities at the instance level and to comprehend and visualize the use of modalities. Moreover, a common challenge in deep learning is when the model fails to learn due to vanishing gradients; to overcome this issue, we focus on strategically placing residual connections in the architecture. We propose a novel architecture to compete with current state-of-the-art (SOTA) models in movie genre classification, outperforming them by 2% on Moviescope and 1% on MM-IMDB datasets. In the emotion recognition task, we get competitive performance with SOTA models on the IEMOCAP dataset, and we improve by 1% on the CMU-MOSEI[1] dataset.

**Keys words:** multimodal classification, transformers, movie genre classification, emotion detection, multimodal transformer, BERT, GMU, multimodal fusion, deep learning.

[1] Dai, Cahyawijaya, Liu, and Fung (2021)

# Contents

# List of Figures

# Chapter 1

# Introduction

Multimodal classification has become increasingly relevant for analyzing data from many sources, Baltrušaitis, Ahuja, and Morency (2017), and Xu, Zhu, and Clifton (2022). For instance, many digital platforms are interested in recommending relevant and appropriate content for us. Besides, multimodal data are also present in detecting emotions and sentiments for behavior studies, for example, the Social Anxiety Disorder Nikolić et al. (2020), the engagement while learning Charland, Léger, Sénécal, and Courtemanche (2015), or the stress detection Yao, Papakostas, Burzo, Abouelenien, and Mihalcea (2021). The word "multimodal" refers to using the information from various representations or transmission channels to perform classification, e.g., in a video, we can obtain the language which can be transformed into text, the sequence of images, and the audio. The effective use of these modalities is a current area of research called multimodal classification task, Sleeman-IV, Kapoor, and Ghosh (2021).

In our case, we focus on the movie genre classification task and perform different experiments on the multimodal emotion recognition task. We use the movie genre categorization task because it is a multimodal problem (which may involve video, audio, and text). Also, there is a recent open-access multimodal movie genre classification dataset called Moviescope. Moreover, this task is essential in our lives to have appropriate content when using social media or to see a movie. On the other hand, we use the emotion recognition task because it also is a multimodal problem since we

can analyze gestures, intonation, and the words that a person says to detect their sentiment. This task is commonly used to compare models that work with multimodal data, and we introduce our proposed model into this benchmark. Traditional text-based approaches to sentiment analysis collect large amounts of text data, from which various algorithms are used to extract sentiment information. However, multimodal sentiment analysis offers a way to perform opinion analysis based on a combination of video, audio, and text that goes far beyond traditional text-based sentiment analysis in understanding human behavior, P. Chauhan, Sharma, and Sikka (2021). Furthermore, this task finds various applications like movie reviews on YouTube, product opinions, or health care applications, including stress and depression analysis, Chandrasekaran, Nguyen, and Hemanth (2021).

In this context of multimodal data analysis, various methods were designed to handle and combine data from many sources. A typical recent model mainly used in supervised classification is a transformer-based neural network called Multimodal Transformer (MulT). Some recent works for multimodal classification use the MulT model as their base form. The problem with this model is that it fusions the information from a distinct source with a Transformer which is a heavy architecture, and it is not clear what modalities are taken into account for the final decision. Our research focuses primarily on solving modalities combination within the MulT model, adding a dynamic combination module and a reorder in the crossmodal transformers called **biprojections** in this work. We proposed a novel architecture to tackle current state-of-the-art models in the movie genre classification. We extended our experiments to the emotion recognition task to introduce our proposed model into this standard used multimodal benchmark.

## 1.1 Problems

1) The modalities interaction inside the MulT model, Figure 1.1, is given by a complete search of each modality pair combination using the Crossmodal Transformer. We say in this work that a modalities combination is a projection from

one modality space to another. For example, in the Yellow Crossmodal Transformer blocks of Figure 1.1, there is a projection from Video (V) to the Language (L) (V—›L). In this context, a simple projection from the space of modality $B$ to the space of $A$ does not rescue any information of other different modality $C$ because $B$ does not have access to $C$. In the example, the projection from Video to Text (language) does not have any information on the Audio modality. We hypothesize that it is a problem in the MulT that we can solve by taking a **biprojection**. A Biprojection first involves projecting modality $C$ to the space of $B$, then $B$ would have relevant information of $C$ and $B$. Therefore, we project the modified modality $B$ to the space of $A$ to have an enriched modality representation. In the example, we first take a projection from Audio to Video, then this Video features projected to the Text modality.

2) Secondly, the MulT model combines two representations of one modality by a transformer. For example, it takes two simple projections $B$ to $A$ and $C$ to $A$ to get an enriched representation of modality $A$. In Figure 1.1, we can see what we mentioned after the Crossmodal Transformers concatenation. The transformer to combine these vectors takes a lot of memory space and is hard to interpret. We believe that we can solve this problem by substituting this transformer for a proposed dynamic fusion module based on the GMU architecture, Arevalo et al. (2017).

3) Finally, the inefficient learning caused by the vanishing gradient is a prevalent problem in deep learning neural networks. Note that the MulT model has just residual connections inside the Crossmodal Transformer but does not have any outside these modules to address this problem. We hypothesize that adding strategical residual connections between levels of these projections will help the model to learn.

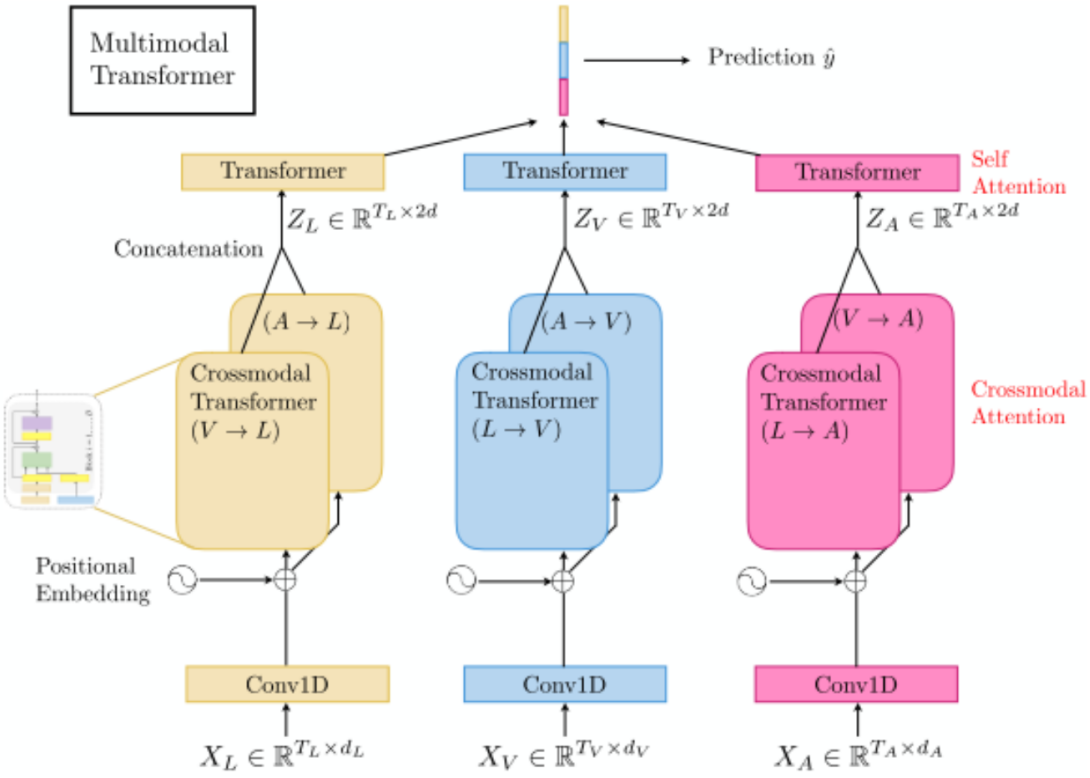



**Figure 1.1:** Multimodal Transformer architecture. Yellow blocks are the related attention to the text, blue blocks to the video, and pink to the audio. Crossmodal Transformer blocks are trained to integrate modalities, Tsai et al. (2019).

## 1.2 Objectives

Our **main** objective is to improve the representation of modalities combination within the Multimodal Transformer architecture to classify with better precision and achieve similar or superior performance to the state-of-the-art models in the supervised multimodal classification task.

### 1.2.1 Specific Objectives

1) Our **first specific** objective is to find a better representation of the modalities combination using the crossmodal transformers proposed in Tsai et al. (2019).

2) The **second specific** objective is to substitute the information fusion method (transformer) with a better, lower-cost method without decreasing the performance in classification.

3) The **third specific** objective is to improve the learning by introducing strategically residual connections within the architecture.

4) Finally, our **fourth specific** objective is to analyze and understand the information flow inside the architecture. We aim to have an interpretable model which shows each modality combination's relevance.

## 1.3 Contribution

The contributions of this thesis are fivefold. We propose a novel multimodal architecture that considers three or more modalities to competitively classify movie genres and emotions. Our first contribution is that this architecture considers a reorder of crossmodal transformers called biprojection to enrich each modality with the other sequences. It is a crucial step because this module reordering allows a better flow of information, introduces residual connections, and finds better patterns across the modalities—our second contribution related to the combination of multimodal information. We introduce the Fusion Gated Multimodal Unit (FGMU) based on the

GMU, Arevalo et al. (2017) to fuse information from the same modality. This module modifies the original GMU, just adding a residual product inside the module. It is an essential piece because we show that if we use the original GMU, it affects the classification performance. Our third contribution is introducing two types of residual connections to address the efficient learning problem in deep neural networks. One type is an adding connection between the biprojections, and the second is a fusion of projections followed by an adding. We showed that it is the best configuration for better classification performance. The fourth contribution is related to a novel taxonomy inspired by the hybrid method. We called this taxonomy a parallel method where one part learns to classify heavily, and the other learns lightly. To reduce the over-fitting, we add to our model a parallel and simple modalities fusion to find patterns in sequences at a high level. Then, we fusion these parallel architectures (light and heavy) with a GMU to dynamically fuse and predict. Finally, our fifth contribution is that we outperform the proposed model's current state-of-the-art results in Moviescope and MM-IMDb for movie genre classification. It has competitive performance in the emotion recognition task.

## 1.4 Thesis Structure

This thesis is composed of six following essential parts. In Chapter 2, we present the main aspects to understand what these works are attacking and the tools we are using. Chapter 3 briefly describes the previous work that has influenced this research regarding datasets, models, structures, and ideas. In Chapter 4, we can find the complete details about our proposed architecture for multimodal classification. Chapter 5 contains a brief description of the considered datasets, current SOTA models on each corpus, their metrics, and statistics. Chapter 6 has the experiments we did on Moviescope, an ablation study to determine the best configuration, and a study of modalities' relevance and the results in other datasets. Finally, Chapter 7 describes our research conclusions.

# Chapter 2

# Theoretical Framework

We review relevant machine learning and deep learning techniques inside the NLP area. This chapter presents the theoretical framework on which we based this work which we describe in Chapter 4. Section 2.1 includes traditional NLP machine learning approaches, such as Bag of Words with its term weighting schemes and some word embeddings used in this work when we perform experiments in the IMDb dataset. Section 2.2 describes the most common deep learning architectures, such as transformers, BERT, GMU, and Multimodal Transformers. We use these previous architectures as a part of our novel proposed architecture. Finally, Section describes the evaluation metrics we use in our experiments.

## 2.1 Supervised Text Classification

Text classification is one of the most common problems in natural language processing (NLP) and machine learning. It can be used for sentiment analysis, topic classification, movie genre detection, and more. In general, supervised text classification consists in assigning a category or label to a piece of text which is typically done by training a classifier on a dataset of labeled text. Then, the classifier can predict the category for new, unlabeled text. Various algorithms are commonly used for text classification, including support vector machines, naive Bayes, and decision trees.

Though text classification is one of the fundamental tasks in NLP, recent work is

not only using text for its applications. Instead, the current research orientation is on how to benefit from combining text with extra information that could provide audio and video frames. This blend of information is called Multimodal Classification and will be discussed in the following sections.

### 2.1.1 Bag of Words

The Bag-of-Words (BoW) representation is a language representation commonly used in natural language processing (NLP) and information retrieval (IR). The BoW represents a text (such as a sentence or a document) as a set of its words' frequencies, disregarding grammar and word order. This model is typical for document classification methods, where each word's frequency is used as a feature to train a classifier.

#### BoW Weighting

In practice, a Bag-of-Words model is mainly a tool for feature generation. After transforming the text into a "bag of words," we can calculate various measures to describe the text. The most common feature calculated from the Bag-of-words model is term frequency, i.e., the number of times a term appears in each document. For example, if we take two documents: "*Hello, my name is Diego, what is your name?*" and "*Hello, my name is Aron.*" Then, we can construct the following two vectors taking into account the term frequencies of all the distinct words ordered by the tokens (nine columns for each token: [*"name", "is", "Hello", "my", "Diego", "what", "your", "?", "Aron"*]):

$$\begin{aligned} BoW_1 &:= [2, 2, 1, 1, 1, 1, 1, 1, 0], \\ BoW_2 &:= [1, 1, 1, 1, 0, 0, 0, 0, 1], \end{aligned}$$

where each entry is the count of the corresponding token (each row represents a token).

The first entry corresponds to the word *"name"* which has a value equal to 2 for

$BoW_1$ because *"name"* appears in the first document twice. On the other hand, in the second document ($BoW_2$), the word *"name"* appears just once. Note that this vector representation does not preserve order words in the original sentences.

However, term frequency is not always the best representation of the text. Notably, common words like "is," "a," or "the" are the terms with the highest frequency in the text. Thus, having a high raw count does not necessarily mean that the corresponding word is essential.

One way to solve this problem is to weigh a term by its inverse document frequency (TF–IDF).In this case, *"name"* and *"is"* are two words that appear in both, so these tokens are penalized by the IDF score. For example, for the BoWs above, we have IDF scores as:

$$IDF := \left[\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 1, 1, 1, 1, 1\right],$$

then, the TF-IDF scores are described as:

$$\begin{aligned} BoW\,TF\text{-}IDF_1 &:= \left[1, 1, \frac{1}{2}, \frac{1}{2}, 1, 1, 1, 1, 0\right], \\ BoW\,TF\text{-}IDF_2 &:= \left[\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, 0, 0, 0, 1\right]. \end{aligned}$$

## 2.2 Neural Networks in NLP

In order to avoid becoming dependent on handed-crafted rules, machine learning text classification learns to make a classification based on past observations. Machine learning algorithms can use labeled examples as training data to learn different patterns between pieces of text (or other modalities) and output, i.e., labels. A "label" is a predetermined category into which any given sequence could fall.

The first step towards training an NLP machine learning classifier is to perform a feature extraction: a method is used to transform each sequence into a vector with numerical representation. For example, a vector represents the word's frequency in a predefined dictionary of words with the Bag of words approach. If we have defined

our dictionary to have the following words {"Hello," "my," "name," "is," "Diego" }, and we want to vectorize the text "Hello Diego," we have the following vector representation of that text: (1, 0, 0, 0, 1).

The machine learning algorithm is then fed training data consisting of feature sets and tags to generate a classification model. The machine learning model can begin to make accurate predictions after it has been trained with enough training samples. The same feature extractor is used to convert unseen text into feature sets that can be fed into the classification model to generate predictions. Text classification using machine learning is typically much more accurate than hand-crafted rule systems, especially for complex NLP classification tasks.

### 2.2.1 Word Embedding

The word representations we presented are either a vocabulary index or a weight premised on occurrences in a document or a corpus. However, none of these representations recall the word's semantic meaning. The embedding notion is based on representing words with continuous vectors in a space with geometry such that words with similar meanings are close even if they do not have the same root.

There were several approaches to generating word embeddings, but word2Vec Mikolov, Corrado, Chen, and Dean (2013) caught the attention of the NLP community. There is substantial evidence that the embedding space produced by this representation meets the requirement of similar words having close embeddings. Moreover, the word2vec representations keep their semantic properties under some algebraic operations. For instance, the vector

$$\text{vec}(queen) - \text{vec}(woman) + \text{vec}(man)$$

will generate a vector close to the *vec(king)*. There was no specific instruction to promote this behavior among the resulting embedding. Nonetheless, it demonstrates how rich the geometry in the embeddings' space is.

Word2Vec is a skip-gram model augmentation, Mikolov et al. (2013). The skip-

gram model aims to predict nearby words in a phrase based on embedding a single word.

The Word2Vec model modifies the skip-gram loss function for Negative Sampling Estimation, which is defined as

$$P(w)[log(-v_0 w_i^T v w_I)] \sim \log(v_0 w_O^T v w_I) + X_k \sum_{i=1} w_i, \tag{2.1}$$

where $v_0 w_O$ is the output projection of a positive sample, $v w_I$ is the input projection of the fixed word, and $v_0 w_i$ is the output projection of the negative samples. Negative Sampling Estimation compares the fixed word to a positively related sample and $k$ noisy vectors.

Other successful word embeddings, such as Fast Text Bojanowski, Grave, Joulin, and Mikolov (2017), were proposed by Facebook. Moreover, GloVe was advocated by Stanford NLP in Pennington, Socher, and Manning (2014). While these representations already include semantic information, we would like to develop a specific word representation for each job depending on the surrounding environment; this can be accomplished using Neural Networks such as Recurrent Neural Networks and Transformers.

### 2.2.2 Transformer

The RNN Seq2Seq network models (LSTM and GRU in particular) and attention mechanisms for addressing transduction problems, such as machine translation and language modeling, inspired the Transformer design, Vaswani et al. (2017). Given that input and output processing must be sequential, there are some limitations on the parallelization potential of RNN's Seq2Seq models.

On the other hand, attention mechanisms have enhanced performance in various tasks involving the modeling of transduction and sequences, enabling the network to model dependencies between items in the input or output sequence regardless of their proximity. However, such attention methods were typically utilized in recurrent neural network models.

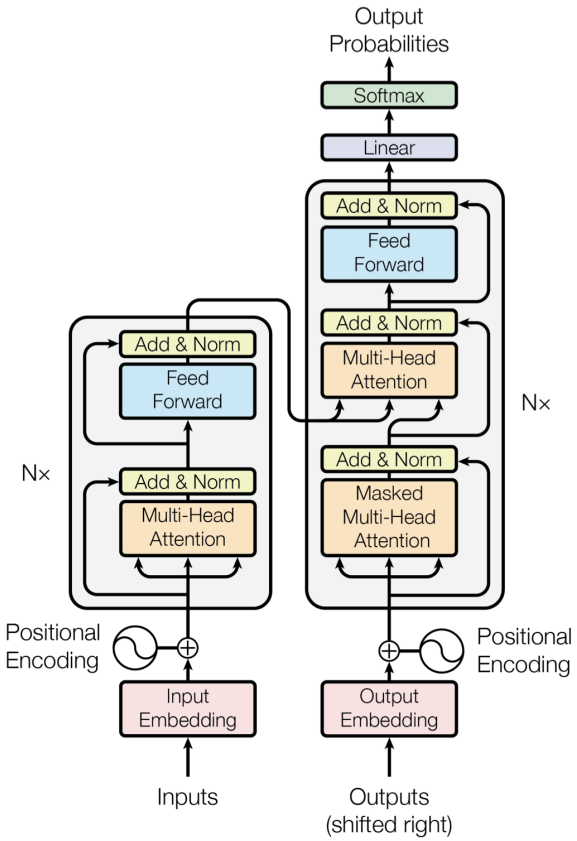



**Figure 2.1:** The Transformer architecture, Vaswani et al. (2017). On the left side, we see the encoder, which learns how to represent the input. On the right side is the decoder, which learns how to use the encoder information to produce an output.

By not processing the input sequentially, the Transformer architecture avoids recurrence. It relies on attention techniques to establish global dependencies between input and output. It enables high parallelization, and recent model variants attain SOTA outcomes across various activities.

An encoder-decoder architecture is used in the Transformer model. Figure 2.1 depicts both the encoder and the decoder. The architecture includes a stack of self-attention and fully-connected units for both the encoder and decoder. The encoder converts the input sequence $(x_1, ..., x_n)$ into a series of continuous representations $z = (z_1, ..., z_n)$. The decoder outputs an output sequence $(y_1, ..., y_n)$ one element at a time, given $z$.

The encoder comprises a stack of $N = 6$ identical layers (left side gray box in Figure 2.1). Each layer is divided into two sublayers. The first sublayer consists

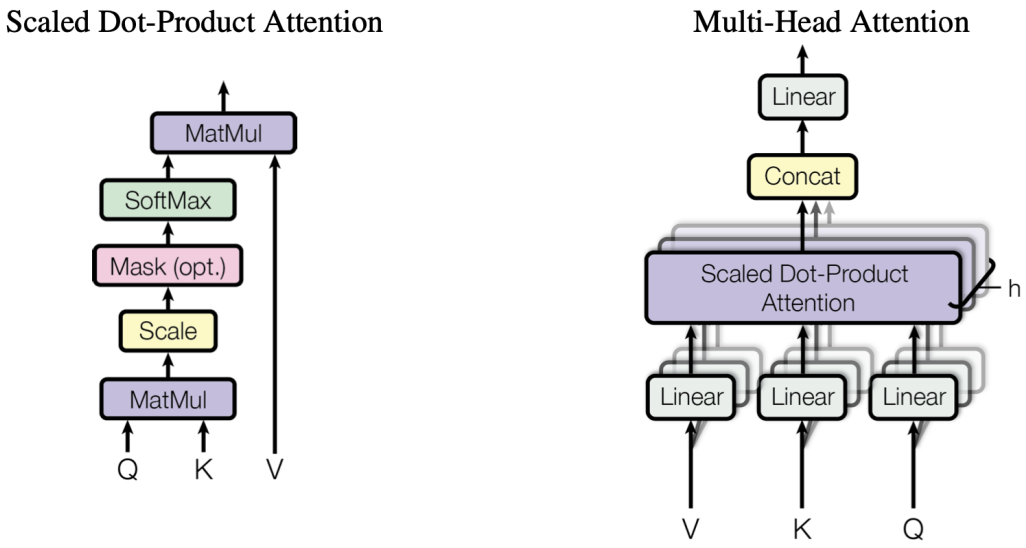



**Figure 2.2:** Scaled Dot-Product and Transformer Attention mechanism which takes queries (Q), keys (K), and values (V) to perform the attention, Vaswani et al. (2017).

of a multi-head self-attention mechanism (shown in orange in the same figure). On the other hand, the second sublayer is made up of a fully-connected feed-forward layer (shown in blue in 2.1). Around these two sublayers, a residual connection is also employed, followed by layer normalization. LayerNorm($x$+ SubLayer($x$)), where SubLayer($x$) is the function implemented by the same sublayer, is the output of each sublayer. The model's sublayers (including the embedding layers) all yield 512-dimensional output.

The decoder is also comprised of a stack of $N = 6$ similar layers (2.1, right side huge gray box). The decoder introduces a third sublayer, which performs multi-head attention over the output of the last encoder layer and the two sublayers in each encoder layer. Following layer normalization, the decoder implements residual connections surrounding each sublayer. The decoder's self-attention layer is adjusted to prevent positions from attending to following positions.

Masking refers to the change of the self-attention layer. The transformer combines this masking with a one-position shift to the right of the output embeddings so that the prediction at position $i$ can only rely on the known outputs at places less than $i$.

A query and a set of key-value pairs are mapped to output by an attention function, where queries, keys, and values are all vectors. The result is a weighted sum of the

data, with each weight supplied by a compatibility function between the query and the relevant key. Figure 2.2 gives a visual description of the Attention process.

Dot-Product Scaled Attention Vaswani et al. (2017) is the name given to the suggested attention mechanism in the Transformers paper and is represented in (Figure 2.2). The input comprises queries and keys of dimension $d_k$ and values of dimension $d_v$. The dot product of a specific query and all keys are computed by dividing each by $D_K$. Then, applying a softmax function to produce the weights on the values.

This operation is conducted in matrix form so the model can simultaneously process a series of questions packed in a matrix Q. The keys and values are encoded in matrix form in matrices K and V. The following equation is how the output matrix is computed:

$$\text{Attention}(Q, K, V) = \text{Softmax}((\text{multihead-attention})/\sqrt{d}) \tag{2.2}$$

The model may attend to information from distinct representation subspaces at different positions using Multi-Head Attention Vaswani et al. (2017). Instead of conducting only one attention function with the dimension model's queries, keys, and values, Vaswani et al. (2017) discovered that it was more effective to project them $h$ times using learned linear projections to dimensions $d_k$, $D_K$, and $D_V$, respectively. In this projected version of each query, key, and value, each head calculates the attention function in parallel, resulting in dimension $d_v$ output values. Finally, as illustrated in Figure 2.2, these outputs are concatenated and projected again, providing the final values.

$$\text{MultiHead}(Q, K, V) = \text{concat}(\text{head}_1, ..., \text{head}_h)W_O \tag{2.3}$$

where $head_i = \text{Attention}(Q_{W_i}, K_{W_i}, V_{W_i}) = \text{Softmax}((QK(QK)^T)/\sqrt{d}))V$, and projections are parameter matrices $W_{iQ} \in R^{d_{model} \times d_k}$, $W_{iK} \in R^{d_{model} \times d_k}$, $W_{iV} \in R^{d_{model} \times d_v}$, and $W_O \in R^{h \times d_v \times d_{model}}$.

In the first Transformer proposal, each layer used $h = 8$ attention heads simultaneously. The dimensions used for each of these heads were $d_k = d_v = d_{model}/h = 64$.

It provides a computational cost equivalent to a full-dimension model with only one attention head.

#### Positional Encoding

The Transformer model takes no account of convolution or recurrence. As a result, the model must include the order and placement of the tokens in the sequence information. The initial plan was to add "positional encodings" to the input embeddings at the encoder and decoder's bottom. Because these positional encodings share the same dimension model as token embeddings, the addition operation is possible. The positional encodings are computed using sine and cosine functions of varying frequency. The main aim is to generate or learn a signal for each sequence element that allows the model to extract and utilize the sequence's order.

$$\text{PE}(pos, 2i) = sin(pos/10000^{2i/d_{model}}) \tag{2.4}$$

$$\text{PE}(pos, 2i+1) = cos(pos/10000^{2i/d_{model}}), \tag{2.5}$$

where $i$ is the dimension in the embedding vector and $pos$ is the position of the sequence.

### 2.2.3 BERT

Following the transformer architecture, numerous models have been proposed to improve its performance in specific tasks. For example, the GPT Radford, Narasimhan, Salimans, and Sutskever (2018) improves the performance but relies solely on decoder layers, omitting the encoder stack.

The name means Bidirectional Encoder Representations from Transformers. Contrary to previous works, the proposal BERT Devlin et al. (2019) has been built solely on encoder stacks without decoder layers. It is intended to pre-train deep bidirectional representations of text by simultaneously conditioning on both left and correct context from unlabeled in all layers.

A prediction head can be used to fine-tune the resulting pre-trained BERT model

(usually a tiny MLP with non-linear activation). In conjunction with the unsupervised pre-training and fine-tuning approach, the BERT architecture enabled the model to obtain SOTA outcomes in various NLP tasks, such as question answering and language inference. In the following sections, we will go deeper into BERT's architecture.

### Model Architecture

Based on the original implementation described in Section 2.2.2, BERT's model architecture is a multi-layer bidirectional Transformer encoder. The authors suggest two models, BERTBASE and BERTLARGE, that only differ in size. We shall refer to the BERTBASE model as BERT for the duration of this work because it is the most commonly utilized. The BERT model has the following attributes: L, the number of encoder layers; H, the hidden size; and A, the number of self-attention heads: L = 12, H = 768, A = 12, and a total number of parameters = 110M. Figure 2.3 depicts the model's general design.

BERT employs WordPiece for text representation (embeddings) Wu, Schuster, Chen, Le, and Norouzi (2016). When two sentences are packed together in the input, the model inserts a unique separation token called [SEP] between them. Every sequence's initial token will begin with a unique categorization token termed [CLS], the final hidden state of which will be used at the model's output to make predictions about the entire sequence.

Aside from these two distinct tokens, the authors add a learned embedding to each token that indicates whether it belongs to the first sentence (sentence A) or the second sentence (sentence B) (sentence B).

### Pretraining and Fine-Tuning

Masked-LM and Next Sentence Prediction are two unsupervised tasks used to train BERT. In the former, a random percentage of the input tokens are masked, and the model's goal is to forecast those masked tokens. In the latter, while selecting sentences A and B for each pre-training example, B is substituted 50% of the time by a random sentence from the corpus, and the other 50%, B is the actual next sentence

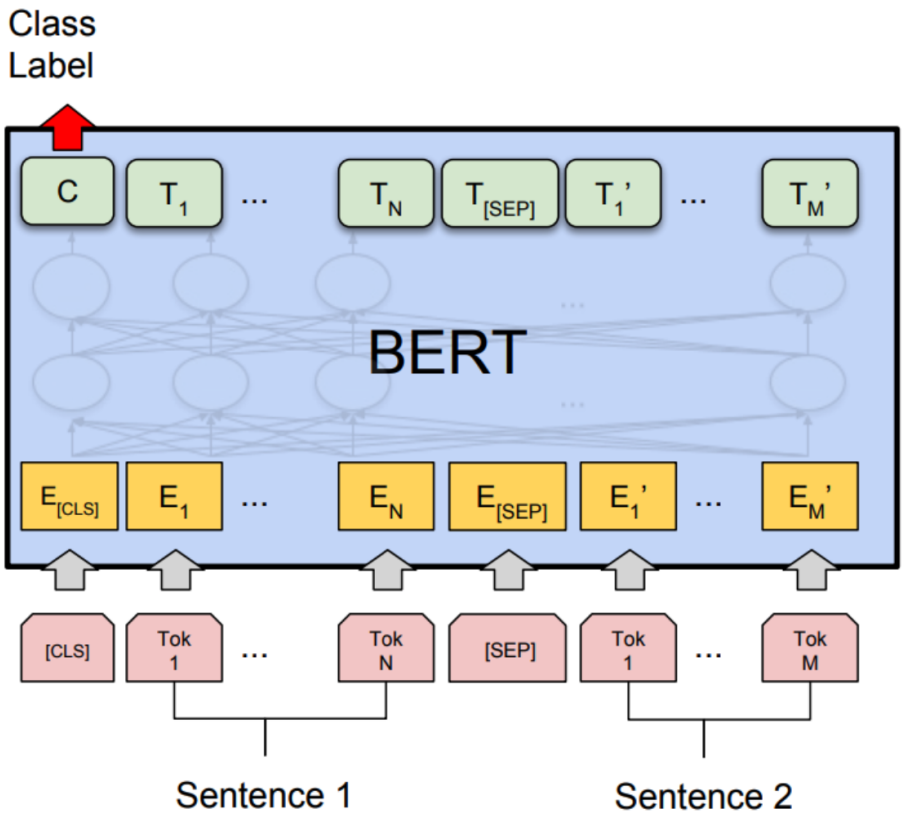



**Figure 2.3:** The architecture of the BERT model. At the bottom, input word-piece tokens are shown in pink. Tokens with position and sentence information are in yellow in the final input representation, Devlin et al. (2019).

that follows A.

For this job, the prediction is made using vector C (the final hidden vector linked with the CLS token) (IsNext, NotNext). For instance, the accurate prediction should be NotNext if sentence A is "The man went to the store," and sentence B is "Penguins are flightless," as there is no logical link between the two sentences.

### BERT Fine-tuning

For each task, the model is trained plus an additional prediction head (i.e., a Feed-Forward Neural Network) on top of the BERT architecture, as illustrated in Figure 2.4. BERT receives the inputs and gives outputs formatted as indicated. Then all the parameters are fine-tuned from start to finish. The only change required to fine-tune BERT is to encode the input as previously described, i.e., insert the [CLS] token at

the beginning of the sequence. If only one sentence is necessary, append a [SEP] token at the end of the sequence; if two sentences are required, append the second sentence after the [SEP] token and then add another [SEP] token at the end of the sequence. The model will be able to perform bidirectional cross-attention between the two sentences as a result of this.

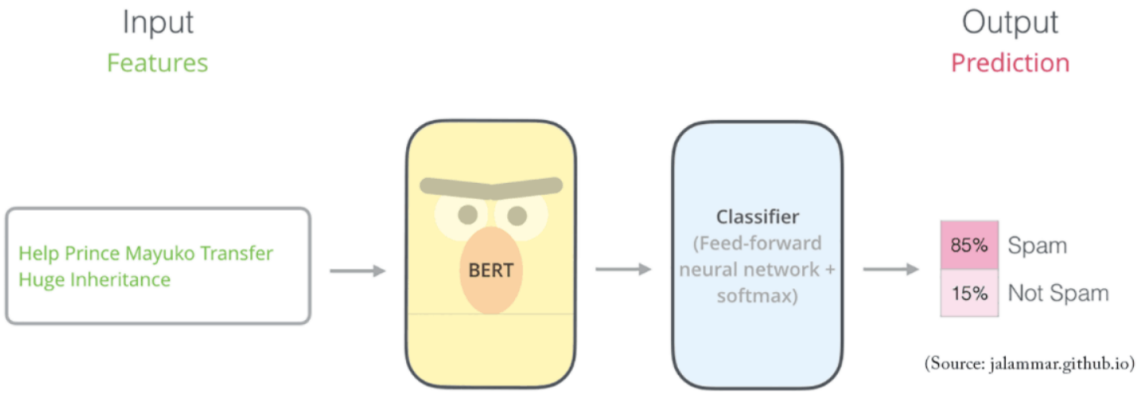



**Figure 2.4:** Fine-tuning a pre-trained BERT model by adding a Feed Forward Neural Network on top to generate the output, Alammar (2018).

The experiments began with the GLUE Benchmark, which shows a performance boost of 4.5 percent above the prior state-of-the-art, demonstrating that pre-training deep bidirectional models can improve performance in a wide range of NLP tasks.

### 2.2.4 Multimodal Transformer

Combining information from several modalities with a modified self-attention mechanism is the main contribution of the proposed Multimodal Transformer (MulT) Tsai et al. (2019). This mechanism's input is a pair of modalities (e.g., text and video or audio and video). It enables the model to accommodate unaligned modalities, that is, modalities with no token connection between two supplied modalities.

MulT is an architecture for representing unaligned multimodal language sequences. MulT merges pairs of multimodal time series using blocks of crossmodal transformers, each of which aims to reinforce a target modality with low-level information from another source modality by learning attention across the two modalities' features.

The complete model (MulT) that considers three separate modalities (L, V, A) is developed using six Crossmodal Transformers, one for each interaction between two modalities. MulT can simulate all pairs of modalities by linking numerous cross-modal transformers. Three modalities are used in the MulT paradigm described here: language (L), video (V), and audio (A).

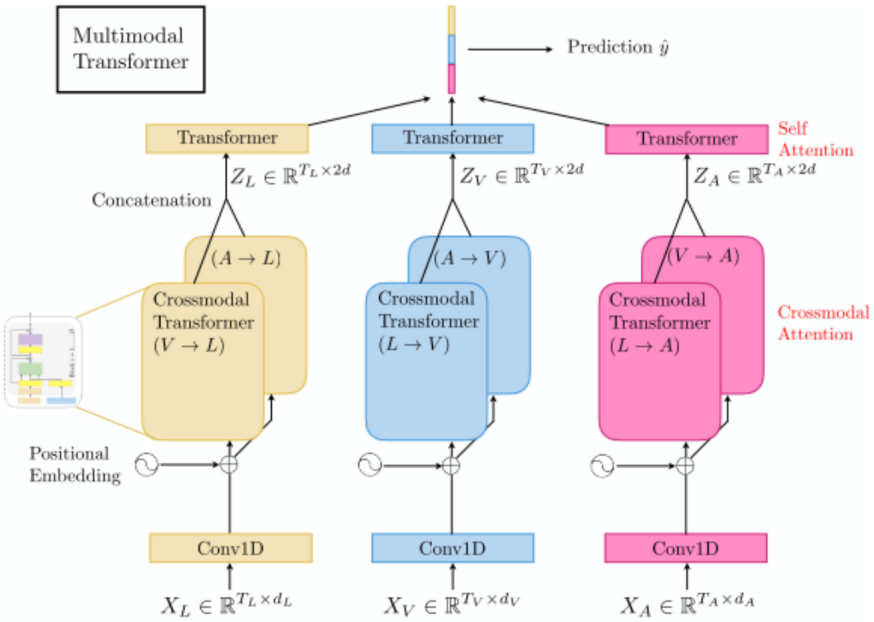



**Figure 2.5:** A summary of the MulT architecture. $X_L$ represents text input, $X_V$ represents vision input, and $X_A$ represents audio input. Crossmodal Transformer blocks are trained to integrate modalities, Tsai et al. (2019).

MulT, unlike the models discussed in earlier subsections, does not use a pre-training procedure. It allows us to arbitrarily set the sequence length and number of modalities required for each modality, which is helpful for movie genre categorization, where we have information on the text, video, audio, and metadata. As a result, we used this model as the foundation for our experiments.

### 2.2.5 Gated Multimodal Units

Concatenating the representations of each modality into a lengthy vector or performing a simple operation like adding, taking the product, or the maximum value in each component is a convenient solution to the challenge of fusing information from

distinct modalities. However, this is inconvenient because we presume that the modalities are equally relevant, which is not always the case. Depending on the situation and goal, one or a subset of modalities may be more informative in providing an accurate forecast.

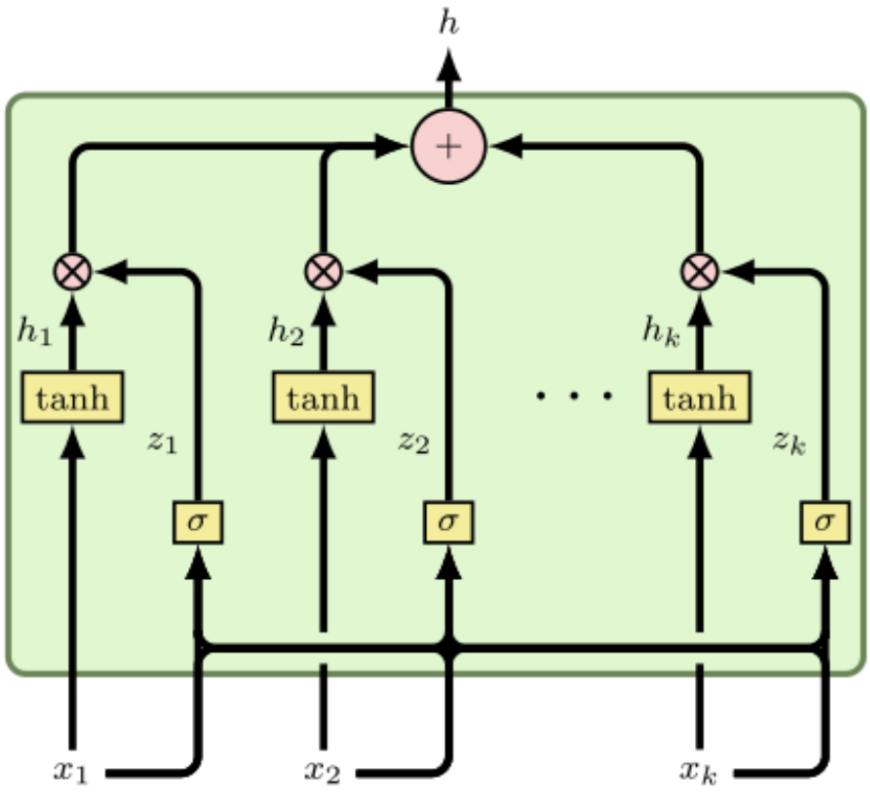



**Figure 2.6:** Overview of GMU module. Where $x_i$ represents the ith input modality. The final fused representation of all modalities is represented by h at the top, Arevalo et al. (2017).

This paper offers a novel hidden unit called Gated Multimodal Unit (GMU) Arevalo et al. (2017), inspired by the control flow mechanism in gated recurrent units. The primary components are the gates that allow the model to manage the flow of diverse input into the subsequent phases.

Before combining the input modalities, the GMU learns to weigh them based on their significance. GMU was designed to be used as a neural network layer to discover an intermediate representation by mixing data from several modalities.

Given $x_i \in R^{d_i}$, the feature vector associated with modality $i$, a weight $z_i$ deter-

mines the modality's contribution to the total output of the GMU layer. Obtaining the final fused representation as a weighted sum $h = n_i = 1, z_i.hi$, where . denotes component-wise vector multiplication. Figure 2.6 depicts a high-level overview of the GMU module.

These gates will enable the model to determine how each modality impacts the unit's output. The interpretability of such a GMU unit is one of its advantages. Following model training, the weights $z_i$ can be visualized to understand better which modalities had more influence or were more helpful in making the prediction.

In Chapter 4, we give many options for using an extension of the GMU module to increase the capacity to weigh the relevance of modalities and make dynamic choices.

## 2.3 Evaluation Metrics

In order to evaluate our models, we consider many metrics to be consistent with the current SOTA models on each dataset. For multilabel classification, the Average Precision Score is the one most commonly used as the same as the F1 score on its weighted version. We also use the F1 score, Accuracy, and Weighted Accuracy in the datasets considered for our work. We took this information from "SciKit-Learn API Reference" (2022).

### Average Precision Score

The Average Precision score summarizes a precision-recall curve as the weighted mean of precisions achieved at each threshold, with the increase in recall from the previous threshold used as the weight:

$$\text{AP} = \sum_n (R_n - R_{n-1}) P_n \tag{2.6}$$

This implementation is not interpolated because $P_n$ and $R_n$ are the precision and recall at the nth threshold.

It can be used with different versions, which are:

$\mu$: Calculates metrics globally by considering each element of the label indicator matrix as a label (Micro).

m: Calculates metrics for each label and finds their unweighted mean. It does not take label imbalance into account (Macro).

W: Calculates metrics for each label and finds their average, weighted by support (the number of true instances for each label), (Weighted).

S: Calculates metrics for each instance and finds their average. (Samples).

### Accuracy Classification Score

This function computes subset accuracy in multilabel classification: the set of labels predicted for a sample must match the corresponding set of labels in $y_{true}$.

For the Weighted version, we take the formula.

$$WAcc = \frac{TP \times N/P + TN}{2N}, \tag{2.7}$$

where $P$ means total positive, $TP$ true positive, $N$ total negative, and $TN$ true negative.

### F1 Score

Compute the F1 score, also known as balanced F-score or F-measure.

The F1 score can be interpreted as a harmonic mean of precision and recall, where an F1 score reaches its best value at one and worst score at 0. The relative contribution of precision and recall to the F1 score are equal. The formula for the F1 score is:

$$F1 = 2 * (precision * recall)/(precision + recall) \tag{2.8}$$

In the multi-class and multilabel cases, it is the average F1 score of each class with weighting depending on the average parameter.

It can be used with different versions, which are:

$\mu$: Calculates metrics globally by counting the total true positives, false negatives, and false positives. (Micro).

m: Calculates metrics for each label and finds their unweighted mean. It does not take label imbalance into account. (Macro).

W: Calculates metrics for each label and finds their average weighted by support (the number of true instances for each label). It alters "macro" to account for label imbalance; it can result in an F-score that is not between precision and recall. (Weighted).

S: Calculates metrics for each instance and finds their average. It is only meaningful for multilabel classification, which differs from the accuracy score. (Samples)

# Chapter 3

# Related Work

In this chapter, we show a selection of previous publications that has the most relevant relation to our research regarding datasets, models, and components. Section 3.1 describes one of the used datasets (MM-IMDb) to compare our model and the GMU module we use in our proposal. Therefore, Section 3.2 explains a competitive architecture called MMBT that greatly performs in the MM-IMDb dataset for multimodal classification. Section 3.3 briefly describes the main dataset publication we use in our experiments. Then, Section 3.4 describes the model of Rodríguez-Bribiesca et al. (2021) in which we are mainly inspired, and it is the current state-of-the-art in the Moviescope dataset. Finally, Section 3.5 gives information about the publication where they modified and cleaned the CMU-MOSEI dataset we use for comparison.

## 3.1 MM-IMDb and GMU

The work of Arevalo et al. (2017) was one of the first publications for movie genre classification tasks. This work produced the Multimodal IMDb (MM-IMDb) dataset for this task. They also proposed a novel module for the dynamic combination of two modalities called the Gated Multimodal Unit (GMU). It is a crucial publication for our research since we perform experiments in the MM-IMDb dataset, we use the GMU in our proposed model to fuse multimodal information, and we propose a variation of this module with a residual connection inside the architecture.

Their primary motivation was that existing similar datasets for multimodal classification, at that time, were small, containing less than 10,000 movies. Additionally, they integrate extra and helpful information from the IMDb website for a better understanding, like plot, poster, and metadata.

They built the dataset using the IMDb IDs provided by the Movielens 20M dataset [1]. MM-IMDb contains ratings of 25,959 movies, plots, posters, genres, and 50 other additional metadata fields such as year, director, or aspect ratio. Each movie has one or more genres as labels, so the task is a multi-label classification using a multi-class dataset. The words' average number is 92.5, and the average number of genres is 2.48. There are 23 different genres with a notorious class imbalance. For instance, the Drama genre has 13,900 observations while Sports have around 450 observations.

The principal contributions are the multimodal MM-IMDb dataset and the Gated Multimodal Unit (GMU). For our experiments, we use the MM-IMDb for comparison with this publication and the MMBT model explained in Section 3.2. The GMU is a module compounded by gates that allow dynamically classifying based on several modalities' relevance. We explain this module in detail in Section 2.2.5 and are using this module as a part of our final architecture.

## 3.2 Multimodal BiTransformer

This work Kiela et al. (2019) uses the MM-IMDb dataset to compare their proposed model (MMBT) against GMU. They perform better on this dataset considering just the poster and text movie data.

The inspiration of the MMBT model is to take advantage of the pre-trained BERT model (on text modality) by embedding the visual features into the text token space. The model learns how to integrate and fuse embedded visual features with the text features and then performs the multimodal classification. Therefore, they perform fine-tuning like a regular BERT model.

Image features are extracted from a pre-trained ResNet-152, He, Zhang, Ren, and

[1] https://grouplens.org/datasets/movielens/20m/

Sun (2015). They extract N features because the BERT model handles sequences, so they define the N features of an image as a sequence of length N. Hence, they extract N features of dimension 2048 from a single image. Then, they perform a pooling operation instead of taking the final fully connected layer of the ResNet-152. They project the image features to dimension $D = 768$ the BERT's hidden dimension with:

$$I_n = W_n f(x, n) \in \mathbb{R}^D, W_n \in R^{2048 \times D}, \tag{3.1}$$

where $f(x, n)$ is the nth visual feature obtained from the ResNet-152 followed by a pooling operation. The final visual representation is obtained by adding each projected feature $I_n$ with the corresponding position and segment token, like in the BERT model. Fine-tuning and prediction are similar to the BERT model with the CLS token for prediction. We can see this architecture in Figure 3.1.

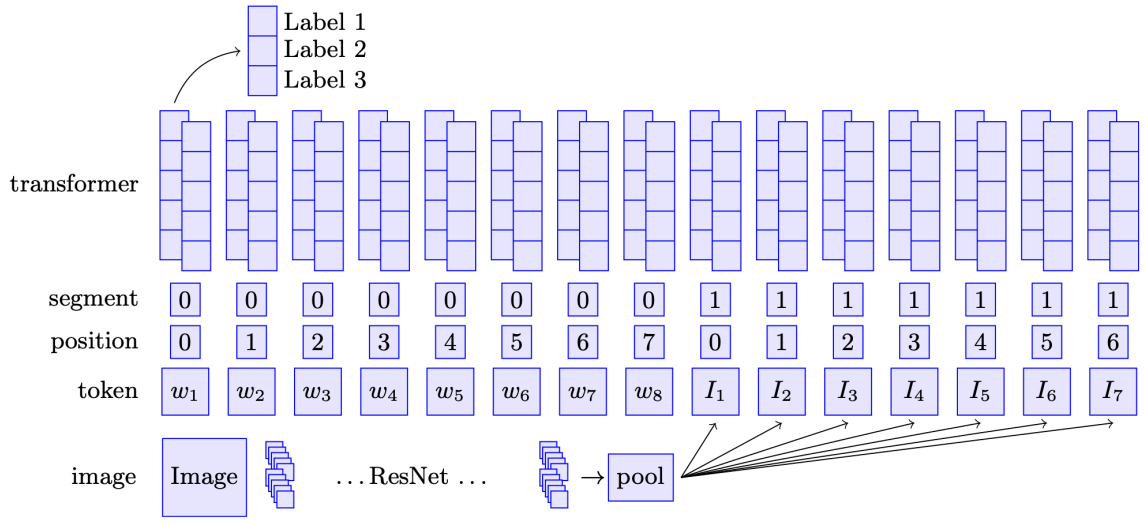



**Figure 3.1:** Multimodal Bitransformer architecture (MMBT). In this example, N = 7 features are extracted from the input image and combined with the language tokens, Kiela et al. (2019).

As we mentioned, this model was trained on three different sets, and one of them was the MM-IMDB dataset. MMBT outperforms the GMU model, and we will compare our proposed model with these classification results.

## 3.3 Moviescope: Large-scale Analysis of Movies using Multiple Modalities

This work corresponds to Cascante-Bonilla et al. (2019), and it is a novel large-scale multimodal dataset for movie genre classification tasks. The primary purpose is to study the effectiveness of visual, audio, text, and metadata-based features in predicting movie genres and budget estimation. Data includes 5,043 movie trailers with their respective synopsis, poster, and metadata information like language, movie title, or movie rating.

They use a weighted average of the GloVe embeddings for the text representation. In the case of trailer videos, they extract 200 feature vectors of size 4096 from a pre-trained VGG16 CNN. These 200 video frames are a subsample given by taking one every ten frames. For the trailer audio, log-mel scaled power spectrograms are used and processed by an LSTM. Then, they take the last LSTM hidden vector as the final audio representation. For the poster, they extract a feature vector from a pre-trained VGG16 CNN with a fully connected layer for transformation. Finally, they use a vector of dimension 312 to represent the metadata information. Furthermore, they perform a weighted sum to fuse the representations of the five different modalities through a matrix $\alpha$ of size: number of genres $\times$ number of modalities.

We use Moviescope as the primary dataset to perform our experiments and to adjust our models. We found it appropriate since it has a moderate size (number of movies) and a vast amount of information for each movie (video, audio, poster, plot, and metadata). This dataset is also used by Rodríguez-Bribiesca et al. (2021) using a different late fusion strategy: the GMU, which allows combining the different modalities dynamically according to their relevance.

## 3.4 Multimodal Transformer - GMU (MulT-GMU)

The Rodríguez-Bribiesca et al. (2021) work is our primary base model and inspiration for our research. The authors propose to classify movie genres of the Moviescope

dataset using three main components: the pre-trained BERT model to represent word sequences 2.2.3, the Multimodal Transformer 2.2.2 to fuse the text, video, and audio features, and a GMU module 2.2.5 to combine each modality and give a prediction dynamically. For audio, video, and metadata features, they use the given pre-processed features in Cascante-Bonilla et al. (2019).

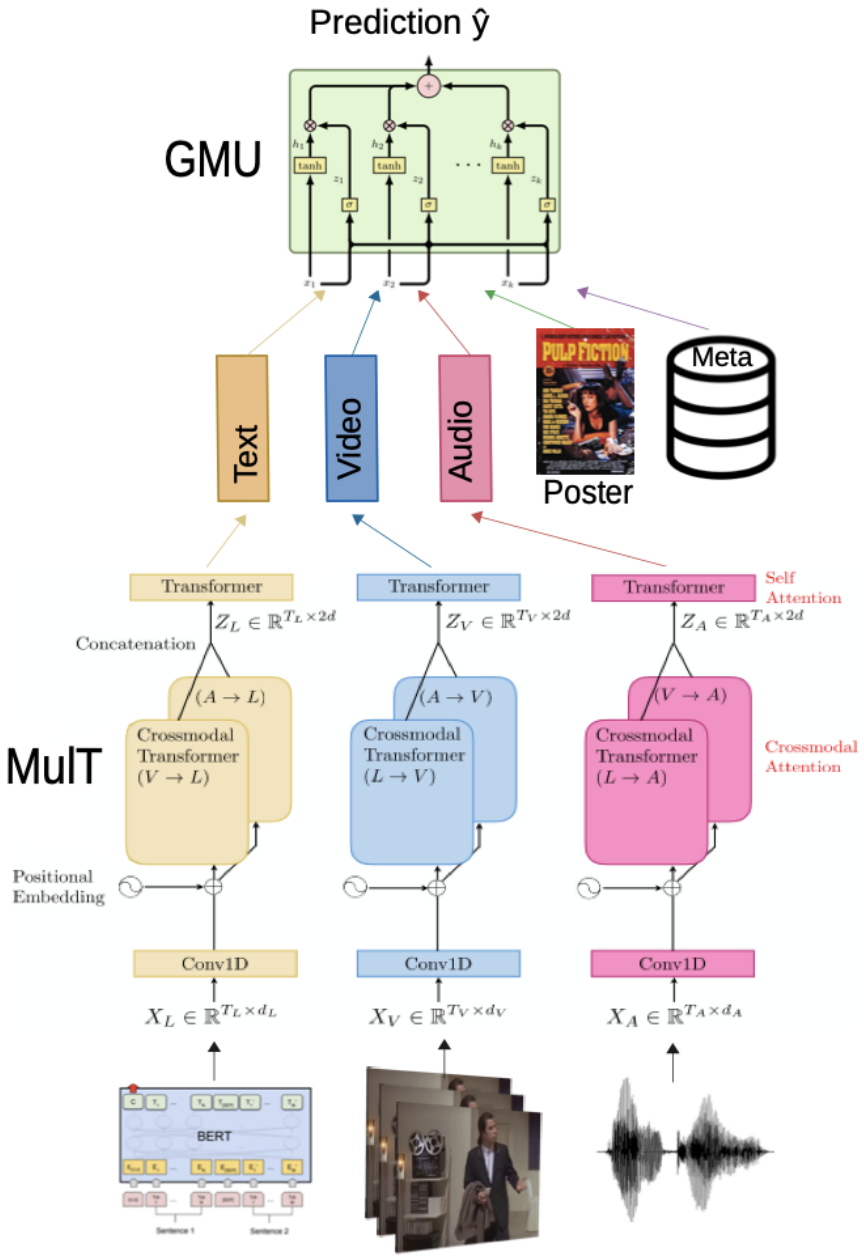



**Figure 3.2:** The MulT-GMU architecture is an extension of the MulT with a GMU module for a weighted late fusion for modalities, Rodríguez-Bribiesca et al. (2021).

The authors present their proposed MulT-GMU as an extension of the MulT model combined with the GMU module to boost its capacity. GMU weighs modalities' relevance and adds interpretability. We can see this architecture in Figure 3.2.

They perform experiments with their proposed model using different modalities. For example: text + video + audio against text + video. In conclusion, they found that the best performance is when the modalities combination is compounded by text, video, audio, and poster. Metadata adds complexity to the model, and they need to avoid essential transformer attention for better results. Hence, for our experiments, we use just the text, video, audio, and poster features for a fair comparison.

## 3.5 Multimodal End-to-End for Emotion Recognition

In this work Dai, Cahyawijaya, Liu, and Fung (2021), the author solves the existing problem on multimodal affective computing tasks, such as emotion recognition. The problem is that, generally, models use a two-phase pipeline. The first phase is to extract feature representations for each modality with hand-crafted algorithms. The second is to perform end-to-end learning with the extracted features. Note that extracted features are fixed and cannot be further fine-tuned, which does not generalize or scale well to different tasks. To solve this, the authors develop a fully end-to-end model that jointly connects the two phases and optimizes them. Hence, they restructure the CMU-MOSEI dataset (and others) to enable fully end-to-end training. Also, they introduce a sparse cross-modal attention mechanism for the feature extraction, and their fully end-to-end model is the current state-of-the-art model.

We are using this modified version of the CMU-MOSEI[2] dataset to perform our comparison experiments since this model is the current state-of-the-art.

[2]http://immortal.multicomp.cs.cmu.edu/raw_datasets/processed_data/cmu-mosei/seq_length_50/mosei_senti_data_noalign.pkl

# Chapter 4

# Proposal

This chapter describes the proposed architecture called **Biprojection Multimodal Transformer (BPMulT)** that aims to solve the problems discussed in Section 1.1. The first motivation is that the MulT model does not attend to a third modality's relevant information when doing a Crossmodal Attention between the other two modalities. Our second motivation is to substitute the transformer used for information fusion with a lighter, dynamic, and interpretable fusion module. Then, the last motivation is to introduce strategic connections between the projection within the BPMulT. With these specific objectives, this proposal aims to achieve our primary goal, which is to improve the representation of modalities combination and achieve similar or superior performance to the state-of-the-art models in the supervised multimodal classification task.

At this section's end, we observe the proposed Biprojection Multimodal Transformer (BPMulT) architecture in Figure 4.6. The proposed BPMulT model has achieved state-of-the-art movie genre classification results on the Moviescope and MM-IMDb datasets and is competitive in the emotion recognition task with the IEMOCAP and CMU-MOSEI datasets. For the experiments, we also use a lighter version of our model to compare performance when we do not care about over-fitting. We will see that the BPMulT architecture contains a simple parallel part. We call the lighter version BPMulT-no-parallel which has no parallel part. Indeed, the BPMulT-no-parallel does not have the modalities' general and straightforward information, so

it over-fits the data.

Our proposed model is compounded by five main parts described below and divided into sections. The first module (4.1) corresponds to the low-level Crossmodal Transformers, called for us: projections that are explained with our notation to have clarity, but they are precisely the same in the MulT model, Tsai et al. (2019). The second one (4.2) is a second Crossmodal Transformer, which is the proposed biprojection and is the solution to our first specific motivation. The following section (4.3) is a proposed GMU module for a dynamic and interpretable fusion of modalities called Fusion GMU (FGMU). The FGMU corresponds to the solution of our second specific motivation. The fourth module (4.4) corresponds to the parallel fusion of modality vectors we propose not to over-fit the data. The final part of this chapter (4.6) describes the strategic components of our proposed architecture used to solve vanishing gradient problems. It is the solution to our third specific motivation.

As an input, our model will consider three primary modalities: the text modality or language (L), vision modality with video frames (V), and audio (A) from a given multimodal dataset. For each modality, we have a feature sequence denoted as $X_M \in \mathbb{R}^{S_M \times d_M}$, where $M \in \{L, V, A\}$, and $S_M$ and $d_M$ corresponds to sequence length and feature dimension of $M$ modality, respectively.

## 4.1 First Crossmodal Projections

Based on Tsai et al. (2019), the MulT model uses crossmodal projections to find relevant patterns of modality $M_1$ in modality $M_2$ space ($M_1 \rightarrow M_2$), where $M_1 \neq M_2$, and $M_1, M_2 \in \{L, V, A\}$. We are using exactly these kinds of projections as our first module. We proceed to describe how is the information flow across these low-level projections.

### 4.1.1 Temporal Convolutions and Positional Encoding

According to Tsai et al. (2019), as a complement to the crossmodal blocks and to ensure that the input sequence has sufficient awareness of its neighbors, they proposed

to use 1D temporal convolution layers and positional encoding before the crossmodal projections.

Temporal convolution layers are described as follows:

$$\overline{X}_M = \textbf{Conv1D}(X_M, k_M), \tag{4.1}$$

where $\overline{X}_M \in \mathbb{R}^{S_M \times d}$, $d$ is a common dimension for all modalities, and $k_M$ is the convolution kernel for modality $M$.

Adding token order within the sequence to the model is useful for crossmodal projections. Positional encoding is described as follows:

$$\textbf{PE}(p, i) = \begin{cases} \sin(\frac{p}{10000^{i/d}}), & \text{if } i \equiv_2 0 \\ \cos(\frac{p}{10000^{i-1/d}}), & \text{if } i \equiv_2 1 \end{cases}, \tag{4.2}$$

where $p$ is the token position, and $i = 0, 1, ..., d-1$, corresponds to the $i$th element of the embedding vector of dimension $d$. Note that the encoding dimension should be divisible by 2 ($d \equiv_2 0$).

Hence, the input sequence is now given by:

$$\hat{X}_M = \begin{bmatrix} \overline{X}_M^{(0)} + [\textbf{PE}(0, i)]_{i=0,1,...,d-1} \\ \overline{X}_M^{(1)} + [\textbf{PE}(1, i)]_{i=0,1,...,d-1} \\ ... \\ \overline{X}_M^{(S_M-1)} + [\textbf{PE}(S_M - 1, i)]_{i=0,1,...,d-1} \end{bmatrix} \in \mathbb{R}^{S_M \times d} \tag{4.3}$$

### 4.1.2 Crossmodal Transformer

The crossmodal transformer block inspired the self-attention mechanism based on Vaswani et al. (2017). This block takes a pair $(M_1, M_2)$ of modalities and is trying to adapt $M_1$ into $M_2$, $(M_1 \rightarrow M_2)$. Let be $\hat{X}_{M_1} \in \mathbb{R}^{S_{M_1} \times d}, \hat{X}_{M_2} \in \mathbb{R}^{S_{M_2} \times d}$ the corresponding input sequences. In Vaswani et al. (2017), we have a **single attention** with keys $K_M$, queries $Q_M$, and values $V_M$. On the other hand, in **crossmodal attention** we have queries $Q_{M_2} = \hat{X}_{M_2} W_{Q_{M_2}}$ of the modality that we want to map to, while we have keys $K_{M_1} = \hat{X}_{M_1} W_{K_{M_1}}$ and values $V_{M_1} = \hat{X}_{M_1} W_{V_{M_1}}$ from the modality

that we want to map. Note that $W_{Q_{M_2}} \in \mathbb{R}^{d \times d_k}, W_{K_{M_1}} \in \mathbb{R}^{d \times d_k}$, and $W_{V_{M_1}} \in \mathbb{R}^{d \times d_v}$ are learnable weights, and $d_k, d_v$ are variable dimensions fixed to $d$ according to Tsai et al. (2019).

We define this crossmodal adaptation from $M_1$ to $M_2$ as:

$$Y_{M_2} = \underset{M_1 \to M_2}{\mathbf{CM}} (\hat{X}_{M_2}, \hat{X}_{M_1}),$$

where $Y_{M_2} \in \mathbb{R}^{S_{M_2} \times (d_v = d)}$, and

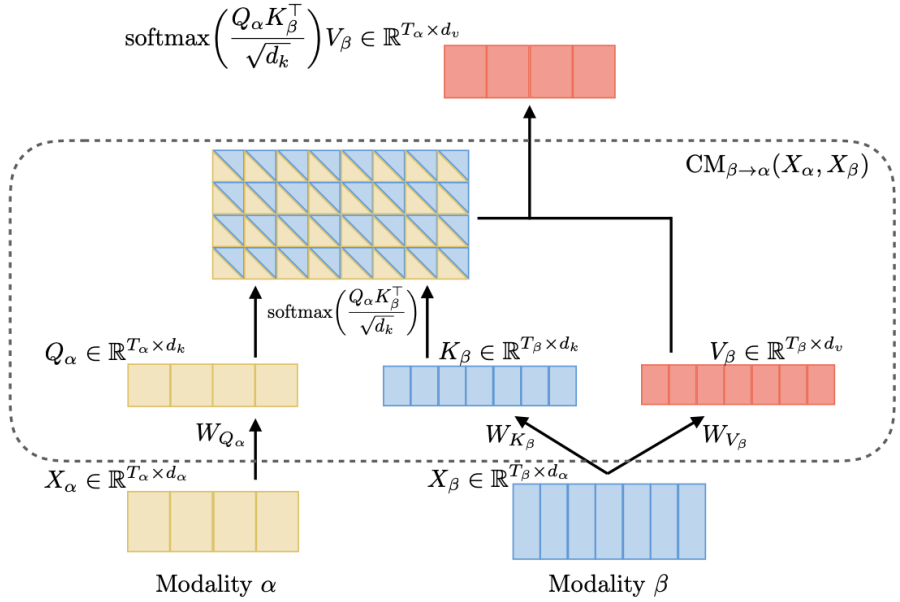
$$\begin{aligned} Y_{M_2} &= \underset{M_1 \to M_2}{\mathbf{CM}} (\hat{X}_{M_2}, \hat{X}_{M_1}) && (4.4) \\ &= \text{softmax}\left(\frac{Q_{M_2} K_{M_1}^T}{\sqrt{d_k = d}}\right) V_{M_1} && (4.5) \\ &= \text{softmax}\left(\frac{\hat{X}_{M_2} W_{Q_{M_2}} W_{K_{M_1}}^T \hat{X}_{M_1}^T}{\sqrt{d}}\right) \hat{X}_{M_1} W_{V_{M_1}} && (4.6) \end{aligned}$$

A clear visualization of this crossmodal attention mechanism between modalities $M_1$ and $M_2$ is given in Figure 4.1a.

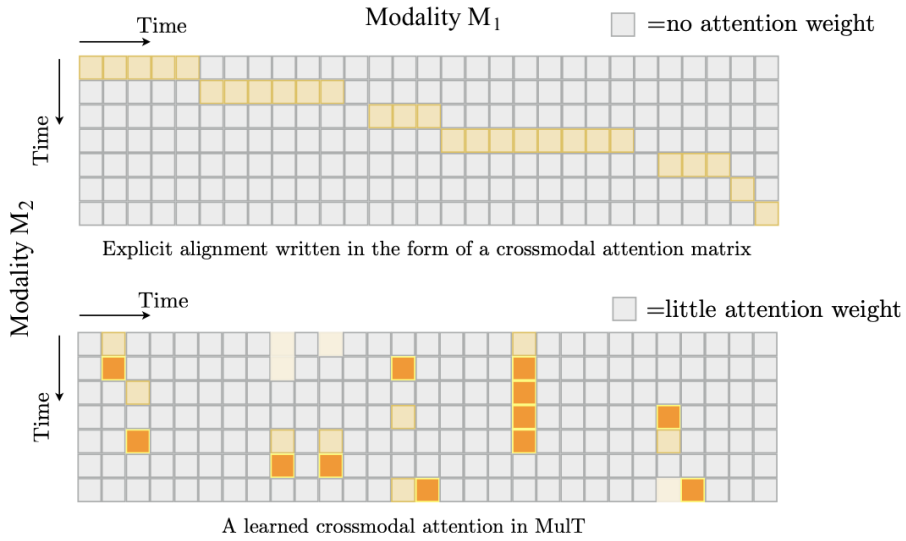
Furthermore, the $(i, j)$th entry of the scaled softmax measures the attention given by the $i$th time step of modality $M_2$ to the $j$th time step of modality $M_1$. Hence, the $i$th sequence token of $Y_{M_2}$ is interpreted as a weighted summary of $V_{M_1}$ tokens. The $i$th row determines this weight in the softmax part. An illustration of that can be seen in Figure 4.1b.

Finally, as explained in Tsai et al. (2019), a normalization layer is followed by a point-wise feed-forward layer. A residual connection is applied to the crossmodal attention module, getting one crossmodal transformer layer. Stacking $D = 3$ crossmodal transformer layers makes the complete Crossmodal Transformer block which is visualized in Figure 4.2.

Note that in Figure 4.2, we have hidden states $Z_{M_{1,2}}^{[i]}$ given by each feed-forward step for $i = 1, 2, ..., D$ of the Crossmodal Transformer layers. This feed-forward




**(a)** Crossmodal attention between sequences $\hat{X}_{M_1} = X_\beta$ and $\hat{X}_{M_2} = X_\alpha$ where $\alpha = M_2, \beta = M_1$, and $d_\alpha = d_\beta = d_k = d_v = d$.




**(b)** Visualization of the time-step (sequence tokens) attention between modality $M_1$ and $M_2$.

**Figure 4.1:** Examples of crossmodal attention taken from Tsai et al. (2019).

mechanism is described as follows using that $Z_{M_1}^{[0]} = \hat{X}_{M_1}$:

$$Z_{M_1 \to M_2}^{[0]} = Z_{M_2}^{[0]} = \hat{X}_{M_2}, \tag{4.7}$$

$$\hat{Z}_{M_1 \to M_2}^{[i]} = \underset{M_1 \to M_2}{\mathbf{CM}} \left[ \mathbf{LN}\left(Z_{M_1 \to M_2}^{[i-1]}\right), \mathbf{LN}\left(Z_{M_1}^{[0]}\right) \right] + \mathbf{LN}\left(Z_{M_1 \to M_2}^{[i-1]}\right), \tag{4.8}$$

$$Z_{M_1 \to M_2}^{[i]} = f_{\theta_{M_1 \to M_2}^{[i]}} \left[ \mathbf{LN}\left(\hat{Z}_{M_1 \to M_2}^{[i]}\right) \right] + \mathbf{LN}\left(\hat{Z}_{M_1 \to M_2}^{[i]}\right), \tag{4.9}$$

where $f_\theta$ is a point-wise feed-forward layer parametrized by $\theta$, and $\mathbf{LN}()$ means a normalization step.

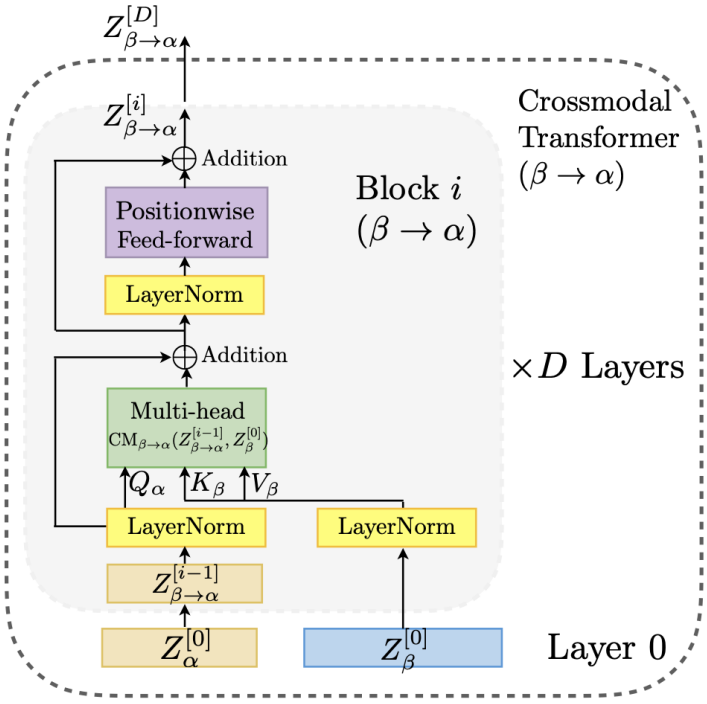



**Figure 4.2:** Crossmodal Transformer (CT) block compounded by $D$ CT layers, where $\alpha = M_2$ and $\beta = M_1$. Image from Tsai et al. (2019).

Under Rodríguez-Bribiesca et al. (2021) and Tsai et al. (2019), if we have three different modalities $(L, V, A)$, the MulT model can perform six various Crossmodal Transformers (CT) given by a combination of all possible pairs. These six CT output sequences are used as input for the second level of crossmodal projections.

## 4.2 Crossmodal Biprojection

*This section is one of the most important in the proposal because it constitutes the primary motivation (the biprojection).*

Taking the six possible Crossmodal Transformers (CT) between $(L, V, A)$ modalities, second crossmodal attention is proposed to ensure that the adaptation to a modality $M_3$ contains information of the other set of two modalities $M_1$ and $M_2$.

We are using the CT output sequences of $(M_1 \to M_2)$ and $(M_2 \to M_1)$ in order to perform a third CT $(M_1 \to M_2 \rightrightarrows M_3)$ or $(M_2 \to M_1 \rightrightarrows M_3)$.

### 4.2.1 Second Crossmodal Transformers

Similar to the first level of crossmodal transformers, we use the exact attention mechanism described in Equation (4.6).

Following the last section notation, let be $\hat{X}_{M_3} \in \mathbb{R}^{S_{M_3} \times d}$ the input feature sequence defined in Equation (4.3), and $Z^{[D]}_{M_1 \to M_2} \in \mathbb{R}^{S_{M_2} \times d}$ the last hidden state of a CT. Hence, crossmodal attention for $M_3$ is given by $Y^{(1,2)}_{M_3} \in \mathbb{R}^{S_{M_3} \times d}$ as follows:

$$
\begin{aligned}
Y^{(1)}_{M_3} &= \underset{M_1 \to M_2 \rightrightarrows M_3}{\mathbf{CM}}(\hat{X}_{M_3}, Z^{[D]}_{M_1 \to M_2}) && (4.10)\\
&= \text{softmax}\left(\frac{Q_{M_3} K^T_{M_1 \to M_2}}{\sqrt{d}}\right) V_{M_1 \to M_2} && (4.11)\\
&= \text{softmax}\left(\frac{\hat{X}_{M_3} W_{Q_{M_3}} W^T_{K_{M_1 \to M_2}} Z^{[D]}_{M_1 \to M_2}{}^T}{\sqrt{d}}\right) Z^{[D]}_{M_1 \to M_2} W_{V_{M_1 \to M_2}}, && (4.12)
\end{aligned}
$$

and analogous, $Y^{(2)}_{M_3}$ is defined just permuting the order of the first projections $M_1$ and $M_2$ as:

$$
Y^{(2)}_{M_3} = \underset{M_2 \to M_1 \rightrightarrows M_3}{\mathbf{CM}}(\hat{X}_{M_3}, Z^{[D]}_{M_2 \to M_1}), \tag{4.13}
$$

where $W_{Q_{M_3}}, W_{K_{M_1 \to M_2}}, W_{V_{M_1 \to M_2}} \in \mathbb{R}^{d \times d}$ are learnable weights.

Note that in this case, the $(i, j)$th entry of the softmax part is measuring the attention given by the $i$th time step (or token) of modality $M_3$ to the $j$th time step of

modality $M_2$ which at the same time, the $j$th time step of $M_2$ is a weighted summary of $V_{M_1}$. It is the main reason to interpret the second crossmodal projection to modality $M_3$ as a weighted summary of the other modalities $M_1$ and $M_2$.

Then, we complete the Crossmodal Transformer (CT) architecture stacking again $D = 3$, CT layers as described below:

$$Z^{[0]}_{M_1 \to M_2 \rightrightarrows M_3} = Z^{[0]}_{M_3} = \hat{X}_{M_3}, \tag{4.14}$$

$$\hat{Z}^{[i]}_{M_1 \to M_2 \rightrightarrows M_3} = \underset{M_1 \to M_2 \rightrightarrows M_3}{\mathbf{CM}} \left[ \mathbf{LN}\left(Z^{[i-1]}_{M_1 \to M_2 \rightrightarrows M_3}\right), \mathbf{LN}\left(Z^{[D]}_{M_1 \to M_2}\right) \right] + \mathbf{LN}\left(Z^{[i-1]}_{M_1 \to M_2 \rightrightarrows M_3}\right), \tag{4.15}$$

$$Z^{[i]}_{M_1 \to M_2 \rightrightarrows M_3} = f_{\theta^{[i]}_{M_1 \to M_2 \rightrightarrows M_3}} \left[ \mathbf{LN}\left(\hat{Z}^{[i]}_{M_1 \to M_2 \rightrightarrows M_3}\right) \right] + \mathbf{LN}\left(\hat{Z}^{[i]}_{M_1 \to M_2 \rightrightarrows M_3}\right) \tag{4.16}$$

Once we have the CT's last output sequence, we obtain a single modality vector for classification. Note that at this time, for each modality, we have two sequence representations. In the following subsection, there is an explanation of how these two modality sequences are fused.

## 4.3 Modalities Fusion with FGMU

*This section is also one of the most critical parts of the proposal because it contains the proposed fusion with the Fusion GMU (FGMU), which substitute the transformer in the MulT model.*

We have acquired two sequence representations for one modality which are given by $Z^{[D]}_{M_1 \to M_2 \rightrightarrows M_3}$ and $Z^{[D]}_{M_2 \to M_1 \rightrightarrows M_3}$. Following Tsai et al. (2019) and Rodríguez-Bribiesca et al. (2021), last time step (last token) is used for prediction. We are using a BERT encoder for modality L. BERT has a special token at the beginning called [CLS], which has all the information for classifying tasks. Thus, when we perform a CT projecting to the $L$ space, we use the first token (time step) for classification because this is the corresponding attention of the [CLS] token.

Let be

$$\begin{aligned} L_1 = L_{M_1 \to M_2 \rightrightarrows M_3} &= \left[ Z^{[D]}_{M_1 \to M_2 \rightrightarrows M_3} \right]_{S_{M_3}-1} \in \mathbb{R}^{1 \times d}, \text{ and} \\ L_2 = L_{M_2 \to M_1 \rightrightarrows M_3} &= \left[ Z^{[D]}_{M_2 \to M_1 \rightrightarrows M_3} \right]_{S_{M_3}-1} \in \mathbb{R}^{1 \times d}. \end{aligned}$$

A concatenation similar to $[Z^{[D]}_{M_1 \to M_3}, Z^{[D]}_{M_2 \to M_3}]$ is used for prediction in both cases of Tsai et al. (2019) and Rodríguez-Bribiesca et al. (2021). Notice that they have one crossmodal projection level to modality $M_3$, so they need to find common patterns on each CT sequence output because these CT outputs come from different modalities ($M_1$ and $M_2$).

It is not the case with our architecture because $L_1$ and $L_2$ come from a similar set of low-level crossmodal projections, ($M_1$ adapted to $M_2$ and $M_2$ adapted to $M_1$).

Hence, instead of finding patterns between these CT outputs, we are fusing dynamically both $L_1$ and $L_2$ with a proposed fusion GMU module based on Arevalo et al. (2017). The proposed Fusion GMU (FGMU) is visualized in Figure 4.3. This module is described as follows using that $x_v = L_1$ and $x_t = L_2$:

$$h_v = \tanh(W_v x_v), \tag{4.17}$$

$$h_t = \tanh(W_t x_t), \tag{4.18}$$

$$z = \sigma\left(W_z [x_v, x_t]\right), \tag{4.19}$$

$$H_{M_3} = h = z h_v x_v + (1 - z) h_t x_t, \tag{4.20}$$

where $W_v, W_t \in \mathbb{R}^{d \times d}, W_z \in \mathbb{R}^{d \times 2d}$ are the weights to be learned, and [,] is the concatenation operation.

In summary, we acquire a fused feature vector $H_{M_3}$ for modality $M_3$, which is used to predict. To clarify, we are getting three different $H_{M_3}$ vectors for each modality $(L, V, A)$. These feature vectors are denoted as $H_L, H_V$, and $H_A$.

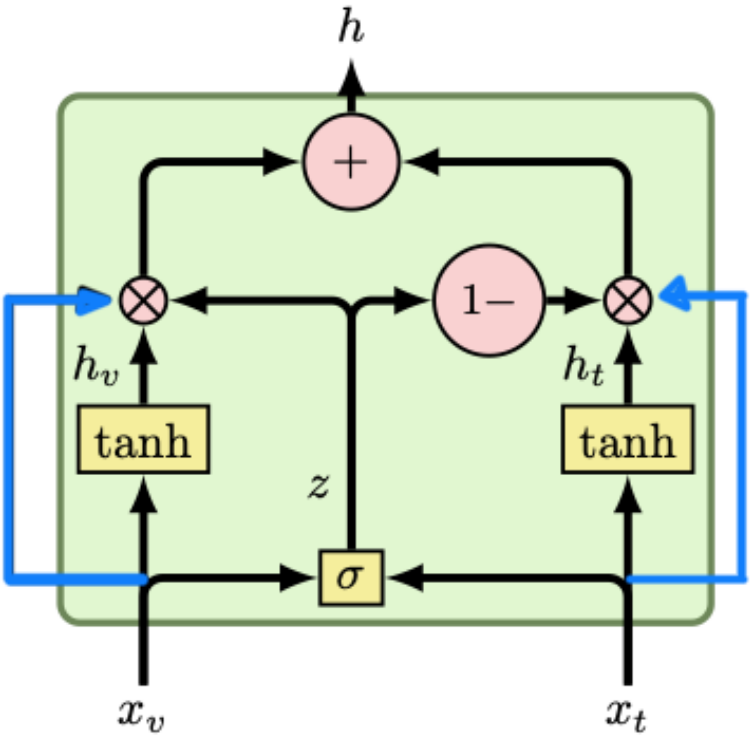



**Figure 4.3:** Proposed Fusion GMU (FGMU) for a dynamic weighted combination of two modality vectors $x_v$ and $x_t$ based on Arevalo et al. (2017). Blue lines are the proposed residual connections.

## 4.4 Simple Parallel Architecture

We have already explained the biprojection part of our proposed architecture **BP-MulT**. In order to decrease the architecture depth effect, we also proposed a **simple parallel modality fusion** method that has inspiration in early, late, and hybrid fusions, Sourav and Ouyang (2021). This parallel architecture is considered the part that helps to not over-fit in the **BPMulT**. .

Let be $\overline{X}_M \in \mathbb{R}^{S_M \times d}$ the feature sequence for modality $M$ as we have in Equation (4.1). To make our model simple, we are reducing the sequence dimension of every modality to a fixed $S_m = 32$. This reduction is made by:

$$\widehat{X}_M = W_m \overline{X}_M, \tag{4.21}$$

where $\widehat{X}_M \in \mathbb{R}^{S_m \times d}$, and $W_m \in \mathbb{R}^{S_m \times S_M}$ a learnable weight matrix.

Then, we proceed to compute its positional embedding as in Equation (4.2), and we obtain a feature sequence $\hat{X}_M^m \in \mathbb{R}^{S_m \times d}$ defined analogously as in (4.3).

### 4.4.1 Self-Attention

For every $\hat{X}_M^m$ of each $M \in \{L, V, A\}$ we perform a self-attention based on Vaswani et al. (2017). This self-attention could be seen as the crossmodal attention defined in Equation (4.6) having keys ($K_M$), queries ($Q_M$), and values ($V_M$) of a single modality.

Hence, we have the feature sequence attention defined as:

$$\begin{aligned} Y_M^m &= \underset{M}{\mathbf{SA}}\left(\hat{X}_M^m\right) && (4.22)\\ &= \text{softmax}\left(\frac{Q_M K_M^T}{\sqrt{d}}\right) V_M && (4.23)\\ &= \text{softmax}\left(\frac{\hat{X}_M^m W_{Q_M} W_{K_M}^T \hat{X}_M^{mT}}{\sqrt{d}}\right) \hat{X}_M^m W_{V_M}, && (4.24) \end{aligned}$$

where $Y_M^m \in \mathbb{R}^{S_m \times d}$, and $W_{K_M}, W_{Q_M}, W_{V_M} \in \mathbb{R}^{d \times d}$ are the weight to be learned.

Finally, we use $D = 3$ stacked self-attention layers to complete the transformer

block as follows:

$$Z_M^{[0]} = \hat{X}_M^m, \tag{4.25}$$

$$\hat{Z}_{M_1 \to M_2}^{[i]} = \underset{M}{\mathbf{SA}} \left[ \mathbf{LN}\left(Z_M^{[i-1]}\right), \mathbf{LN}\left(Z_M^{[0]}\right) \right] + \mathbf{LN}\left(Z_M^{[i-1]}\right), \tag{4.26}$$

$$Z_M^{[i]} = f_{\theta_M^{[i]}} \left[ \mathbf{LN}\left(\hat{Z}_M^{[i]}\right) \right] + \mathbf{LN}\left(\hat{Z}_{M2}^{[i]}\right), \tag{4.27}$$

where $f_{\theta_M^{[i]}}$ and $LN()$ are defined the same way as before.

We now proceed to take the first sequence token of $Z_L^{[D]}$, $L_l$, and the last of $Z_V^{[D]}, Z_A^{[D]}$, $L_v$ and $L_a$ respectively, for classification. Note that every token is a $d$-dimensional vector.

### 4.4.2 Modalities Fusion with Extended FGMU

Since we have three $d$-dimensional representations (one of each modality), for simplicity of this parallel architecture, we are fusing each representation with the proposed **FGMU** in Figure 4.3 but extended for more than two modalities. The equations for this FGMU version, taking $x_i$ as one of $L_l, L_v$, or $L_a$, are:

$$h_i = \tanh(W_i x_i), \tag{4.28}$$

$$z_i = \sigma\left(W_z^i [x_1, x_2, ..., x_k]\right), \tag{4.29}$$

$$H_P = h = \sum_{i=1}^{k} z_i h_i x_i, \tag{4.30}$$

where $W_i \in \mathbb{R}^{d \times d}, W_z^i \in \mathbb{R}^{d \times kd}$ are the weights to be learned for $i = 1, 2, ..., k$ (in this case $k = 3$), and [,] is the concatenation operation. With this method, we have a dynamically combined feature vector $H_P \in \mathbb{R}^{1 \times d}$ representing the **simple parallel attention** of the multimodal sequences that are also used in our classification.

In Figure 4.4 we have an illustration of this simple method proposed, and the proposed **FGMU** extension can be viewed in Figure 4.5.

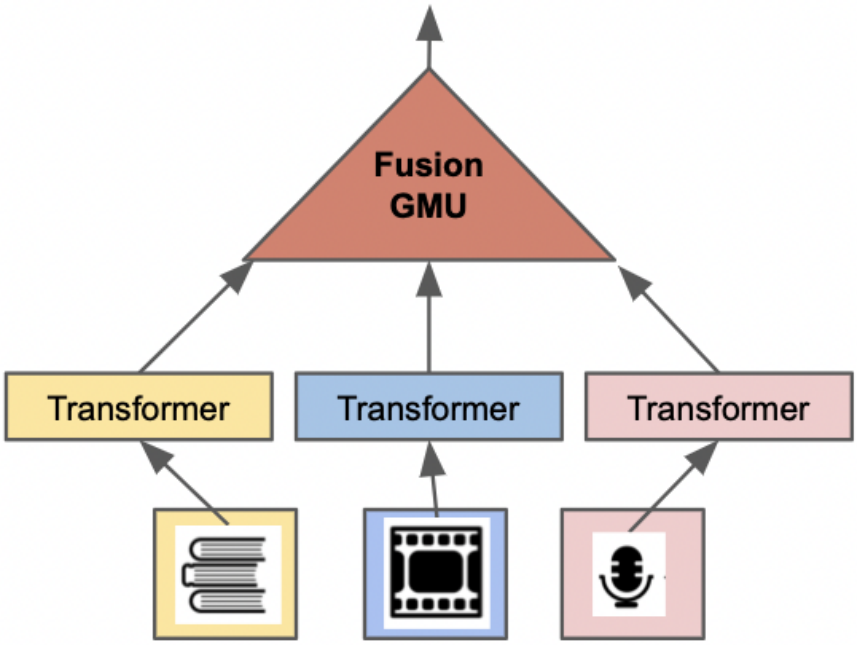



**Figure 4.4:** Proposed Parallel architecture with simple modalities representation and fusion with an FGMU extended to three modalities.

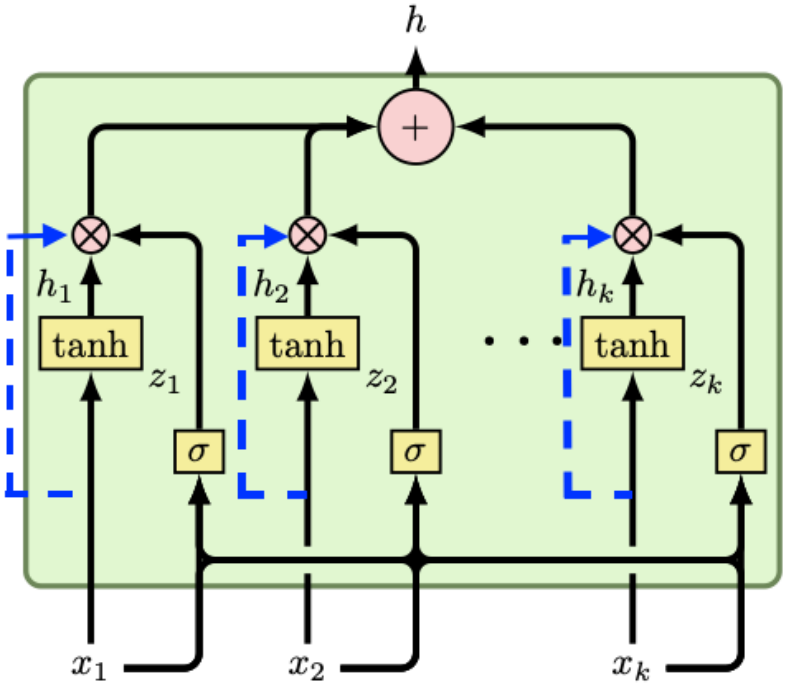



**Figure 4.5:** Proposed Fusion GMU (FGMU) extension for a dynamic weighted combination of three or more modality vectors $x_1, ..., x_k$ based on Arevalo et al. (2017). Blue lines are the proposed residual connections.

## 4.5 Dynamic Modalities Fusion with GMU

In Section 4.3, we have obtained the $H_L, H_V$, and $H_A$ feature vectors representing a modality summary that takes into account the attention given between the other modalities. In Section 4.4, a simple summary vector $H_P$ was computed to recover general and relevant information. Furthermore, depending on the taken dataset, we might have extra non-sequential important features we want to use to classify, e.g., images or metadata. To address this wanted fusion, we worked with a GMU module as it is proposed in Arevalo et al. (2017).

We mainly take our modality feature vectors ($H_L, H_V$, $H_A$) and our simple summary vector $H_P$ into the GMU. Extra relevant features are denoted as $H_E \in \mathbb{R}^{1\times d}$. These features are dynamically combined with the original GMU explained in Section 3. The GMU module gives an output $O \in \mathbb{R}^c$, where $c$ is the number of classification categories.

Hence, this process is described as:

$$\begin{aligned}
h_1 &= \tanh(W_1 H_L), \\
h_2 &= \tanh(W_2 H_V), \\
h_3 &= \tanh(W_3 H_A), \\
h_4 &= \tanh(W_4 H_P), \\
h_5 &= \tanh(W_5 H_E), \\
z_i &= \sigma\left(W_z^i [H_L, H_V, H_A, H_P, H_E]\right), \text{ for } i = 1, 2, 3, 4, 5, \\
O &= h = \sum_{i=1}^{5} z_i h_i,
\end{aligned} \tag{4.31}$$

where $W_{1,2,3,4,5} \in \mathbb{R}^{c\times d}$ and $W_z^i \in \mathbb{R}^{c\times 5d}$ are learnable weights.

After dynamic fusion, we perform a traditional residual block before giving the final prediction:

$$O_f = W_{f_1}\left[\text{dropout}(\text{ReLu}(W_{f_2} O))\right] + O, \tag{4.32}$$

where $W_{f_1}, W_{f_2} \in \mathbb{R}^{c\times c}$ are weights to be learned.

## 4.6 Tackling Vanishing Gradient

*This section is also one of the essential parts of the proposal because it involves the solution to our third specific motivation.*

One of the most common problems in deep neural networks is the vanishing gradient. Our proposed architecture has three depth primary levels where residual connections are intuitive between levels and is a potential tool for preventing this inconvenience.

We introduce two kinds of residual connections to the model. One is a connection between the first crossmodal projections 4.1 and the second 4.2 just at a single token level (specific token used for prediction). Following the preceding notation:

$$L'_1 = \left[Z^{[D]}_{M_1 \to M_2 \rightrightarrows M_3}\right]_{S_{M_3}-1} + \left[Z^{[D]}_{M_1 \to M_2}\right]_{S_{M_2}-1} \in \mathbb{R}^{1\times d}, \text{ and} \tag{4.33}$$

$$L'_2 = \left[Z^{[D]}_{M_2 \to M_1 \rightrightarrows M_3}\right]_{S_{M_3}-1} + \left[Z^{[D]}_{M_2 \to M_1}\right]_{S_{M_1}-1} \in \mathbb{R}^{1\times d}. \tag{4.34}$$

The second residual connection is a token prediction fusion with an FGMU of the first crossmodal projection to the prediction token fusion of the second crossmodal projection. Respective equations are:

$$H'_{M_3} = \textbf{FGMU}\left[Z^{[D]}_{M_1 \to M_2}\right]_{S_{M_2}-1} + H_{M_3} \in \mathbb{R}^{1\times d}. \tag{4.35}$$

An illustration of our proposed **BPMulT** model for multimodal classification can be observed in Figure 4.6 and the proposed compounding parts in Figure 6.2.

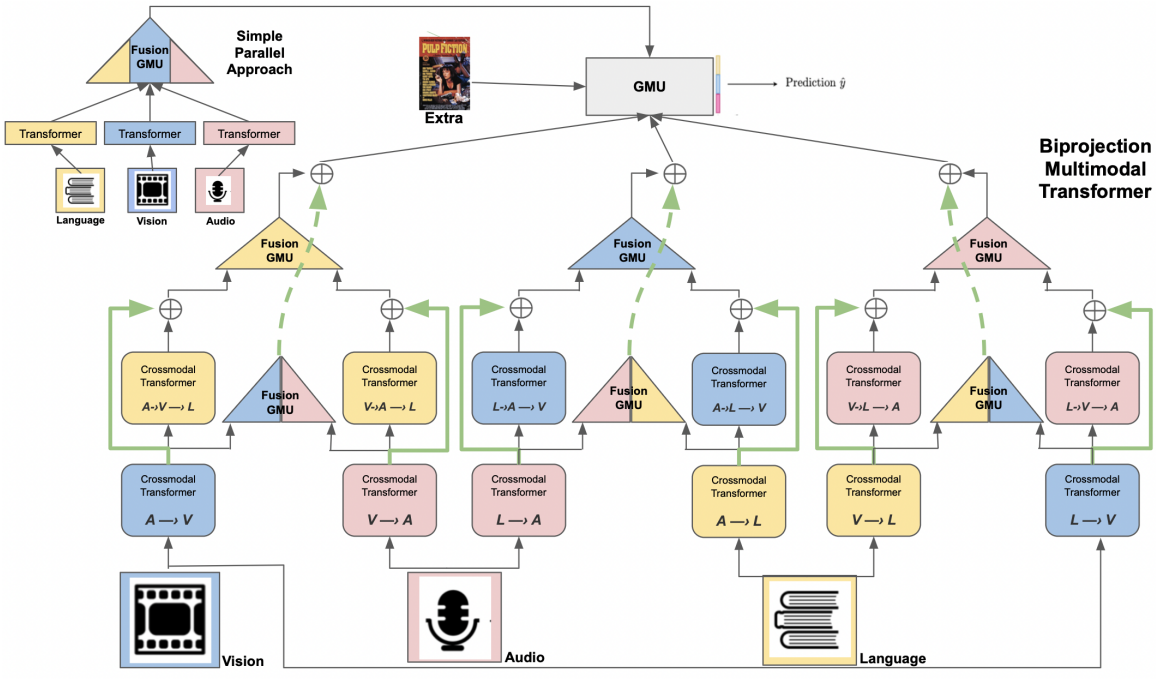



**Figure 4.6:** Proposed Biprojection Multimodal Transformer **BPMulT** architecture for multimodal classification tasks. Yellow blocks correspond to text modality projected Crossmodal Transformers (CT), blue blocks correspond to Video modality projected CT, and red blocks correspond to audio modality projected CT. Its color represents the FGMU as the modalities that they are fusing. The GMU in block color gray weighs to combine the information from a simple parallel architecture, a heavy model, and extra metadata features. The simple parallel part is introduced to reduce the over-fitting of the heavy architecture.

# Chapter 5

# Datasets

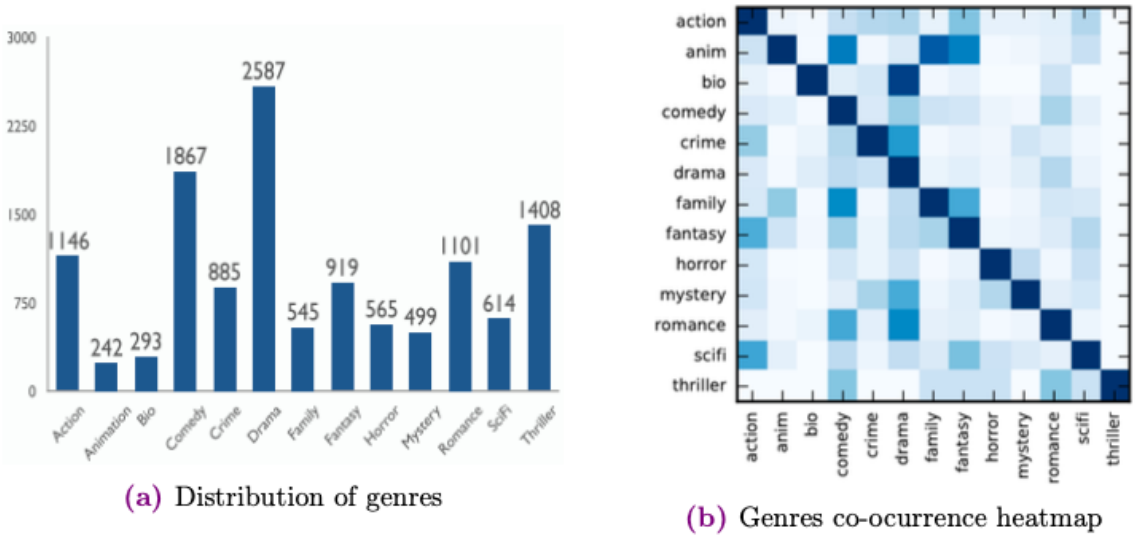
This chapter describes the datasets we perform experiments with to test our proposed model. In each of the following sections, we describe the corresponding dataset within its statistics of class labels, its metrics used to evaluate and have a fair comparison against other models, and a brief description of the SOTA models and their performance for each dataset. Note that we have competitive scores with these SOTA models.

## 5.1 Datasets

### 5.1.1 Moviescope

Movisecope is a large-scale dataset proposed in Cascante-Bonilla et al. (2019), containing 5,043 movies with corresponding video trailers, posters, plots, and metadata. Each movie has an IMDb-based classification of rating and genres. We are using the movie classification by 13 different genres for this work. This classification is multilabel, i.e., a movie can have multiple labels. In Figure 5.1 (b), we can observe how some movie genres appear together more often than others. For example, it is common to label Drama movies as Bio kind and Family movies as Animation. In the case of the label distribution (5.1 (a)), we can observe that the most frequent genre belongs to Drama, followed by Comedy. In contrast, the least frequent genres are Bio

and Animation.




**(a)** Distribution of genres

**(b)** Genres co-ocurrence heatmap

**Figure 5.1:** Label statistics for the Moviescope dataset, Cascante-Bonilla et al. (2019). On (a), we can observe the label distribution, showing an imbalance problem. On (b), we can observe labels' correlation since it is a multilabel classification.

For our experiments with the Moviescope dataset, we use the preprocessed video, audio, and poster features given by Cascante-Bonilla et al. (2019). We have 200 feature vectors of size 4096 extracted from a pre-trained VGG16 CNN. For the trailer audio, we have the log-mel scaled power spectrograms processed by a Convolutional Recurrent Neural Network and a feature vector given by a pre-trained VGG16 CNN for the poster. Section 3.3 describes with detail this feature extraction. Furthermore, we use the text modality's pre-trained "uncased-base" BERT model to represent the synopsis sequence vectors. This BERT representation has 512 vectors with 768 features, so the synopsis was trimmed to 512-word tokens.

In the Moviescope dataset, the current SOTA model, up to our knowledge, is the Mult-GMU proposed by Rodríguez-Bribiesca et al. (2021). This model uses exact text (BERT), video (VGG16-CNN), audio (log-mel-spect-RNN), and poster (VGG16) features like ours. The metrics used to analyze and evaluate classification's performance are the Precision Score averaged by Micro, Macro, and Samples modes of calculation. We explain these averages and metrics in Section 2.3. Here is the formula:

$$\text{AP} = \sum_{n} (R_n - R_{n-1}) P_n. \tag{5.1}$$

### 5.1.2 MM-IMDb

MM-IMDb is the largest multimodal movie genre classification dataset proposed along the GMU publication Arevalo et al. (2017). MM-IMDb contains 25,959 movies with corresponding plots, posters, genres, and metadata. We are using the movie classification by 23 different genres for this work to have a fair comparison with SOTA results. This classification is multilabel, the same for the Moviescope dataset. We have three MM-IMDb splits corresponding to training, development, and test subsets containing 15552, 2608, and 7799, respectively (60%, 10%, 30%). The distribution of samples is listed in Table 5.1. In Figure 5.2, we have the movie genres co-occurrence matrix, and we observe that the Drama genre has a significant co-occurrence with most genres. It is something to be careful of because our proposed model could always classify a movie as a Drama. On the other hand, note that if a movie classification is Drama, it is not common to have co-occurrence with other genres. Let us take the Romance genre to have an example. If a movie has the Romance genre, it is highly probable that it also has the Drama genre. In contrast, if a movie has the Drama genre, it is unnecessary to have any other genre co-occurrence.

We use the 4096 preprocessed poster features given by Arevalo et al. (2017) for our experiments with the MM-IMDb dataset. Furthermore, we use the text modality's pre-trained "uncased-base" BERT model to represent the synopsis sequence vectors. As mentioned, BERT representation has 512 vectors with 768 features, trimming the synopsis.

In the MM-IMDb dataset, the current SOTA model, up to our knowledge, is the MMBT proposed by Kiela et al. (2019). This model uses the same text (BERT) but with a concatenation of the poster processed by a ResNet. The metrics used to analyze and evaluate classification's performance are the F1 Score averaged by Micro, Macro, Weighted, and Samples modes of calculation. We explain these averages and metrics in Section 2.3. Here is the formula:

MM-IMDb

| Genre | Train | Develop | Test | Genre | Train | Develop | Test |
|---|---|---|---|---|---|---|---|
| Drama | 8424 | 1401 | 4142 | Family | 978 | 172 | 518 |
| Comedy | 5108 | 873 | 2611 | Biography | 788 | 144 | 411 |
| Romance | 3226 | 548 | 1590 | War | 806 | 128 | 401 |
| Thriller | 3113 | 512 | 1567 | History | 680 | 118 | 345 |
| Crime | 2293 | 382 | 1163 | Music | 634 | 100 | 311 |
| Action | 2155 | 351 | 1044 | Animation | 586 | 105 | 306 |
| Adventure | 1611 | 278 | 821 | Musical | 503 | 85 | 253 |
| Horror | 1603 | 275 | 825 | Western | 423 | 72 | 210 |
| Documentary | 1234 | 219 | 629 | Sport | 379 | 64 | 191 |
| Mystery | 1231 | 209 | 617 | Short | 281 | 48 | 142 |
| Sci-Fi | 1212 | 193 | 586 | Film-Noir | 202 | 34 | 102 |
| Fantasy | 1162 | 186 | 585 | | | | |

**Table 5.1:** Movie genre distribution along the MM-IMDb dataset splits, Arevalo et al. (2017).

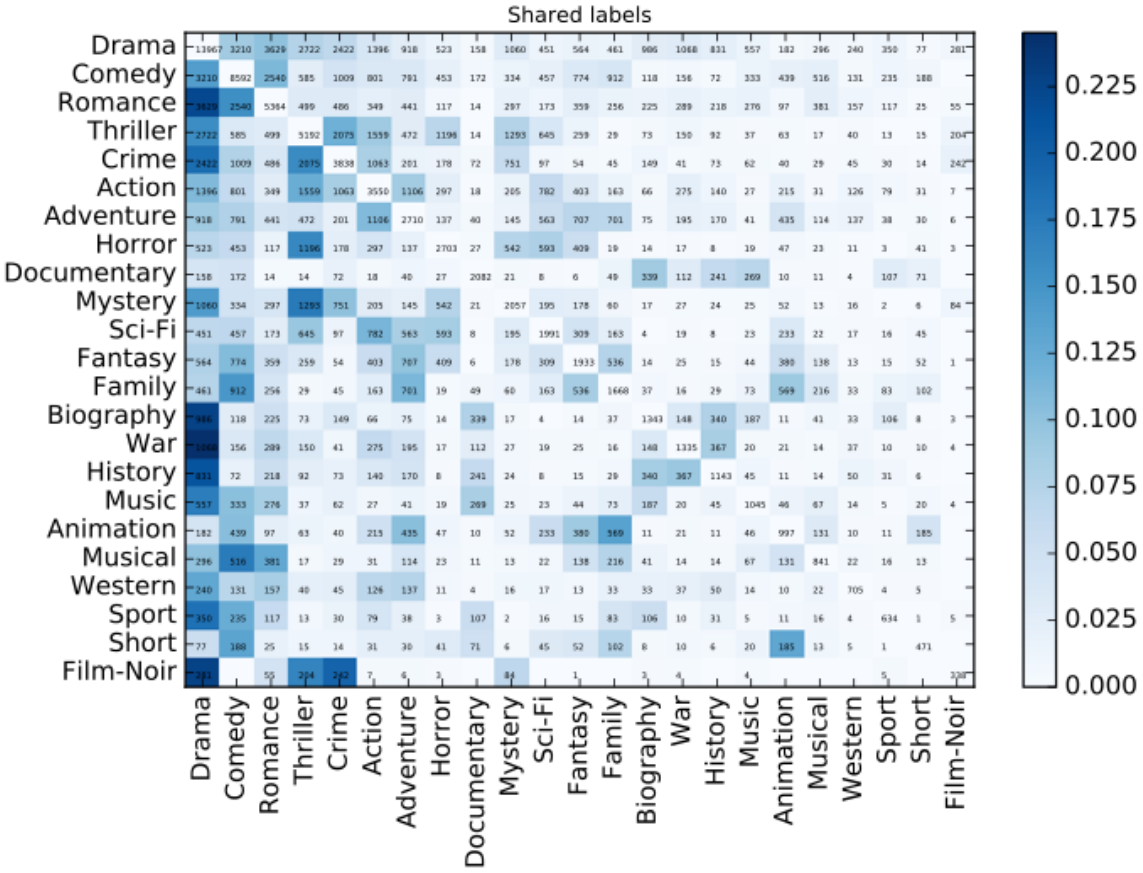



**Figure 5.2:** Movie genre co-occurrence matrix of 23 classes from the MM-IMDb dataset, Arevalo et al. (2017).

$$F1 = 2 * (precision * recall)/(precision + recall). \tag{5.2}$$

### 5.1.3 IEMOCAP

The Interactive Emotional Dyadic Motion Capture (IEMOCAP) is an acted, multimodal, and multispeaker dataset given in Busso et al. (2008). It has 12 hours of video, speech, motion capture of face, and dialogue transcriptions. It consists of dyadic sessions where actors perform scenarios to produce emotional expressions. Multiple annotators annotate the sessions into categorical labels, such as anger, happiness, sadness, and neutrality. Also, it has other sentiment aspects. For this dataset, we are using a preprocessed version given by Carnegie Mellon University on its web page. We take the segmented dataset by a sequence length of twenty with audio, video, and text features. In this preprocessed dataset, we have just four classification labels corresponding to the Neutral, Happy, Sad, and Angry emotions. The emotion distribution is given in the Table 5.2.

IEMOCAP

| Emotion | Train | Develop | Test |
|---|---|---|---|
| Neutral | 954 | 358 | 383 |
| Happy | 338 | 116 | 135 |
| Sad | 690 | 188 | 193 |
| Angry | 735 | 136 | 227 |

**Table 5.2:** Emotion distribution along the preprocessed IEMOCAP dataset.

For our experiments with the IEMOCAP dataset, we use the preprocessed text, video, and audio features given by Busso et al. (2008). We selected the text sequence length of twenty on its aligned and unaligned forms. For the aligned, each sequence of text, video, and utterance has a length of twenty. The unaligned form, video, and audio have a length of 500 and 400, respectively. For comparison with prior works (Tsai et al. (2019), Dai, Liu, Yu, and Fung (2020), Dai, Cahyawijaya, Bang, and Fung (2021), D. S. Chauhan, Akhtar, Ekbal, and Bhattacharyya (2019), Zhang et al. (2020)), we also follow the same four selected categories, namely, neutral, happy, sad,

and angry.

To the best of our knowledge, the current SOTA model is the ModTrans-MMEmoRe proposed by Dai et al. (2020) in the aligned sequence and the MulT in the unaligned sequence. The Accuracy and the Area Under the Curve are used to evaluate the classification's performance in the aligned sequence. For the unaligned sequence, we use the Accuracy and the F1 Score. We explain metrics in Section 2.3. Here is the F1 Score formula:

$$F1 = 2 * (precision * recall)/(precision + recall). \tag{5.3}$$

### 5.1.4 CMU-MOSEI

CMU Multimodal Opinion Sentiment and Emotion Intensity (CMU-MOSEI) dataset is the largest dataset of multimodal emotion recognition and is widely used in recent researches, Zadeh, Liang, Poria, Cam-bria, and Morency (2018). It contains 23,457 sentence utterances extracted from YouTube videos with a balance in gender to classify six emotions Anger, Disgust, Fear, Happiness, Sadness, and Surprise. Monologue videos are randomly chosen from various topics and transcribed. In Figure 5.3, we observe the label distribution that this dataset initially had. We consider a cleaned and modified version given in Dai, Cahyawijaya, Liu, and Fung (2021) for our experiment and comparison. This new version contains a little bit less data but with the same emotions for the classification. It also has a better balance between classes and cleaning bad previous data cases. Table 5.3 gives the distribution of this version, and it will be used for our experiments.

For our experiments with the CMU-MOSEI dataset, we use the preprocessed video and audio features given by Dai, Cahyawijaya, Liu, and Fung (2021), which is a cleaned version of the original CMU-MOSEI dataset. We use the pre-trained "uncased-base" BERT model for the text to represent the dialogue sequence vectors. As we mentioned, we trimmed the text tokens to the length of 512.

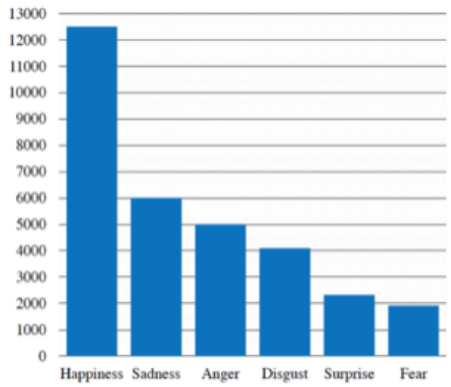



**Figure 5.3:** Emotion distribution along the original CMU-MOSEI dataset.

CMU-MOSEI Unaligned

| Emotion | Train | Develop | Test |
|---|---|---|---|
| Anger | 3267 | 318 | 1015 |
| Disgust | 2738 | 273 | 744 |
| Fear | 1263 | 169 | 371 |
| Happiness | 7587 | 945 | 2220 |
| Sadness | 4026 | 509 | 1066 |
| Surprise | 1465 | 197 | 393 |

**Table 5.3:** Emotion distribution along the preprocessed and modified CMU-MOSEI dataset by Dai, Cahyawijaya, Liu, and Fung (2021).

We follow the same categories and metrics for a fair comparison with the prior work Dai, Cahyawijaya, Liu, and Fung (2021), which is the current state-of-the-art model. These metrics used to evaluate classification's performance are the Weighted-Accuracy and the F1 Score explained in Section 2.3. Here are the formulas:

$$WAcc = \frac{TP \times N/P + TN}{2N}, \tag{5.4}$$

and for the F1 Score:

$$F1 = 2 * (precision * recall)/(precision + recall). \tag{5.5}$$

# Chapter 6

# Experiments and Results

This chapter presents the results obtained by the proposed model (BPMulT) for the task of multimodal classification for predicting movie genres and emotions. We first describe the experimental framework followed to train the models and perform the experiments. We compare the proposal's performance with the state-of-the-art on the Moviescope dataset. Section 6.2 presents the main results for the dataset selected, followed by the ablation experiments in Section 6.3. In Section 6.4, we describe the modalities' relevance to perform a correct classification, where we found the biprojection enriches not-relevant modalities and becomes critical in the classification. One modality finds better patterns in the other modalities, and the biprojection always considers each modality in every step. Finally, Section 6.3 compares the BPMulT against SOTA models in multimodal emotion recognition. Here, we found the robustness of the proposed model in terms of over-fitting and the model's capability to handle modalities as channels of one modality, e.g., different representations of text. We show that the BPMulT architecture performs better than MulT and MulT-GMU with three or more modalities in many tasks.

## 6.1 Experimental Framework

Following previous work (MulT-GMU, Bribiesca 2021), we compare our two base models with the MulT-GMU on Moviescope, which was also compared with the Mul-

timodal BERT (MMBT) and Fast Modal Attention (Fast-MA). For the MM-IMDb dataset, we include the actual results presented in (Arevalo., 2017), and for IEMOCAP and MOSEI, we have the original SOTA results.

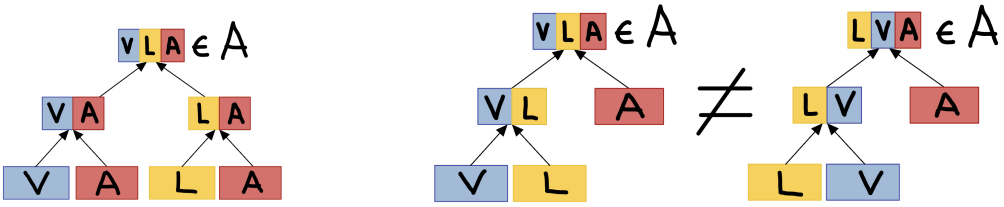
We denote different modalities as V (Video), A (Audio), P (Poster), and T (Text). In the case of the MM-IMDb dataset, we handle V, A, and T as three different text embeddings since this dataset only has text and poster modalities. For our models BPMulT and BPMulT-no-parallel, we show their mean and standard deviation of the performance over five runs with random seeds for all datasets considered.

We select the best hyperparameters for every dataset, including the number of heads, hidden dimension, number of layers in the transformer, batch size, and gradient accumulation step.

We use the Moviescope dataset 5.1.1 to determine the best model configuration (in terms of architecture) for all the experiments below. With this final model, we perform the experiments in the other datasets.

## 6.2 Main Results in Moviescope

The way modalities interact in the multimodal transformer model is the base for developing our architecture. The motivation to create a novel transformer-based architecture is to enrich each modality with information from the others. The primary purpose of this experiment is to improve the best results shown in (Rodriguez, 2021) and its baselines.




**(a)** Single modalities combinations to get an enriched modality A.

**(b)** Joint modalities combinations to get an enriched modality A.

We have the intuition that the **joint projection between two modalities with a third modality is better than just a single projection of one modality to**

**another** as we can see in Figure 6.1b the joint combination and in Figure 6.1a the single combination.

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAPM | MulT-GMU | 79.5±0.5 | 76.4±0.3 | 85.6±0.3 |
| | MulT-GMU-no-transf-encoder | 80.3±0.2 | 76.9±0.2 | 86.1±0.4 |
| TVAP | MMBT | 77.4±0.7 | 74±0.8 | 85.1±0.7 |
| | Fast-MA | 74.9 | 67.5 | 82.3 |
| | MulT | 78.9±0.3 | 75.7±0.5 | 85.6±0.3 |
| | MulT-GMU | 79.8±0.4 | 76.0±0.9 | 86.1±0.4 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.1:** Comparison of our BPMulT model with different modality combinations. Modalities refers to text-video-audio-poster (TVAP) and text-video-audio-poster-metadata (TVAPM). The models use just the specified modalities' information. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP) and sample (sAP) averaged.

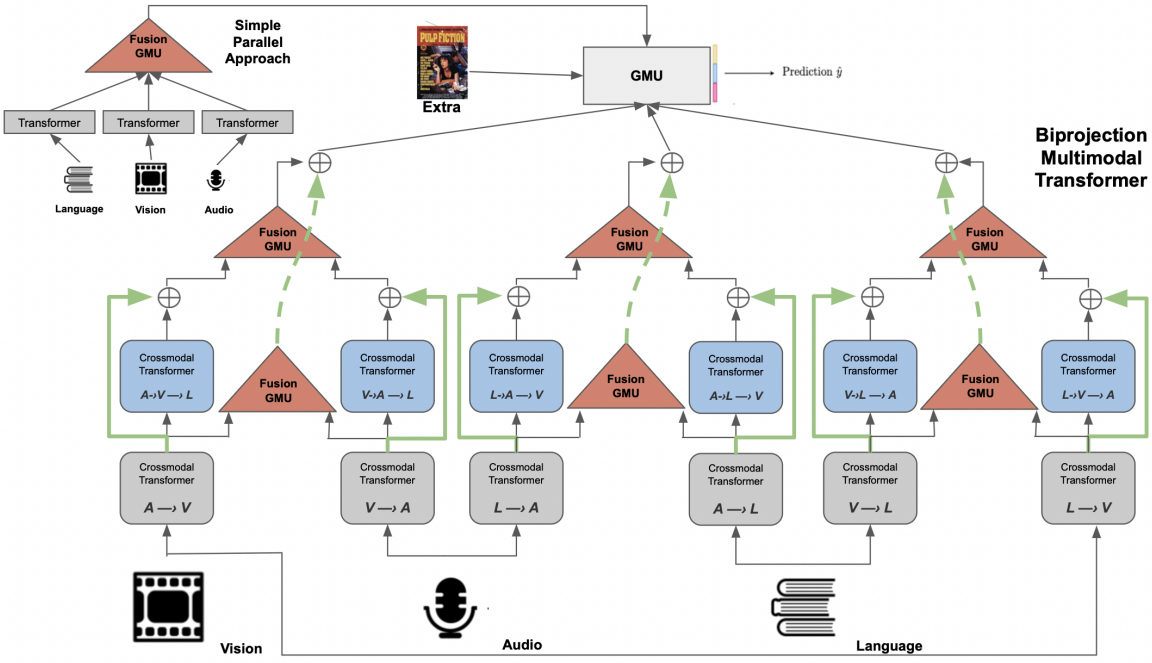



**Figure 6.2:** Proposed **BPMulT** architecture for a multimodal classification task. The blue part corresponds to biprojection modules. Red triangles are the GMU modules for fusion. Green arrows correspond to residual connections from the first projections. Gray modules are the first crossmodal transformers. We can observe the parallel reduced fusion step in the upper left corner. The GMU weighs the heavy and simple architectures and the metadata features to give a classification.

We first compare the Moviescope baseline models, Fast-MA, MulT, and MulT-

GMU, with our BPMulT with and without parallel fusion in the final GMU. Results on the modality setting (TVAP) are shown in Table 6.1. Note that our proposed models outperform the SOTA models by 2% on each metric even when the modality setting of the MulT-GMU is TVAPM (with metadata).

In Figure 6.2, we observe our proposed BPMulT model with the reduced parallel fusion to not overfit the dataset. The BPMulT-no-parallel model is the same architecture removing the upper left corner part (the reduced parallel fusion).

## 6.3 BPMulT Ablation Experiments

We propose a set of experiments to evaluate the main components of our BPMulT model where we remove each one of them. These main components correspond to the FGMU modules for fusion information of a single modality and the residual connections proposed through the crossmodal transformers' first and second levels. This section performs an ablation study to measure the impact on the model's performance.

### 6.3.1 No-FGMU modules

Following the results posted in (Rodríguez, 2021), the best classification results are achieved when the transformer encoder is removed. We propose fusing one modality's different information with an FGMU, e.g., fusing FGMU(A—›L and V—›L). To obtain the relevance of this FGMU module, we follow the idea of (Rodríguez, 2021) removing it and passing the information directly to the GMU. In Figure 6.3, we observe the BPMulT-no-parallel architecture without the red triangles corresponding to the FGMU modules. In the Table 6.2 there is the result of classification for the considered metrics. Note that the performance is affected by removing this FGMU for single-modality information fusion. The GMU at the top has more information to fuse, which is not a good practice. The normal GMU is tested on fusing information of different modalities. In this case, the GMU is also fusing features from the same modality and similar representation, e.g., the set of two modalities to the third ((V›A)—›L and (A›V)—›L).

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | BPMulT-no-FGMU (ours) | 79.9±0.25 | 76.54±0.3 | 86.2±0.4 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.2:** Comparison of our BPMulT model with the same model removing FGMU modules to obtain a measurement of this module's relevance. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP) and sample (sAP) averaged.

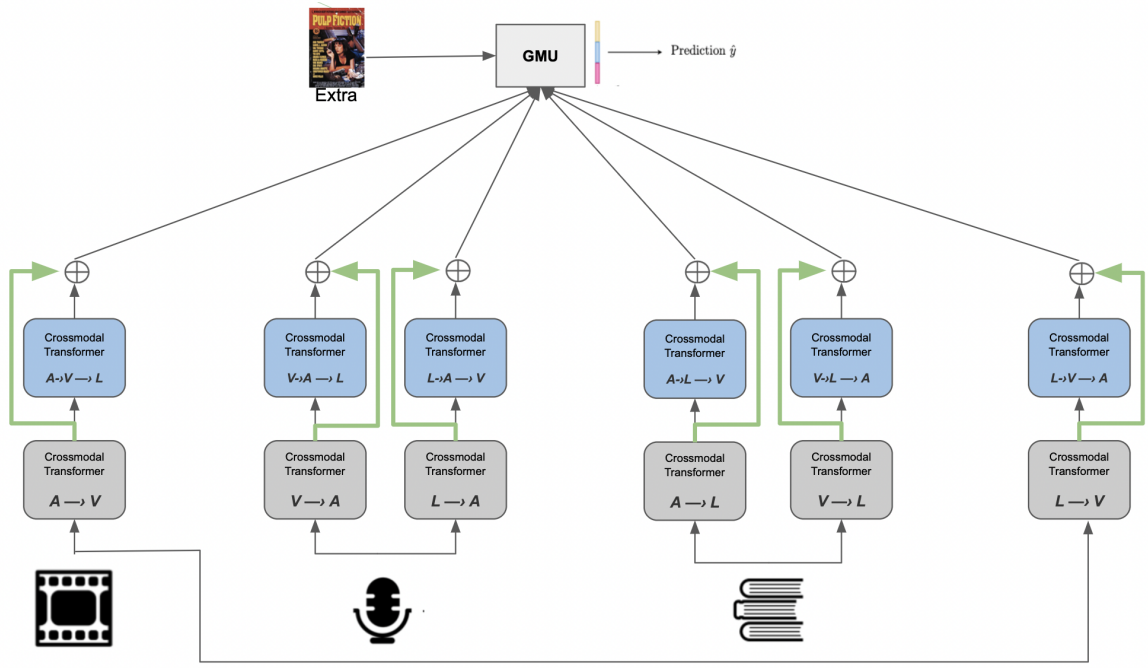



**Figure 6.3:** Proposed **BPMulT-no-parallel** architecture for a multimodal classification task without the FGMU modules for ablation study. The blue part corresponds to biprojection modules, and the green arrows correspond to residual connections from the first projections. Gray modules are the first crossmodal transformers.

### 6.3.2 No-FGMU modules nor Residual Connections

In order to obtain the relevance of the residual connection proposed, we removed them with this experiment. We see if the past information is an important piece. In Figure 6.4, we observe the BPMulT-no-parallel architecture without the red triangles corresponding to the FGMU modules and without the green arrows, which are the residual connections proposed. In the Table 6.3 there is the result of classification for the considered metrics. Remember that the performance is affected by removing this

FGMU for single-modality information fusion. In this case, the performance is favored but is still worse than the BPMulT model by removing the residual connections from one level to another. Note that these residual connections combine information from different modalities because the first level of projections does not have the same modality space. If we do not have any FGMU for fusing this kind of data correctly, the residual is useless in this case. For this reason, the model's performance without residual connections is better when the FGMU modules are omitted.

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | BPMulT-no-FGMU-nor-RC (ours) | 80.5±0.5 | 76.9±0.5 | 86.4±0.3 |
| | BPMulT-no-FGMU (ours) | 79.9±0.25 | 76.54±0.3 | 86.2±0.4 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.3:** Comparison of our BPMulT model with the same model removing FGMU modules and residual connections to obtain a measurement of the connections. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP) and sample (sAP) averaged.

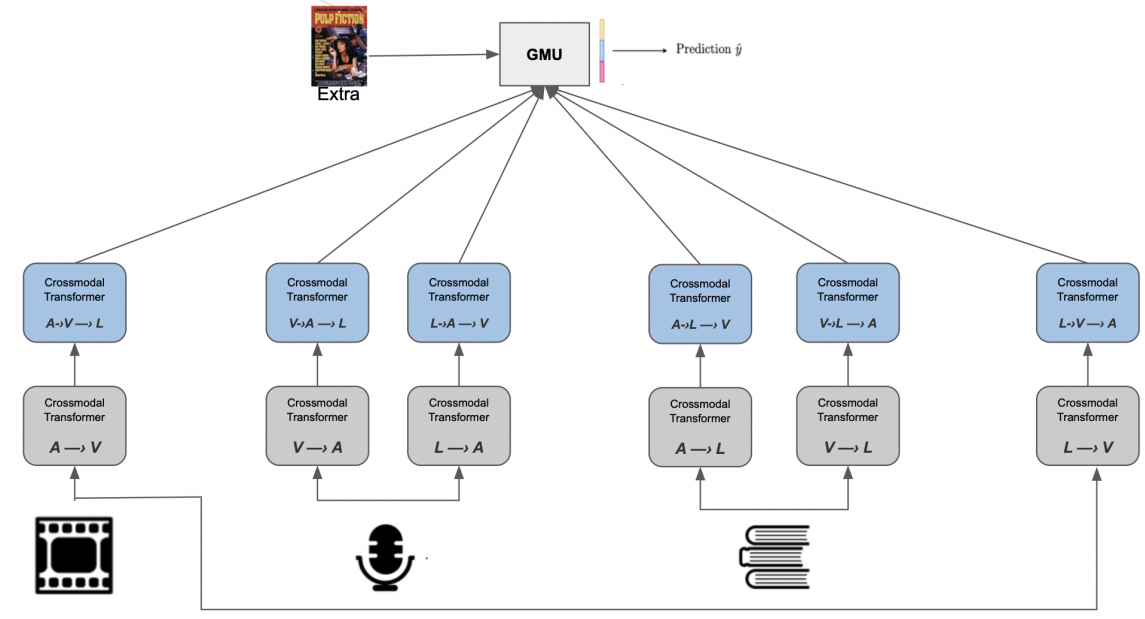



**Figure 6.4:** Proposed **BPMulT-no-parallel** architecture for a multimodal classification task without the FGMU modules and residual connections for ablation study. The blue part corresponds to biprojection modules. Gray modules are the first crossmodal transformers.

### 6.3.3 All Crossmodal Attention Modules at GMU

We obtained the relevance of the FGMU modules and the residual connection proposed. In this experiment, we are looking for details about if a biprojection is useful for the classification. Following (Rodríguez, 2021), the best performance of the MulT-GMU model is when all crossmodal (single projection in gray) are given to the GMU. So, we are giving all the information from the biprojections and the single projections to the top GMU, as shown in Figure 6.6. In the Table 6.4 there is the result of classification for the considered metrics. We remove all FGMU modules and residual connections to get just the crossmodal transformers in their pure form. The performance of this model is worse than every model proposed before. The GMU has much more information, affecting all modalities' fusion performance.

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | BPMulT-all-transformers (ours) | 79.8±0.4 | 76.4±0.3 | 86.1±0.5 |
| | BPMulT-no-FGMU (ours) | 79.9±0.25 | 76.54±0.3 | 86.2±0.4 |
| | BPMulT-no-FGMU-nor-RC (ours) | 80.5±0.5 | 76.9±0.5 | 86.4±0.3 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.4:** Comparison of our BPMulT model with the same model passing all transformer encoders (biprojections and single projections) to obtain a measurement of relevance. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP), and sample (sAP) averaged.

In Figure 6.5, we can note that the activations of the biprojections at the GMU are similar to or better than a single projection. For example, in the Text space, projecting from Video (third bar) is almost the same as the transition Audio to Video and then to text (first bar). On the other hand, the projection of Video to Audio and then to text (second bar) is better than just projection from audio (fourth bar).

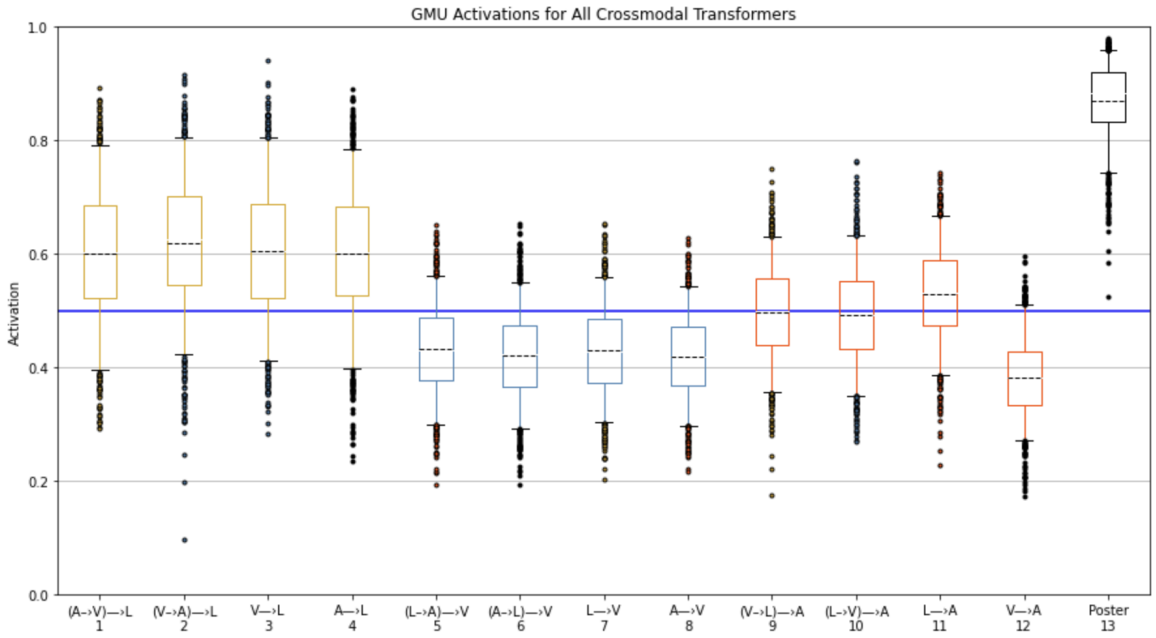



**Figure 6.5:** GMU activations of the experimental proposed architecture passing all crossmodal transformer encoders (biprojections and single projections) for ablation study. Yellow bars correspond to activations of the Language space, blue ones to the Video space, red ones for audio and black ones to the poster features.

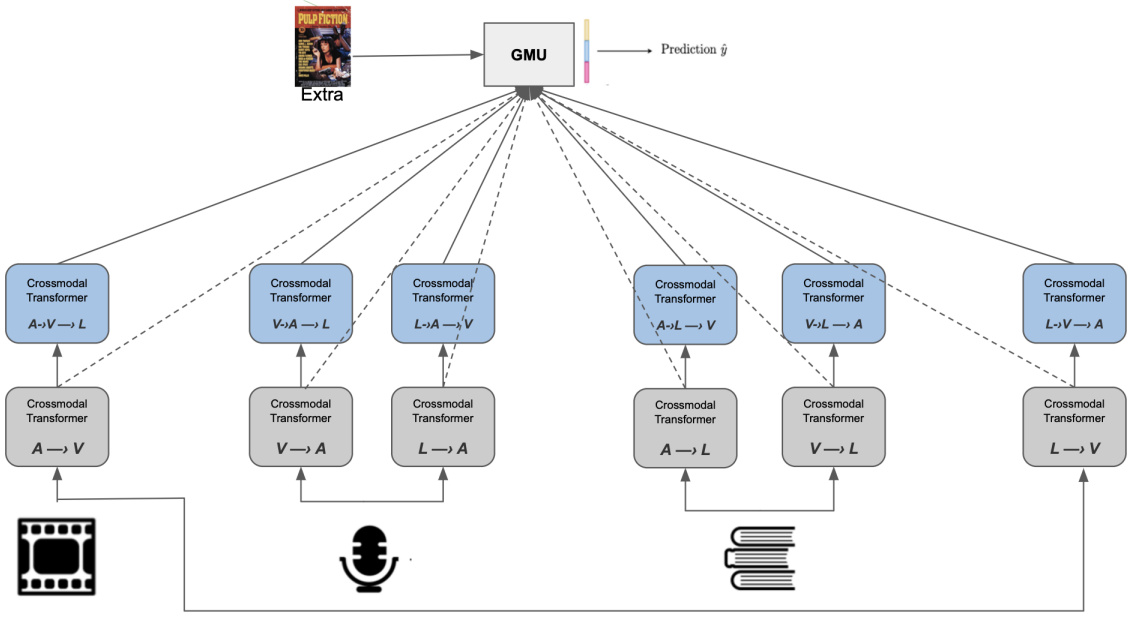



**Figure 6.6:** Proposed **BPMulT-no-parallel** architecture for a multimodal classification task passing all transformer encoders (biprojections and single projections) for ablation study. The blue part corresponds to biprojection modules. Gray modules are the first crossmodal transformers.

Another observation is that a biprojection helps single projections increase activation if it is low. For example, we can see this in the biprojection of bars 7, 8, and 12, which correspond to bars 10, 1, and 2, respectively. But, when the single activation is high, a biprojection of that module is prejudicial. Generally, we confirm that a biprojection to the Language space benefits all modalities. A biprojection to the Audio space is beneficial to the Video modality, and a projection to the video modality is always a bad idea for this particular set. This experiment can conclude that the biprojections are rescuing relevant information for classification that a single projection can not obtain.

### 6.3.4 Other Minor Ablation Modifications

To obtain a complete ablation study, we need to do other minor modifications to remove each proposed part of the architecture. We describe each experiment with a brief discussion and its results.

#### No-Residual Connections

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | BPMulT-no-RC (ours) | 81.2±0.3 | 77.7±0.5 | 87.1±0.3 |
| | BPMulT-no-FGMU (ours) | 79.9±0.25 | 76.54±0.3 | 86.2±0.4 |
| | BPMulT-no-FGMU-nor-RC (ours) | 80.5±0.5 | 76.9±0.5 | 86.4±0.3 |
| | BPMulT-all-transformers (ours) | 79.8±0.4 | 76.4±0.3 | 86.1±0.5 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.5:** Comparison of our BPMulT model without using a residual connection from the first crossmodal transformer to the biprojection. It is to obtain a measurement of relevance. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP), and sample (sAP) averaged.

We show in Section 6.3.2 that not using FGMUs nor residual connections directly affects the classification performance. This section investigates if the residual links from the first crossmodal projections to the biprojection are helpful for the model. The architecture without these connections is in Figure 6.7.

Then, we note in Table 6.5 that the relevance of the residual connections is minimal for the classification task. The best model is only 0.2% above in all metrics. This residual sum does not take much computational cost; hence, we decided to keep it in the final model.

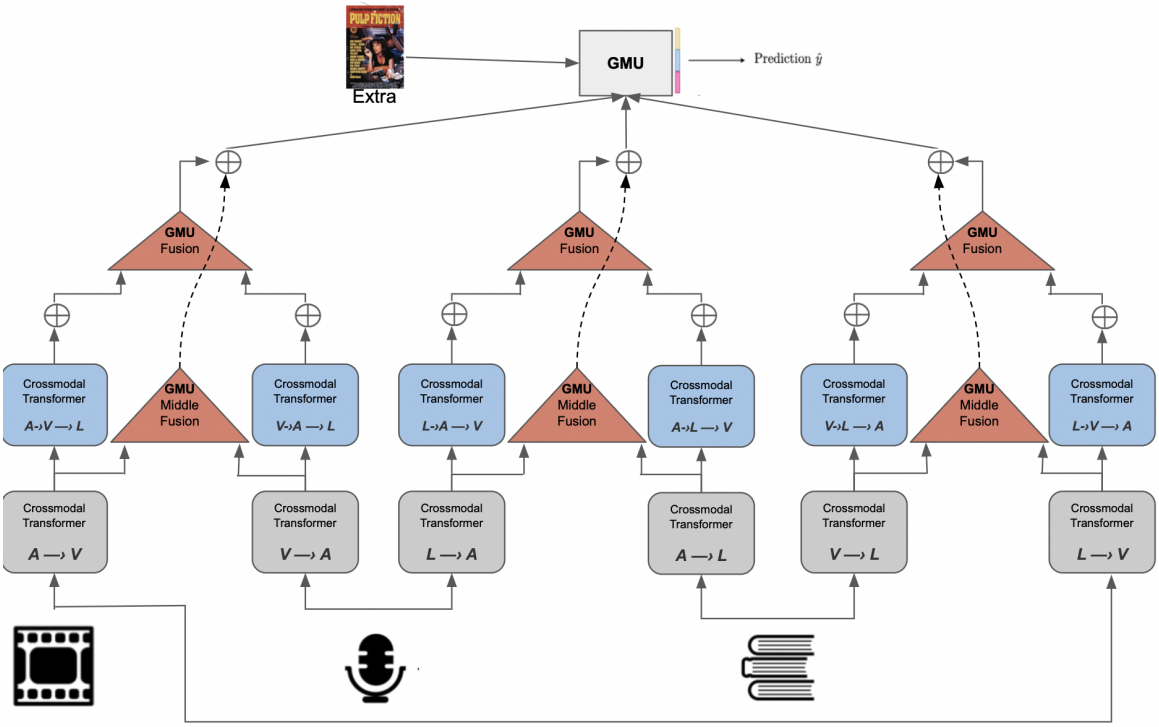



**Figure 6.7:** Proposed **BPMulT-no-parallel** architecture for a multimodal classification task without using a residual connection from the first crossmodal attention to the biprojection for ablation study.

### No-FGMU in the Middle

Similar to the previous analysis, in this section, we investigate if the Fusion GMU in the middle is relevant for the model. The Middle FGMU fuses the first crossmodal projections to add them to the last biprojection. The architecture without these Middle FGMU is in Figure 6.8.

In Table 6.6 the FGMU at the middle is essential for the classification task. The results have concordance with the model without FGMU at all and also without residual connections. These results give us and specific order of relevance. The worst result is when there are no FGMU in the model, but without the Middle FGMU is not

too bad. Then, without residual connections, results are good enough and remove all FGMU, and these connections metrics are lower than just removing Middle FGMU or the connections. Hence, this experiment shows that the most important component (apart from crossmodal transformers) is the FGMU at the top, then the Middle FGMU, and the last is the residual connections.

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | BPMulT-no-FGMU-Middle (ours) | 80.95±0.3 | 77.6±0.5 | 86.9±0.4 |
| | BPMulT-no-RC (ours) | 81.2±0.3 | 77.7±0.5 | 87.1±0.3 |
| | BPMulT-no-FGMU (ours) | 79.9±0.25 | 76.54±0.3 | 86.2±0.4 |
| | BPMulT-no-FGMU-nor-RC (ours) | 80.5±0.5 | 76.9±0.5 | 86.4±0.3 |
| | BPMulT-all-transformers (ours) | 79.8±0.4 | 76.4±0.3 | 86.1±0.5 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.6:** Comparison of our BPMulT model without fusing the first crossmodal transformers to obtain a measurement of relevance. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP), and sample (sAP) averaged.

### Transformer Attention (concat)

We use the FGMU to fuse information between modalities to interpret this dynamic fusion and save disk memory space. On the other hand, the original Multimodal Transformer uses the same crossmodal transformers but fuses the main information using another transformer and concatenates both modality vectors. This process is tremendously expensive, especially when the modality vectors are large. We designed the following ablation experiment to compare if the fusion with a self-attention transformer is better than our proposed fusion. The architecture is substituting the FGMU at the top with a transformer that receives a concatenated vector. The model is in Figure 6.9.

In Table 6.7, we can compare the result obtained by the model with a self-attention transformer at the top. We note a significant decrease in performance in all metrics. This result shows that the FGMU is the best module for fusing information within this model. In addition, with an FGMU, we are saving storage space, and it has a dynamic flow of information that is useful for human interpretation.

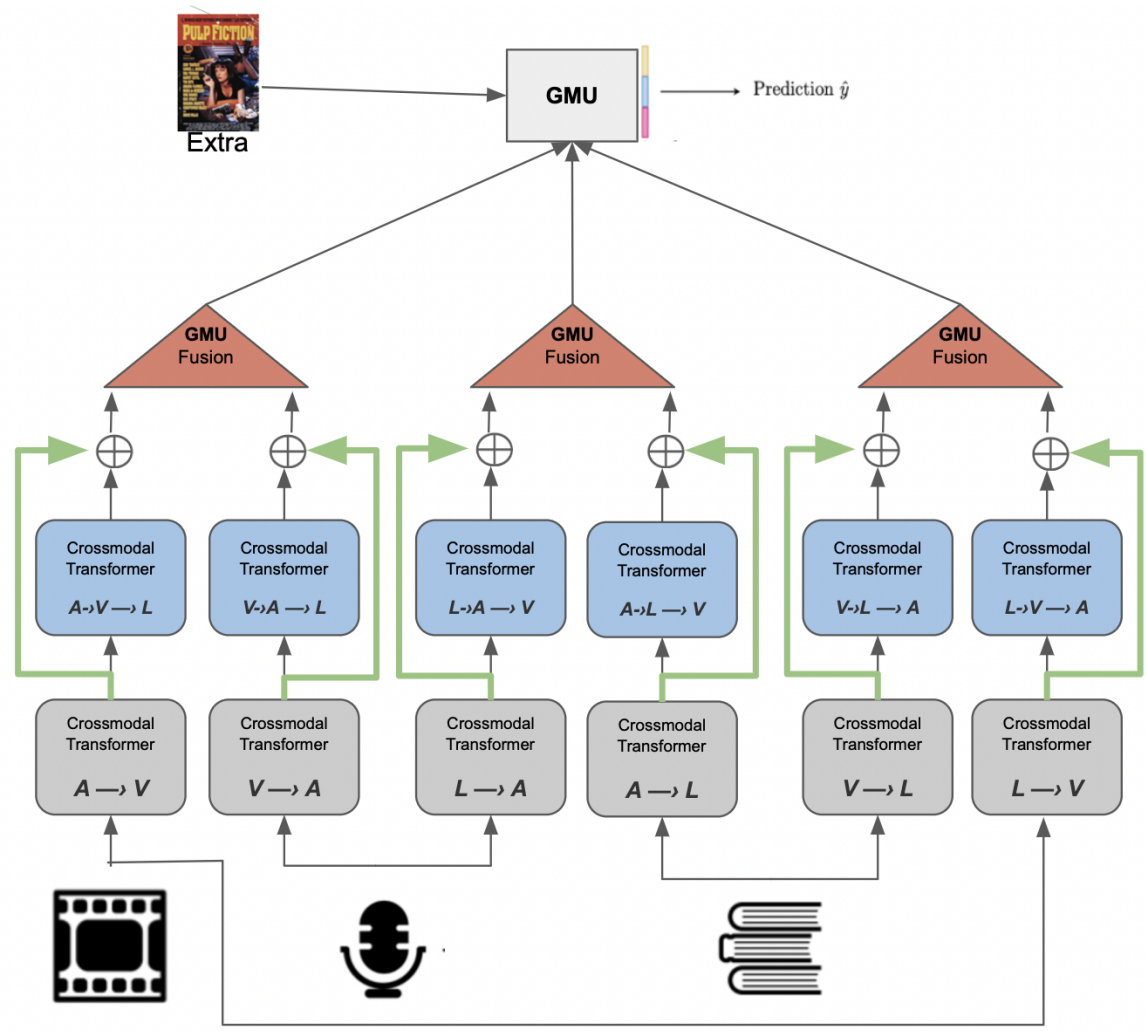



**Figure 6.8:** Proposed **BPMulT-no-parallel** architecture for a multimodal classification task without fusing the first crossmodal attention with a FGMU for ablation study.

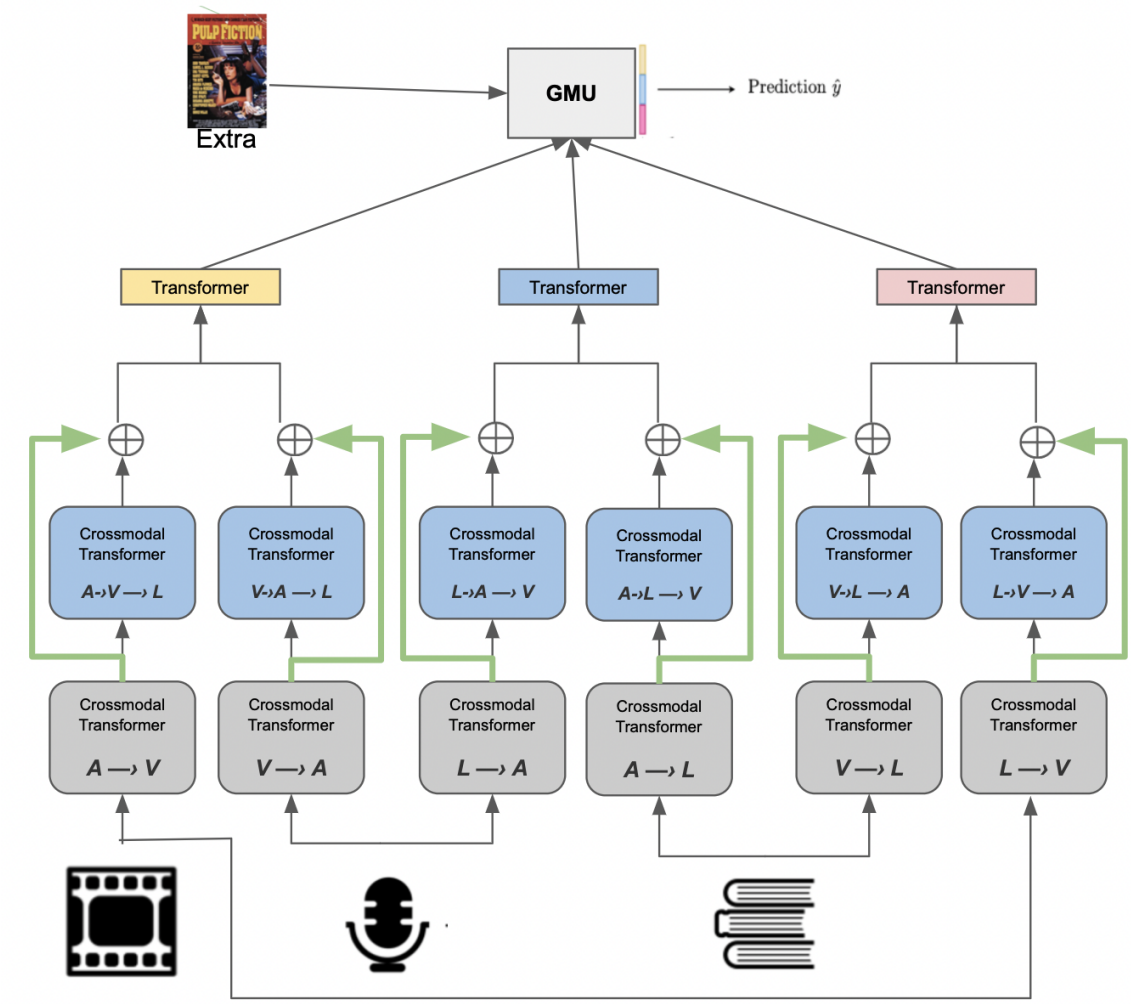



**Figure 6.9:** Proposed **BPMulT-no-parallel** architecture for a multimodal classification task substituting the FGMU at the top by a transformer for ablation study.

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | BPMulT-with-transformer (ours) | 79.97±0.1 | 76.6±0.3 | 86.2±0.3 |
| | BPMulT-no-FGMU-Middle (ours) | 80.95±0.3 | 77.6±0.5 | 86.9±0.4 |
| | BPMulT-no-RC (ours) | 81.2±0.3 | 77.7±0.5 | 87.1±0.3 |
| | BPMulT-no-FGMU (ours) | 79.9±0.25 | 76.54±0.3 | 86.2±0.4 |
| | BPMulT-no-FGMU-nor-RC (ours) | 80.5±0.5 | 76.9±0.5 | 86.4±0.3 |
| | BPMulT-all-transformers (ours) | 79.8±0.4 | 76.4±0.3 | 86.1±0.5 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.7:** Comparison of our BPMulT model substituting the FGMU for a transformer to obtain a measurement of relevance. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP), and sample (sAP) averaged.

### Cross-Attention Modification (Translating)

For this experiment, we investigate the crossmodal attention of the biprojection at its nucleus. This crossmodal transformer was proposed in Tsai et al. (2019), and we use it with other modalities data in a different order to obtain better sequence patterns. In Figure 6.10, we can observe how we are doing the biprojection step replicating the crossmodal transformer for the second attention.

Our intuition was to follow the idea of the original transformer, which was first used as a Neural Machine Translation. We believe that we can handle modalities as languages. In other words, we want to translate the sequence in "language" called **text** to another language called **Video**.

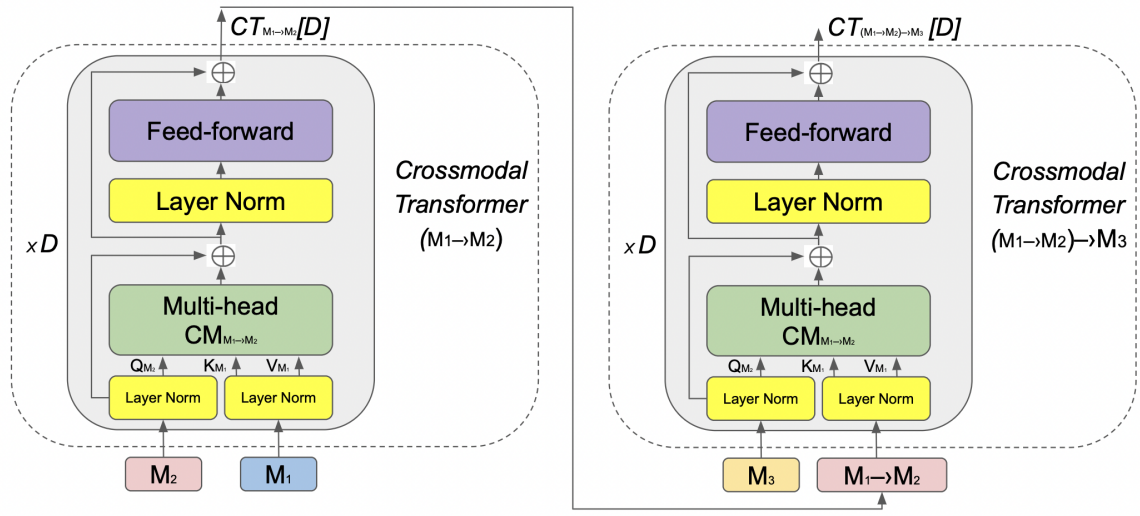



**Figure 6.10:** Proposed **Biprojection** mechanism composed by two crossmodal transformers. It can be viewed as an encoder-decoder between modalities.

Following Vaswani et al. (2017), we propose (for ablation experiment) an attention inspired in a translation. The modification only replicates the "decoder" part of the transformer in our biprojection. It is just adding a self-attention module followed by a normalization layer. We can observe the biprojection as a decoder module in Figure 6.11.

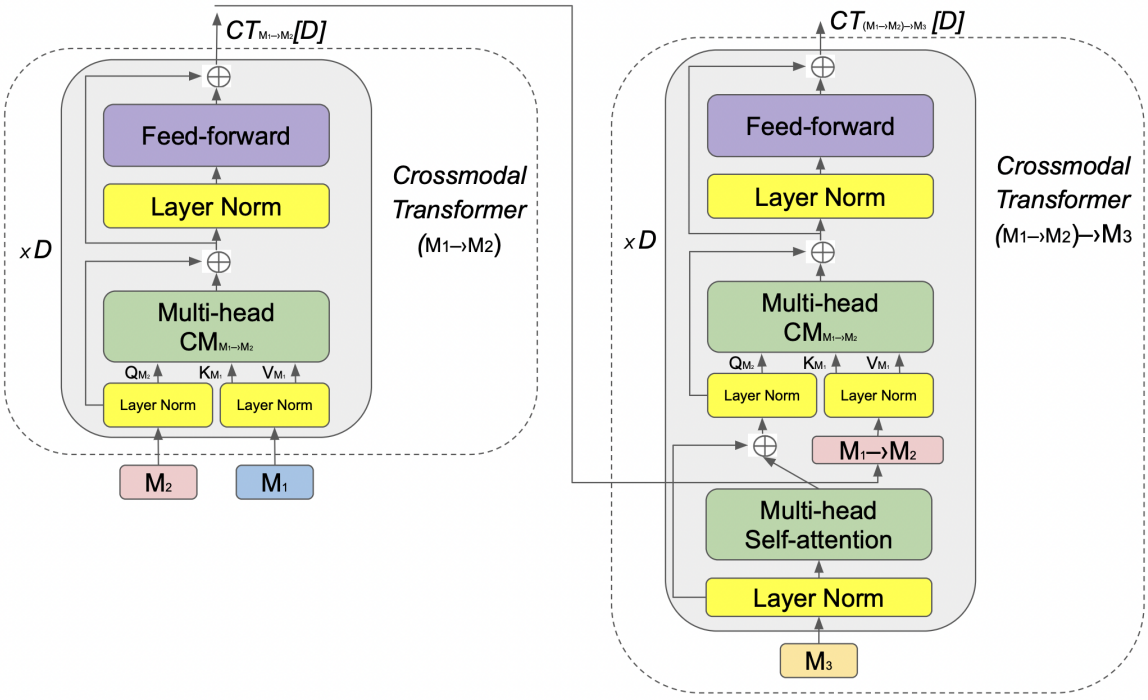



**Figure 6.11:** Proposed **Biprojection** mechanism with a modification for translating. It comprises a crossmodal transformer as an encoder and another crossmodal transformer with a self-attention and a normalization layer as a decoder.

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | BPMulT-translating (ours) | 81.2±0.3 | 77.8±0.5 | 87.0±0.3 |
| | BPMulT-with-transformer (ours) | 79.97±0.1 | 76.6±0.3 | 86.2±0.3 |
| | BPMulT-no-FGMU-Middle (ours) | 80.95±0.3 | 77.6±0.5 | 86.9±0.4 |
| | BPMulT-no-RC (ours) | 81.2±0.3 | 77.7±0.5 | 87.1±0.3 |
| | BPMulT-no-FGMU (ours) | 79.9±0.25 | 76.54±0.3 | 86.2±0.4 |
| | BPMulT-no-FGMU-nor-RC (ours) | 80.5±0.5 | 76.9±0.5 | 86.4±0.3 |
| | BPMulT-all-transformers (ours) | 79.8±0.4 | 76.4±0.3 | 86.1±0.5 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.8:** Comparison of our BPMulT model with translating attention to obtain a measurement of relevance. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP), and sample (sAP) averaged.

The results of this translating mechanism are given in Table 6.8. This translating mechanism achieves similar results to the normal biprojection but slightly lower. It has an equal relevance that was using the model without residual connections. In conclusion, this translating attention is similar in performance but is more expensive

than the regular biprojection attention. Then, it shows that the proposed BPMulT is the best model configuration of their components.

## 6.4 Relevance of Modalities

This section aims to understand which modalities the model considers for predicting. We are analyzing the GMU module activations on the test set and the FGMU activations to see the dynamic flow of information in the model. In addition, we complement the analysis with a TSNE study. This TSNE study shows the vectors' data colored by their genre and plots them in a two-dimensional space by its predicted label in the test case. With this visualization, we can see the data grouped by genre (same color) if the model is doing a correct classification.

### 6.4.1 GMU Activations

To compare our model, we first analyze the activation that the MulT-GMU of Rodríguez-Bribiesca et al. (2021) has. On the left of Figure 6.12 we can observe the modalities feature activation of the MulT-GMU, the most activated modality is text followed by the poster. It shows that text has more relevance for classification than audio or video modalities. The video modality is, on average activated 5% less than the Text modality, and the audio is 10% less activated than text.

Note that the MulT-GMU has various activations for each modality. Remarkably, the text modality is notably less activated with our proposed model. Poster features are still activated for more than 80%. Video and audio modalities are now activated with a big difference. This section aims to find why this is happening and whether this modality is enriched with better sequence patterns or whether they have information that the BPMulT uses, and the MulT-GMU does not.

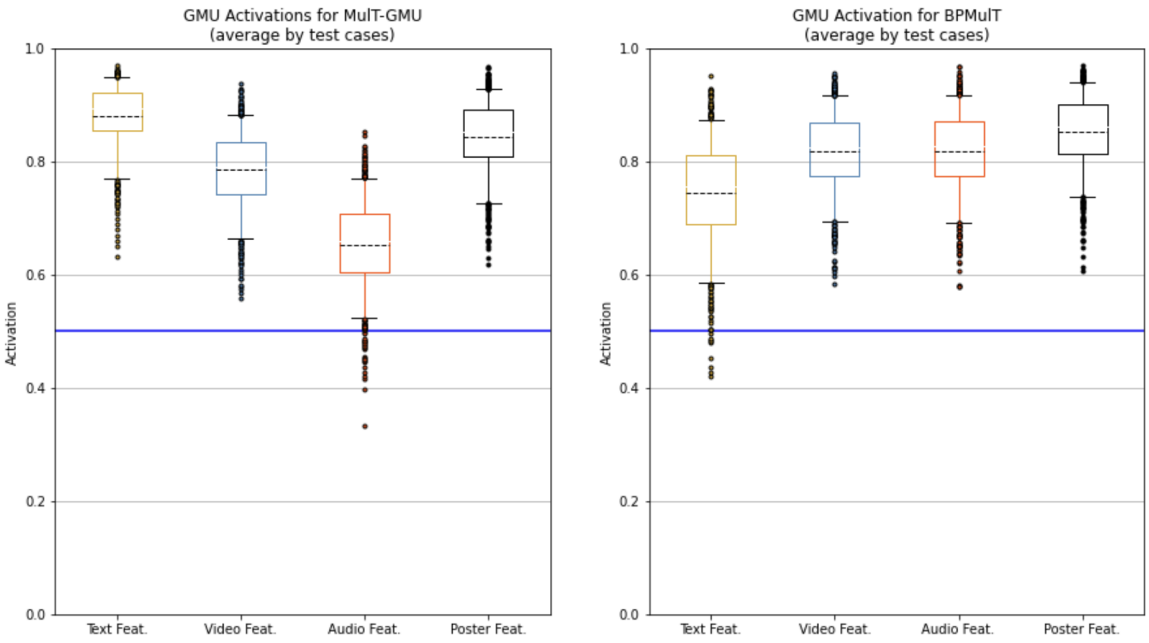



**Figure 6.12:** Comparison of features activation by the GMU for dynamic fusion for the Moviescope dataset (Cascante-Bonilla et al., 2019). On (left), we can observe the activation given by the model MulT-GMU of Rodríguez-Bribiesca et al. (2021), and on the (right), we show the activation of our proposed model BPMulT.

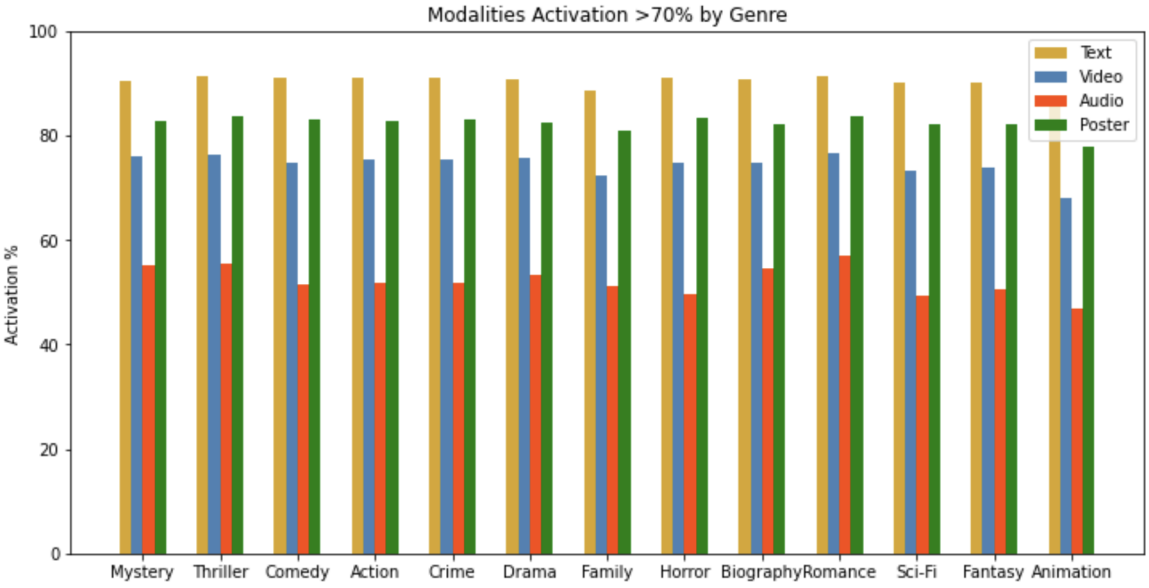



**Figure 6.13:** Label activation of the GMU for dynamic fusion for the Moviescope dataset (Cascante-Bonilla et al. (2019)) given by the model MulT-GMU of Rodríguez-Bribiesca et al. (2021) by genre.

In Figure 6.13, we can observe the GMU activations for each modality but sepa-

rated by genres. We note that these activations are not the same for all genres, but they are similar.

The same case we have for the BPMulT activations is in Figure 6.14. The average activation is similar for all the genres, and the less activated modality is text. Video and audio are almost equal activated, and the poster is more used for the classification.

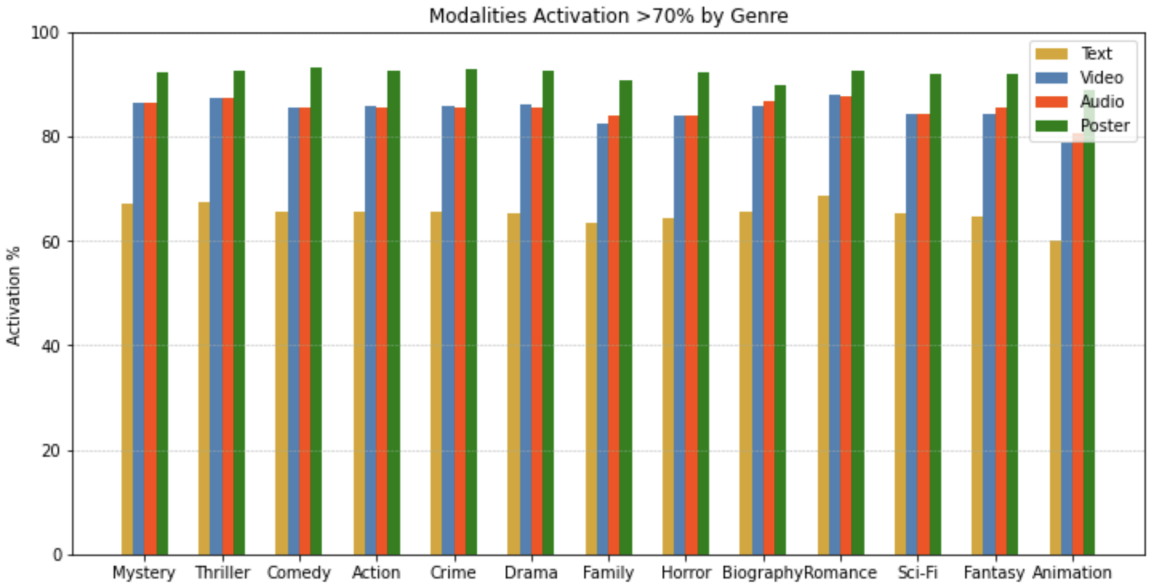



**Figure 6.14:** Label activation of the GMU for dynamic fusion for the Moviescope dataset (Cascante-Bonilla et al. (2019)) given by our proposed model BPMulT by genre.

### 6.4.2 TSNE Study of GMU Activations

To complement this analysis, we do a TSNE study of the GMU's activation for each modality in the BPMulT model. If one modality is more relevant for the model than the others, it is more comprehensive and may have a better clusterization with this study. In Figure 6.15 we can observe the activation of the text modality group labels.

Furthermore, there are clusters in the space. We have the Comedy (blue) and Drama (red) data on one side. Between them, we have some Romance (dark green) movies. On the other side, we can observe ocher and yellow clusters corresponding to Fantasy and Animation genres. Next to them is a pink cluster of Science-Fiction movies, and in the green set, we have Horror movies close to the black cluster corresponding to the Crime genre. Note that these mentioned movies are related, and the

clusters make sense for us as human classifiers. It is a good signal that our model is doing well for classification.

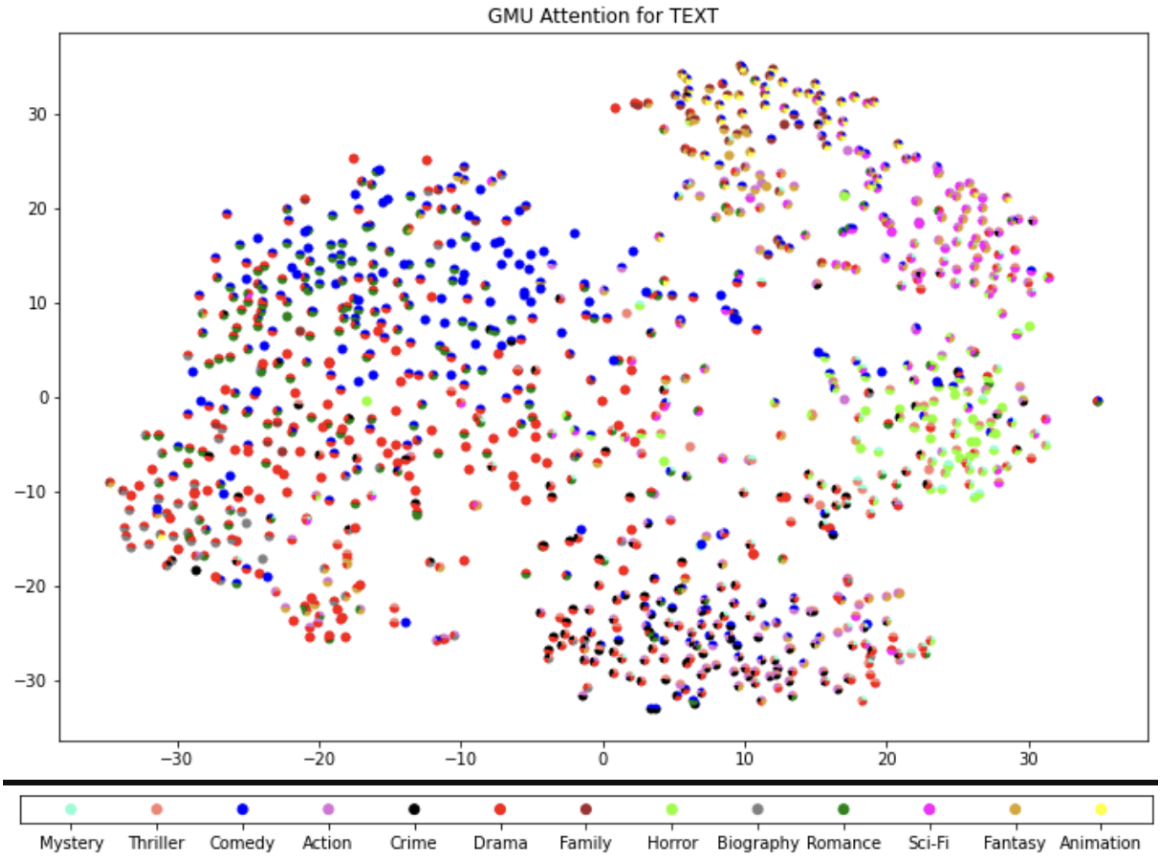



**Figure 6.15:** TSNE study of the output of the last GMU for a dynamic fusion of modalities in the BPMulT model. Each color corresponds to a genre, and data is multi-labeled. We are visualizing the activations corresponding to the **text** part.

Similarly, the activations of video, audio and poster modalities in Figures A.1, A.2, A.3 in the Appendix A are well clustered being the audio and video the modalities with the best performance in activation and with clear clusters. If there is a perfect clusterization of the data, the model performs better since it knows how to classify perfectly by genre.

### 6.4.3 Comparison TSNE Studies of Naive vs. BPMulT

To contrast our model with this study, we design a naive experiment that classifies the movie genre with a Support Vector Machine (SVM) method. With this model,

we are looking for a difference in the TSNE study between our classification model's performance and a traditional NLP multimodal model.

This naive model takes a simple representation of each modality to feed the model. We have selected a Bag of Words (BoW) representation with a TF-IDF weighting for the text sequence. This representation is used with a vocabulary length of $10,000$, and a sequence of a maximum of 512 tokens. We take the concatenation of all of these four modality features to feed the SVM classifier. We take the average of the 200 video frames for the video sequence features and, similarly, for the audio sequence. We only have the features for the poster since there is no sequence. The results of this model are in Table 6.9 followed by the TSNE study of the outputs in Figure 6.16.

| Modalities | Model | $\mu$AP | $m$AP | $s$AP |
|---|---|---|---|---|
| TVAP | Naive SVM | 62.0 | 59.0 | 63.0 |
| | **BPMulT-no-parallel (ours)** | **81.4±0.3** | **78.0±0.5** | **87.2±0.4** |
| | **BPMulT (ours)** | **81.7±0.2** | **78.1±0.1** | **92.4±0.1** |

**Table 6.9:** Comparison of our BPMulT model and a traditional NLP multimodal model in order to contrast classification performance. Metrics reported corresponding to average precision, micro ($\mu$AP), macro (mAP), and sample (sAP) averaged.

We see in 6.16 a massive difference between the classification with a naive method and a TSNE graph of our BPMulT model in the final prediction, as shown in Figure 6.17. With this naive model, the cloud of points is dense, and we can not see any physically separated cluster. On the other hand, we can distinguish a color separation, but it is hard in some regions, for example, at the cloud's right with labels of almost every genre.

### 6.4.4 Understanding the FGMU Activation Flow

Now, we are interested in analyzing the flow of information in the FGMUs to get the actual contribution of each modality. In Figure 6.18, we compare the GMU activations when we fuse the first projections and the biprojections. The first row corresponds to all single projections from one modality to another. The second row has the activations of the biprojections. We can understand the information's flow in

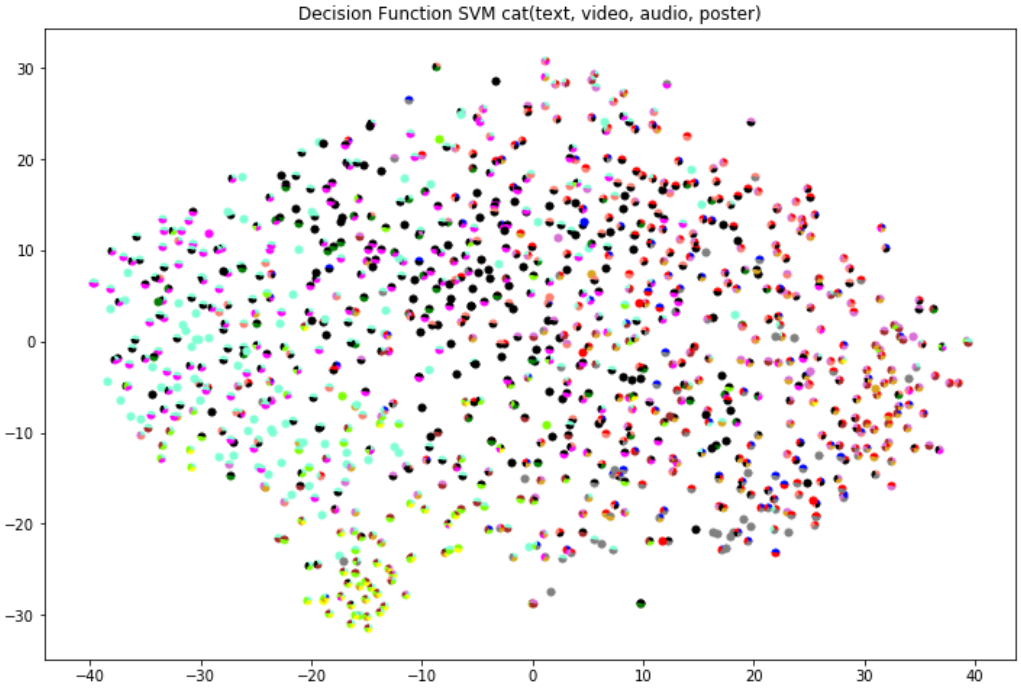



**Figure 6.16:** TSNE study of the output of the naive multimodal model. Each color corresponds to a genre, and data is multi-labeled.

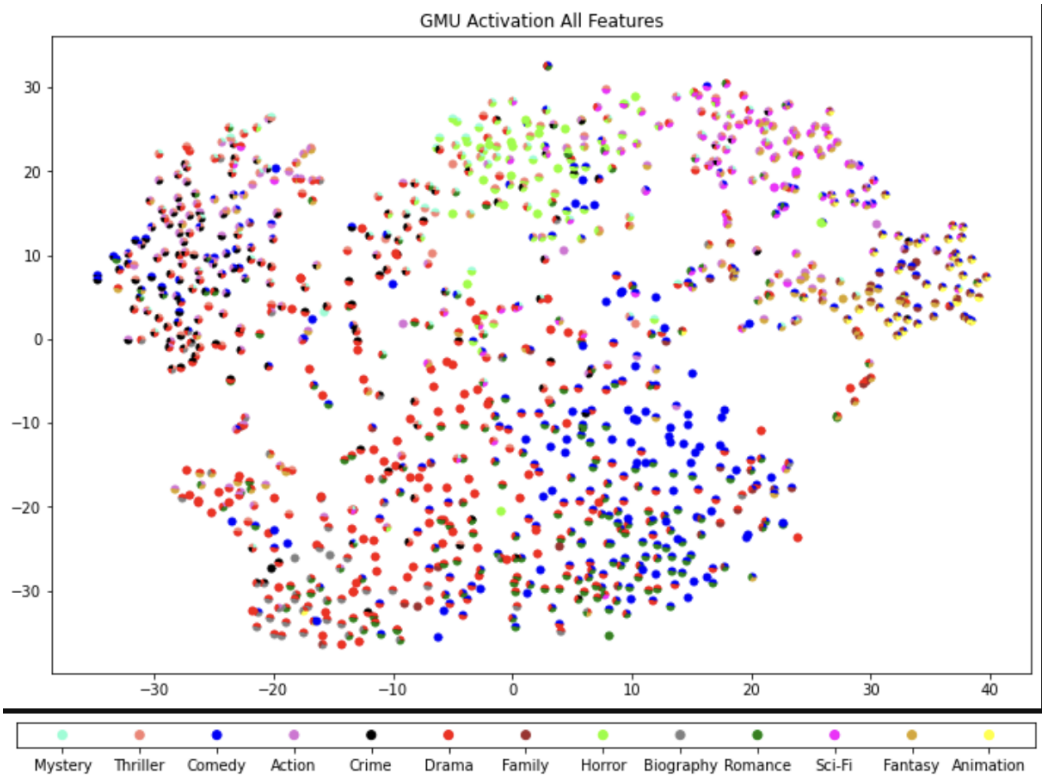



**Figure 6.17:** TSNE study of the output of our proposed BPMulT model. Each color corresponds to a genre, and data is multi-labeled.

the model considering the graphs ordered by columns.

In the first box plot, we have that the features of a projection from Audio to Video are more activated than Video to Audio. Then, below the first graph, we have that the biprojection features of (A–›V)—›L are used more than the features of (V–›A)—›L, which has accordance with the relevance in the first graph. Hence, features of (A–›V) and (A–›V)—›L are mainly used to represent the Text modality. Figure 6.12 shows that the Text modality is not as well activated as the Audio and Video modality in the final GMU. It could be a consequence of using a large amount of Video information.

For the Video representation, note that projections and biprojections are equally used, which could be the reason for a high activation of this modality. We will discuss this case with other studies to confirm this idea.

Finally, the Audio representation is compounded by mostly Text and (V–›L)—›A features. The Audio representation also has a high activation. As a first conclusion, the features with more relevance to the classifications are from (V–›L)—›A and V–›L coming from the Audio modality. For the Video modality, the information comes from A—›L, (A–›L)—›V, L—›A, and (L–›A)—›V.

### 6.4.5 TSNE Study of the Dynamic FGMU Activation Flow

In order to investigate why the FGMU modules are taking the information flow as we have seen in Figure 6.18, we analyze the Crossmodal Transformers involved with a TSNE study. We show below an example analysis of the text branch modality representation. The text branch refers to the BPMulT part used to represent the text modality, and we can see this branch in Figure 6.19. All the other TSNE study graphics for the other modality branches are in Appendix A.

In Figure 6.20, we see that the data distribution is not as accurate as we desired. We can not identify any group with this TSNE graph. In contrast, Figure 6.21 shows a great change in data distribution. We can easily identify the cluster of Horror, Sci-Fi, Drama, or Comedy movies. Hence, this biprojection should be strongly used to represent the text modality.

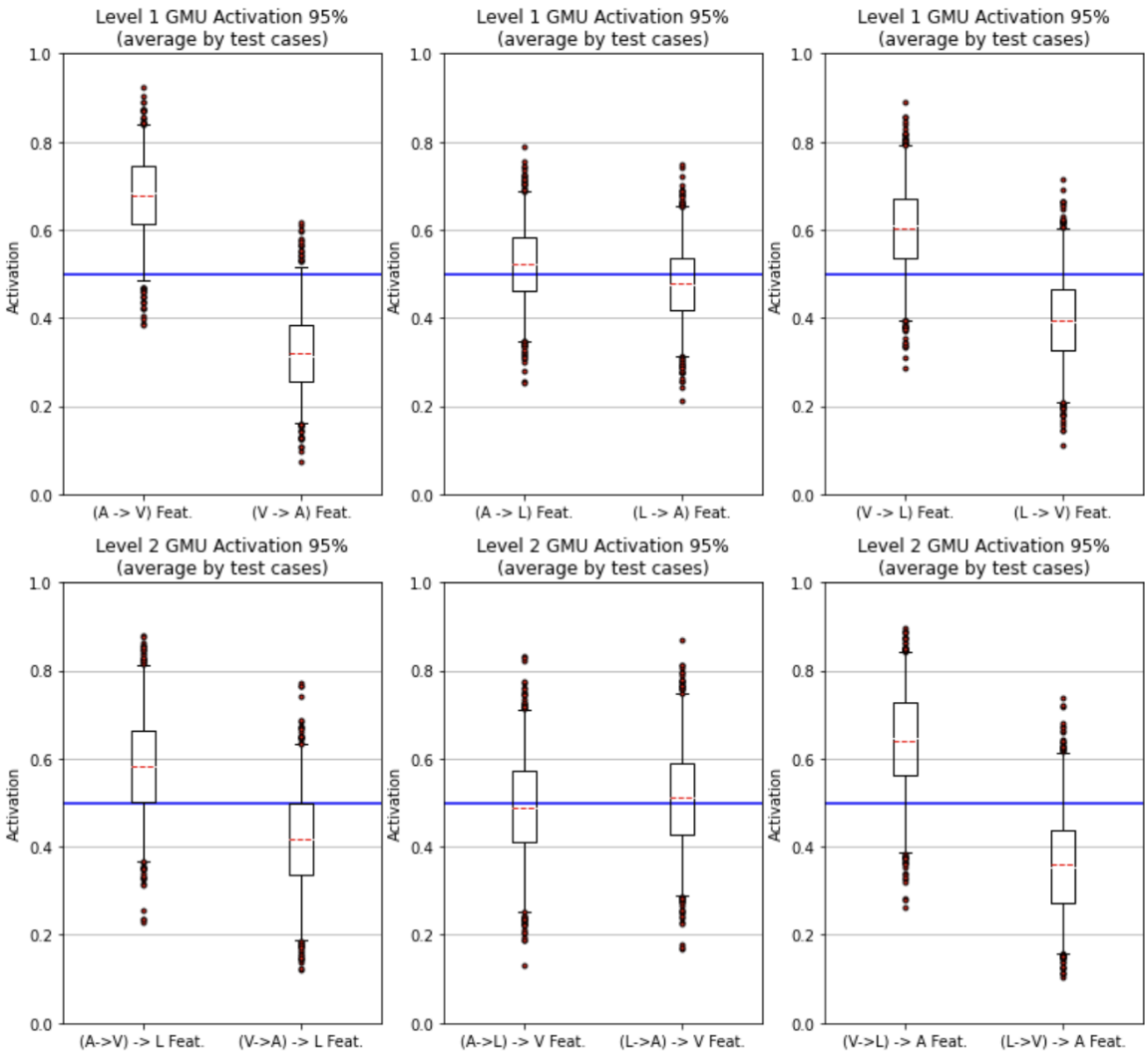



**Figure 6.18:** Comparison of features activation by each FGMU (Top and Middle) for dynamic fusion for the Moviescope dataset (Cascante-Bonilla et al., 2019). In the **first** column we can observe the activation given by the BPMulT to get the **text** modality representation, in the **second** column the activation to get the **video** modality representation, and in the **third** column the activation of the **audio** representation. The first row corresponds to the Middle FGMU and the second row to the Top GMU.

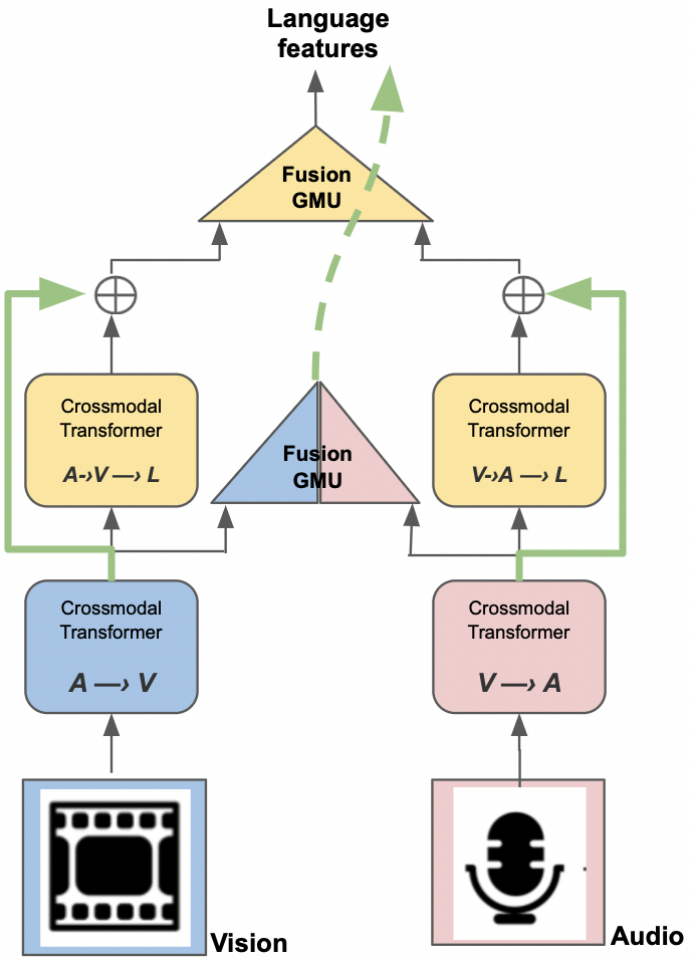



**Figure 6.19:** Text branch of the Biprojection Multimodal Transformer (BPMult) model. It consists of two opposite projections ((A—›V) and (V—›A)) and their respective biprojections to the text space.

On the other branch of the Text representation, we have the projections V—›A and (V–›A)—›L. The corresponding TSNE study of each projection is in Figure 6.22 and 6.23. We can observe in these figures something similar to the other branch, i.e., the projection V—›A is not clearly clustered but biprojecting this to the Text space makes the distribution better to classify.

Remembering that in our proposed model, the Text modality is not well activated (Figure 6.12), and to represent this modality is used with a majority of the A—›V and (V–›A)—›L features (Figure 6.18). We conclude that the activation of the Text modality is disadvantaged by using a large amount of A—›V features.

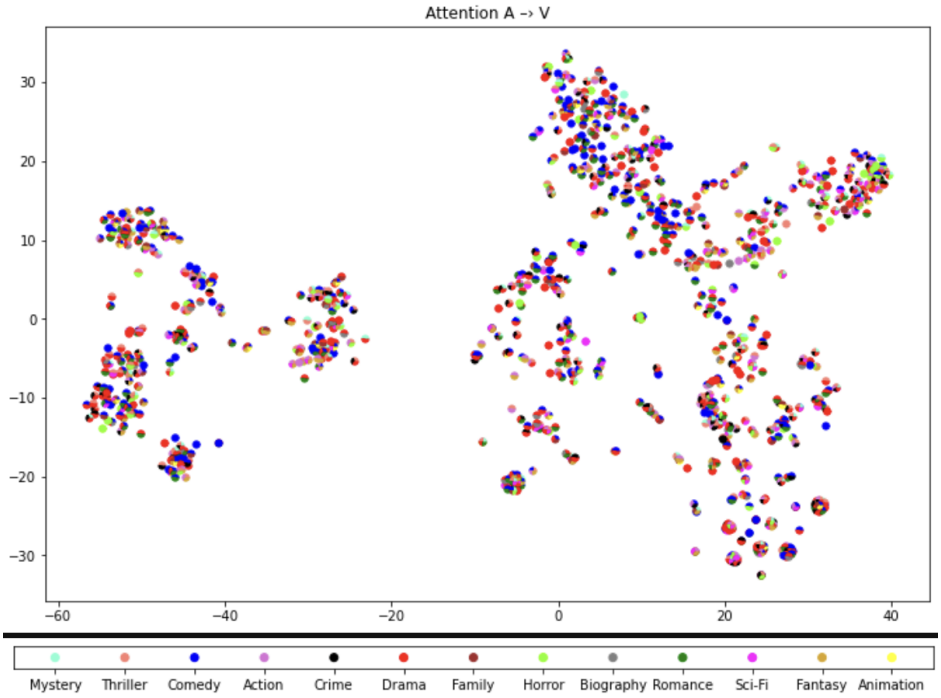



**Figure 6.20:** TSNE study of the output of the crossmodal transformer (A—›V) in the BP-MulT model. Each color corresponds to a genre, and data is multi-labeled. It is considered a **bad** clusterization.

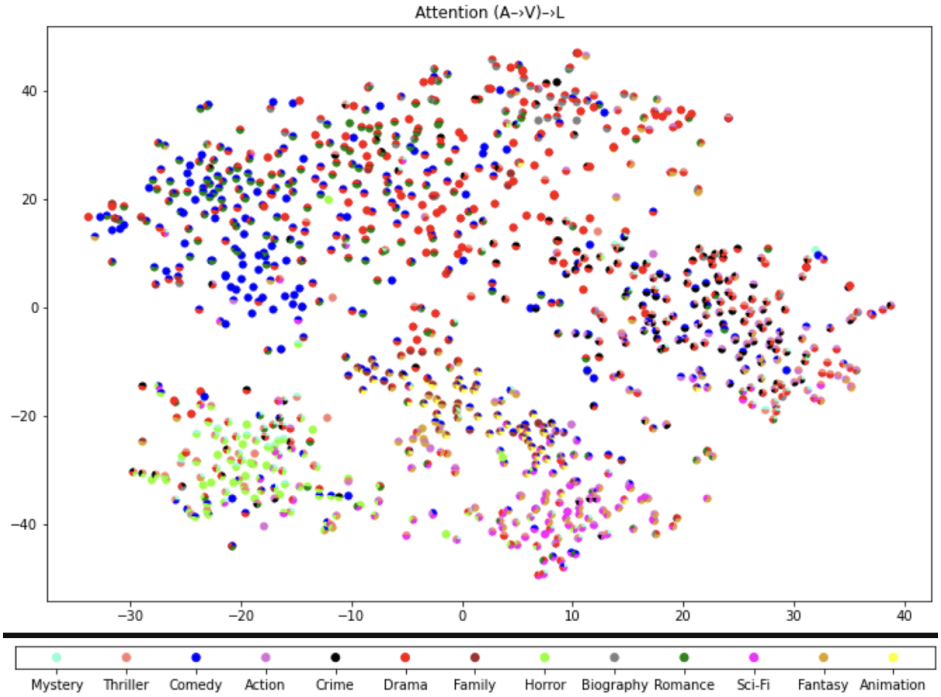



**Figure 6.21:** TSNE study of the output of the crossmodal transformer with biprojection (A—›V)—›L in the BPMulT model. Each color corresponds to a genre, and data is multi-labeled. It is considered a **good** clusterization.

Doing an analogous analysis for Audio and Video modalities, we found that V—›L and (V–›L)—›A are well clustered, and it is because the Audio modality has a high activation. In the case of Video, the A—›L and L—›A are well grouped (the biprojections to the Video modality not). Still, we have the same activation for the four modules, so the modality of Video is also well activated.

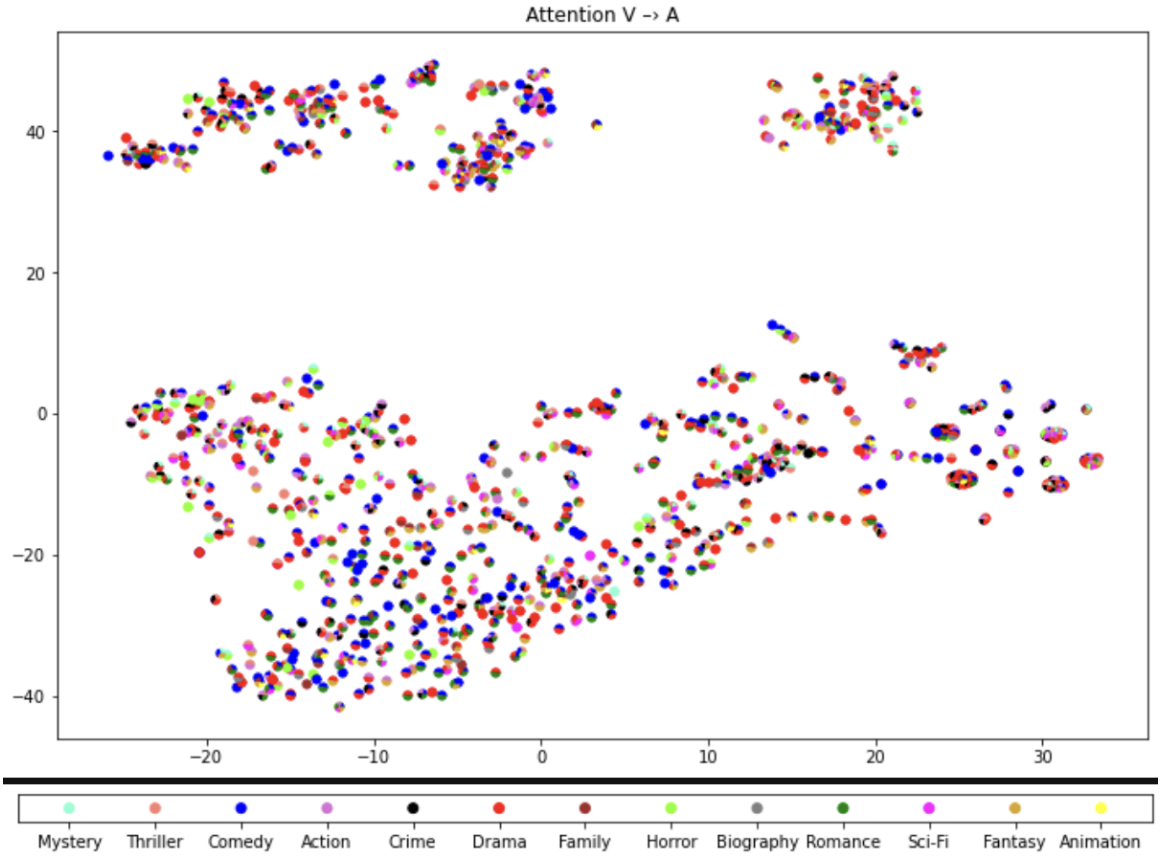



**Figure 6.22:** TSNE study of the output of the crossmodal transformer (V—›A) in the BP-MulT model. Each color corresponds to a genre, and data is multi-labeled. It is considered a **bad** clusterization.

With this analysis, we obtained the exact flow of information. It showed us why modalities like Video and Audio are enriched and the text modality's low activation. However, with the residual connections between levels, all types of crossmodal transformers are considered for the final prediction. They help to find better patterns and to improve the overall score.

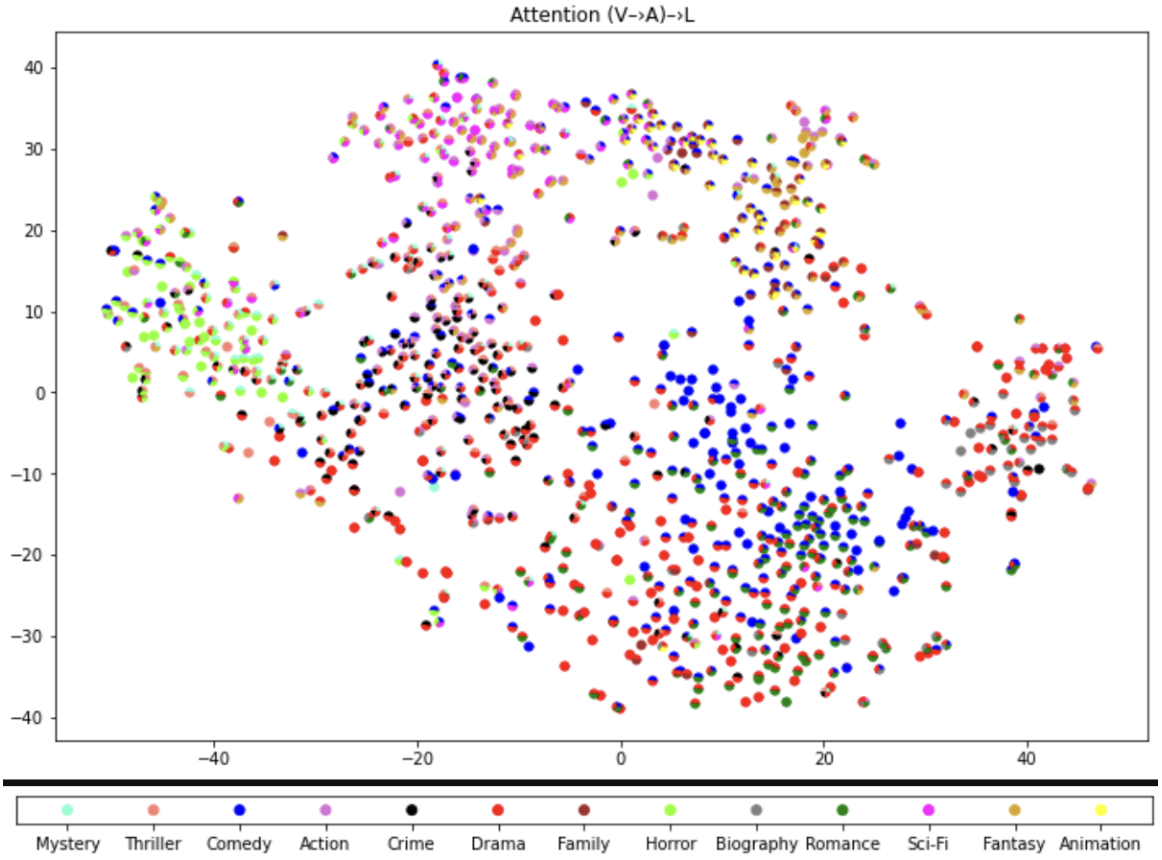



**Figure 6.23:** TSNE study of the output of the crossmodal transformer with biprojection (V—›A)—›L in the BPMulT model. Each color corresponds to a genre, and data is multi-labeled. It is considered a **good** clusterization.

## 6.5 Other Datasets

This section describes the procedures and results to evaluate our proposed model considering the other dataset baselines in Chapter 5. We first consider the data in a similar dataset that tries to classify genres in movies. This dataset corresponds to the MM-IMDb presented for the first time in Arevalo et al. (2017). The problem is that this dataset contains just a synopsis (in Text modality) and its movie poster. BPMulT considers three sequential modalities for its proper functioning. Hence, we take three different Text representations: BERT, GloVe, and BoW, taking these as three different modalities in the model. In Section 6.5.2, we test our proposal in another completely different dataset, the IEMOCAP, where the task is to classify **emotions** in conversation considering the Text, Video, and Audio modalities. Finally, in Section 6.5.3 we prove the BPMulT model in a recent version of the CMU-MOSEI proposed in Dai, Cahyawijaya, Liu, and Fung (2021) to classify **emotions** for unaligned sequences of Text, Audio and Video modalities. We achieve competitive results in metrics, and in most cases, we overpass the SOTA metrics.

### 6.5.1 BPMulT in the MM-IMDb Dataset

We have mentioned that MM-IMDB has just two modalities for information extraction, text (synopsis) and its poster. Since BPMulT works finding patterns between sequences. We proposed to represent the synopsis with various text embedding. Our first approach is to use the BERT representation because we use the transformer architecture. The other representations selected are the GloVe embedding which is a widely used text embedding, and the third one is just simple BoW taking as vocabulary the ten thousand most frequent words used in the training set. The poster will be fused with a GMU at the last part of the BPMulT model since it is not a piece of sequential information.

| Model | $\mu$-F1 | $m$-F1 | $W$-F1 | $s$-F1 |
|---|---|---|---|---|
| GMU | 63.0 | 51.4 | 61.7 | 63 |
| MulT-GMU | 66.3±.6 | 61.1±.6 | - | - |
| ConcatBERT | 65.9±.2 | 60.5±.3 | - | - |
| MMBT | 66.8±.1 | **61.6±.2** | - | - |
| **BPMulT-no-parallel (ours)** | **68.9±.1** | 58.7±.3 | **68.8±.1** | **69.4±.2** |
| **BPMulT (ours)** | **68.9±.1** | 58.7±.3 | **68.8±.1** | **69.4±.2** |

**Table 6.10:** Comparison of our BPMulT model in the MM-IMDb dataset where the model MMBT is the current SOTA model. Metris reported are F1 scores micro ($\mu$), macro ($m$), weighted ($W$), and sample ($s$) averaged.

Table 6.10 shows the results for the metrics considered in past publications. The first model proposed is the GMU which was overpassed before by the MMBT model. We performed five random training seeds to get the mean and standard deviation metrics for the BPMulT model. We achieved better metrics in $\mu$-F1, $W$-F1, and $s$-F1 than reported for the previous SOTA models.

### 6.5.2 BPMulT in the IEMOCAP Dataset

IEMOCAP is one of the most used datasets to test multimodal architectures since it has three modalities for information extraction. It has features from Text (GloVe), Audio, and Video. With this dataset, BPMulT works directly to find patterns between sequences. We do not have any non-sequential modality here. Then, we fuse all three modalities with a GMU at the last part of the BPMulT. IEMOCAP provides an aligned sequence of length **twenty** since the text modality is sectioned in twenty parts. We consider this alignment in Table 6.11 for our first approach, but it is unnecessary. In the Table 6.12 we can see the classifications result for unaligned sequences.

Table 6.10 shows the results for the metrics considered in Dai et al. (2020). We took this publication as the SOTA model because it considers that the F1 metric is inappropriate for unbalanced datasets. To avoid this problem, they propose using the AUC score. We performed random seed training for the BPMulT model. We achieved a better AUC metric and a competitive score in the Accuracy metric.

IEMOCAP Aligned

| Model | Neutral | | Happy | | Sad | | Angry | | Average | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Acc | AUC | Acc | AUC | Acc | AUC | Acc | AUC | Acc | AUC |
| MulT | 71.0 | 77.2 | 83.5 | 71.2 | 85.0 | 89.3 | 85.5 | 92.4 | 81.3 | 82.5 |
| ModTrans-MMEmoRe | 71.1 | 76.7 | 85.0 | **74.2** | **86.6** | 88.4 | **88.1** | **93.2** | **82.7** | 83.1 |
| **BPMulT-no-p (ours)** | **71.9** | **78.4** | **87.0** | 73.0 | 85.8 | **89.0** | 85.5 | 92.3 | **82.6** | **83.2** |
| **BPMulT (ours)** | 65.0 | 70.3 | 85.5 | 73.5 | 78.8 | 76.4 | 81.1 | 85.6 | 77.6 | 76.4 |

**Table 6.11:** Comparison of our BPMulT model in the aligned IEMOCAP dataset where the model ModTrans-MMEmoRe is the current SOTA model corresponding to the model in Dai et al. (2020) for emotion recognition. Metrics reported are the accuracy and the Area Under the Curve (AUC) instead of F1 as Dai et al. (2020) proposed.

We performed random seed training for the BPMulT model. We achieved a better Accuracy score for Happy, Sad, and Angry emotions, which is an excellent result because the class with more labels is the Neutral emotion. On the other hand, Table 6.10 shows the results for the metrics considered in Tsai et al. (2019) because it is the first model used for unaligned sequences.

IEMOCAP Unaligned

| Model | Neutral | | Happy | | Sad | | Angry | | Average | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 |
| MulT | **62.5** | **59.7** | 84.8 | **81.9** | 77.7 | **74.1** | 73.9 | **70.2** | 74.7 | **71.5** |
| **BPMulT-no-p (ours)** | 59.6 | 45.7 | **85.6** | 79.0 | **79.4** | 70.3 | **75.8** | 65.4 | **75.1** | 65.1 |
| **BPMulT (ours)** | 59.2 | 44.0 | **85.6** | 79.0 | **79.4** | 70.3 | 75.6 | 66.2 | 74.9 | 64.9 |

**Table 6.12:** Comparison of our BPMulT model in the not aligned IEMOCAP dataset where the model MulT is the current SOTA model since this model was proposed for unaligned sequences. The metrics reported are the accuracy and the F1 score.

### 6.5.3 BPMulT in the CMU-MOSEI Dataset

CMU-MOSEI is the most helpful dataset to test multimodal models for emotion and sentiment analysis. It also has three modalities for information extraction. Its features came from Text, Audio, and Video. Similar to the IEMOCAP dataset, BPMulT works directly, and we fuse all three modalities with a GMU in the last part. For this version of the CMU-MOSEI, we took the proposed modification of Dai, Cahyawijaya, Liu, and Fung (2021), which provides a reordering and cleaning of some conversations. Hence, the SOTA model corresponds to the proposed models in the same publication (FE2E and MESM) for **unaligned** sequences. Since this dataset provides raw text

and we are working with crossmodal transformers, we select BERT as our encoder for text representation. Audio and Video representation are the provided vectors by the mentioned publication.

Table 6.13 shows the results for the metrics considered in Dai, Cahyawijaya, Liu, and Fung (2021). We performed five random training seeds for the BPMulT model to obtain the reported metrics' mean and standard deviation scores. We achieved better Weighted-Accuracy in almost every emotion.

CMU-MOSEI Unaligned

| Model | Angry W-Acc/F1 | Disgust W-Acc/F1 | Fear W-Acc/F1 | Happy W-Acc/F1 | Sad W-Acc/F1 | Surprised W-Acc/F1 | Average W-Acc/F1 |
|---|---|---|---|---|---|---|---|
| MulT | 64.9/47.5 | 71.6/49.3 | 62.9/25.3 | 67.2/**75.4** | 64.0/48.3 | 61.4/25.6 | 65.4/45.2 |
| FE2E | 67.0/**49.6** | **77.7**/57.1 | 63.8/26.8 | 65.4/72.6 | 65.2/**49.0** | **66.7**/**29.1** | 67.6/**47.4** |
| MESM (0.5) | 66.8/49.3 | 75.6/56.4 | 65.8/28.9 | 64.1/72.3 | 63.0/46.6 | 65.7/27.2 | 66.8/46.8 |
| **BPMulT-no-p (ours)** | 66.7/26.5 | 69.0/**75.6** | **66.2**/**48.7** | **74.7**/50.8 | **67.3**/48.5 | 62.9/26.5 | **68.3±.3**/46.1±.4 |
| **BPMulT (ours)** | **69.3**/26.4 | 68.0/**74.9** | **66.0**/**48.5** | **74.5**/50.9 | **67.1**/48.3 | 63.4/25.1 | **68.0±.6**/45.7±.6 |

**Table 6.13:** Comparison of our BPMulT model in CMU-MOSEI unaligned dataset modified by Dai, Cahyawijaya, Liu, and Fung (2021) where the model FE2E is the current SOTA model proposed in the same mentioned publication. The metrics reported are the weighted accuracy (W-Acc) and the F1 score.

# Chapter 7

# Conclusions

Accordingly to our objectives in Section 1.2, we have four specific objectives to achieve and one main goal, which is the primary purpose of our research. Following the structure of our specific motivations, we these conclusions of our work:

1) Our first specific objective is to improve the modalities' combination. We proposed a novel architecture that involves a biprojection that enriches each modality representation with information from the other modalities. Ablation experiments and results show that this biprojection is crucial to rescue relevant information from the sequences since it also allows residual connections and intermediate fusion modules to be possible. Hence, we achieve our first goal with the proposed crossmodal biprojection.

2) The second specific objective is to substitute the heavy information fusion method (the transformer). We proposed the Fusion GMU (FGMU) module, a lower-cost method that does not decrease the performance in classification. Ablation experiments and results show that the FGMU helps the model to improve its classification performance. It also is easy to interpret because it learns to weigh each modality to fuse them. We can see the used weights and interpret them as feature activations. Then, we achieve our second goal with this proposed module.

3) The third specific objective is to improve learning by introducing strategically

residual connections. Since we proposed to introduce the crossmodal biprojections to our model, we facilitate addressing this problem because we can use the information from the first projection and enrich each modality. Using the FGMU also helped to place better connections linking crossmodal blocks' summarized information. The ablation experiment shows that our proposed configuration of connections has the best performance. Hence, we satisfied our third goal with the connections of biprojections and FGMU modules.

4) Finally, our fourth specific objective is to understand the information flow in the proposed architecture. With the FGMU, we achieved this goal because it provides an interpretable weighing system. We obtained the following conclusions with the activation of the GMU and FGMU modules. With the TSNE analysis, we see the excellent clusterization of Moviescope test labels when we see its FGMU activation in the projections and biprojections. We detect that in the branch of the text features (Figure 7.1), the projections are not well clustered, and the biprojections are. With the FGMU activation analysis in Section 6.4.4, we see that the information (Figure 6.18) is mainly used from (A—›V), which is poorly clustered and (A—›V—›L) which is good. It results in a low activation for the text modality in the final GMU prediction, accordingly to Figure 6.12.

   In contrast, in the branch of the video features, the projections are well clustered, but the biprojections are not. The FGMU activation analysis shows that the information (Figure 6.18) is equally activated and is used even if it is good and bad clustered data. It results in a high, but not highest, activation for the video modality in the final GMU prediction, following Figure 6.12.

   Moreover, in the branch of the audio features, something different happens: one projection is good activated as well as its biprojection, and the other is poorly clustered and also its respective biprojection. The information mainly corresponds to the good clustered data with the FGMU activation analysis (Figure 6.18). It results in the highest activation, the audio modality in the final GMU prediction, accordingly to Figure 6.12.

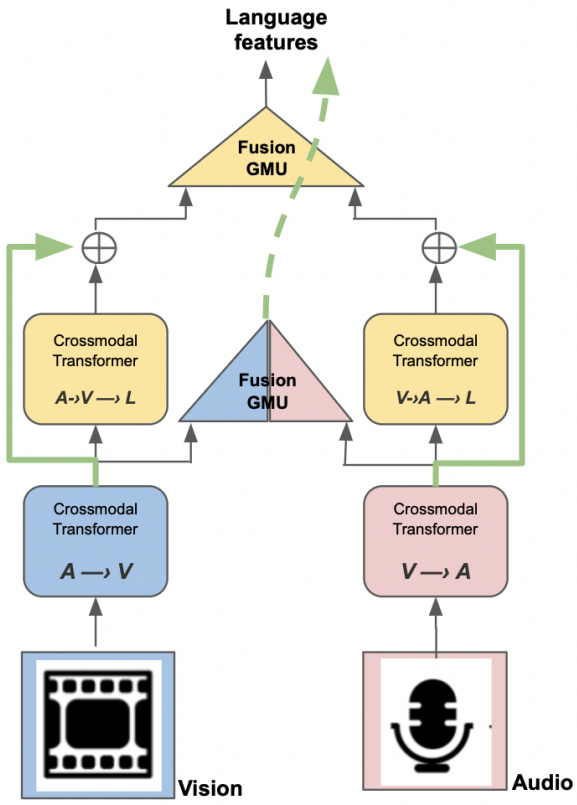



**Figure 7.1:** Text branch of the Biprojection Multimodal Transformer (BPMult) model. It consists of two opposite projections ((A—›V) and (V—›A)) and their respective biprojections to the text space.

In conclusion, biprojections and single projections are relevant if we only see the clustering with the TSNE study. The advantage of the BPMulT model is that it dynamically takes into account features of both single crossmodal transformers and biprojections. However, an ablation experiment shows that the biprojections have more activation than projections, and in consequence, biprojection can be seen as more relevant.

Finally, our main objective is to improve the representation of modalities' combinations within the Multimodal Transformer architecture. Also, we aim to achieve superior performance to the SOTA models in movie genre classification and emotion recognition tasks. The BPMulT model improves the representation of each modality with information from the other modalities. It is done with the help of biprojections and fusion modules. Results show that the activation increase for modalities not rele-

vant before. Our proposed BPMulT model has mainly achieved our goal and obtained the SOTA scores for the Moviescope and MM-IMDb datasets. The BPMulT has also been tested on various multimodal datasets and has achieved competitive results in the IEMOCAP and CMU-MOSEI datasets.

Another marvelous thing is that the BPMulT has been tested even when we have less than three modalities, like in the MM-IMDb dataset. The proposed extension to handle this kind of dataset is that the BPMulT takes the sequence of one modality with different embedding representations.

## 7.1 Future Work

The BPMulT considers the single projections and the biprojections using residual connections and FGMU modules. With a TSNE study, we found that sometimes, a crossmodal transformer does not have an apparent label clustering. It happens with not the same modalities, projections, and biprojections and is different in each dataset. We believe that it could be a specific line to follow. To develop an architecture that could automatically detect whether one crossmodal transformer has a good clustering with the TSNE study or not. If the architecture detects that it does not have a good clustering, it will not have relevant information for the prediction.

Another line to follow is how to reduce the BPMulT because it is a heavy architecture. It could be addressed by taking an efficient crossmodal attention module with attention from two modalities, i.e., the biprojection packed in one attention module.

# References


Alammar, J. (2018). The illustrated transformer. Retrieved from https://jalammar.github.io/illustrated-transformer

Arevalo, J., Solorio, T., Montes-y Gómez, M., & González, F. A. (2017). Gated multimodal units for information fusion. *Workshop track - ICLR*. Retrieved from https://arxiv.org/pdf/1702.01992.pdf

Baltrušaitis, T., Ahuja, C., & Morency, L.-P. (2017). Multimodal machine learning: A survey and taxonomy. *Arxiv*. Retrieved from https://arxiv.org/pdf/1705.09406

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Arxiv*. Retrieved from https://arxiv.org/pdf/1607.04606

Busso, C., Bulut, M., Lee, C., Kazemzadeh, A., Mower, E., Kim, S., ... Narayanan, S. (2008). Iemocap: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, *42/4/335-359*.

Cascante-Bonilla, P., Sitaraman, K., Luo, M., & Ordonez, V. (2019). Moviescope: Large-scale analysis of movies using multiple modalities. *ArXiv*, *abs/1908.03180*.

Chandrasekaran, G., Nguyen, T., & Hemanth, J. (2021). Multimodal sentimental analysis for social media applications: A comprehensive review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. Retrieved from e1415

Charland, P., Léger, P., Sénécal, S., & Courtemanche, F. (2015). Assessing the multiple dimensions of engagement to characterize learning: A neurophysiological

perspective. *JoVE*. Retrieved from doi:10.3791/52627

Chauhan, D. S., Akhtar, M. S., Ekbal, A., & Bhattacharyya, P. (2019). Context-aware interactive attention for multi-modal sentiment and emotion analysis. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, *2019*(1), 5647–5657.

Chauhan, P., Sharma, N., & Sikka, G. (2021, 01). Multimodal sentiment analysis of social media data: A review. In (p. 545-561). doi: 10.1007/978-981-15-8297-4_44

Dai, W., Cahyawijaya, S., Bang, Y., & Fung, P. (2021). Weakly-supervised multi-task learning for multimodal affect recognition. *arXiv preprint arXiv:2104.11560*, *2021*(6), 3584–3593.

Dai, W., Cahyawijaya, S., Liu, Z., & Fung, P. (2021, June). Multimodal end-to-end sparse model for emotion recognition. *Proceedings of the 2021 Conference of the North American*, 5305–5316. Retrieved from https://arxiv.org/pdf/2103.09666v3.pdf

Dai, W., Liu, Z., Yu, T., & Fung, P. (2020, December). Modality-transferable emotion embeddings for low-resource multimodal emotion recognition. *Proceedings of the 1st Conference of the Asia-Pacific*, 269–280. Retrieved from https://arxiv.org/pdf/2009.09629.pdf

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. *Association for Computational Linguistics*. Retrieved from https://aclanthology.org/N19-1423.pdf

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *Arxiv*. Retrieved from https://arxiv.org/pdf/1512.03385

Kiela, D., Bhooshan, S., Firooz, H., Perez, E., & Testuggine, D. (2019). Supervised multimodal bitransformers for classifying images and text. *Arxiv*. Retrieved from https://arxiv.org/abs/1909.02950

Mikolov, T., Corrado, G., Chen, K., & Dean, J. (2013). Efficient estimation of word representations in vector space. *Arxiv*. Retrieved from https://arxiv.org/

pdf/1301.3781

Nikolić, M., Majdandžić, M., Colonnesi, C., de Vente, W., Möller, E., & Bögels, S. (2020). The unique contribution of blushing to the development of social anxiety disorder symptoms: results from a longitudinal study. *J. Child Psychology Psychiatry*. Retrieved from doi:10.1111/jcpp.13221

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. *Association for Computational Linguistics*. Retrieved from https://aclanthology.org/D14-1162

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training.

Rodríguez-Bribiesca, I., López-Monroy, A. P., & y Gómez, M. M. (2021). Multimodal weighted fusion of transformers for movie genre classification. *Proceedings of the Third Workshop on Multimodal Artificial Intelligence*, 1–5. Retrieved from https://aclanthology.org/2021.maiworkshop-1.1.pdf

Scikit-learn api reference. (2022). Retrieved from https://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics

Sleeman-IV, W., Kapoor, R., & Ghosh, P. (2021). Multimodal classification: Current landscape, taxonomy and future directions. *Arxiv*. Retrieved from https://arxiv.org/pdf/2109.09020.pdf

Sourav, S., & Ouyang, J. (2021). Lightweight models for multimodal sequential data. *Proceedings of the 11th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, *2021*(1), 129–137.

Tsai, Y.-H. H., Bai, S., Liang, P. P., Kolter, J. Z., Morency, L.-P., & Salakhutdinov, R. (2019, August). Multimodal transformer for unaligned multimodal language sequences. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6558–6569. Retrieved from https://aclanthology.org/P19-1656.pdf

Vaswani, A., Jones, L., Shazeer, N., Parmar, N., Uszkoreit, J., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *31st Conference on Neural Information Processing Systems*. Retrieved from https://arxiv.org/pdf/

1706.03762.pdf

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., & Norouzi, M. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *Arxiv*. Retrieved from https://arxiv.org/pdf/1609.08144

Xu, P., Zhu, X., & Clifton, D. A. (2022). Multimodal learning with transformers: A survey. *Arxiv*. Retrieved from https://arxiv.org/pdf/2206.06488

Yao, Y., Papakostas, M., Burzo, M., Abouelenien, M., & Mihalcea, R. (2021). Muser: Multimodal stress detection using emotion recognition as an auxiliary task. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, *2021*(1), 2714–2725.

Zadeh, A., Liang, P. P., Poria, S., Cam-bria, E., & Morency, L.-P. (2018). Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph. *ACL*, *2018*(1), 0–5.

Zhang, D., Ju, X., Li, J., Li, S., Zhu, Q., & Zhou, G. (2020). Multi-modal multi-label emotion detection with modality and label dependence. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, *2020*(1), 3584–3593.

# Appendix A

# Figures

In this Appendix, we will find figures of the TSNE studies of the Moviescope test set activation of every GMU and FGMU module. Figure A.1 shows the TSNE study for the video features activation on the GMU at the top of the architecture. We can see that the colored labels are well-grouped in the space, which is a signal of the video modality's high relevance in the final prediction. Figures A.2 and A.3 shows also a good result in the TSNE study grouping the colors similarly. In the Results Chapter (6), we say that we need to analyze the information's flow better and proceed to do other studies.

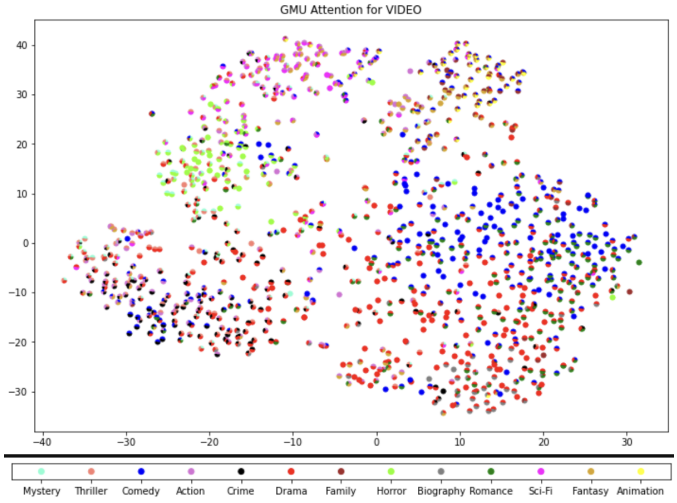



**Figure A.1:** TSNE study of the output of the last GMU for dynamic fusion of modalities in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. We are visualising the activations corresponding to the **video** part. It is considered a **good** clusterization.

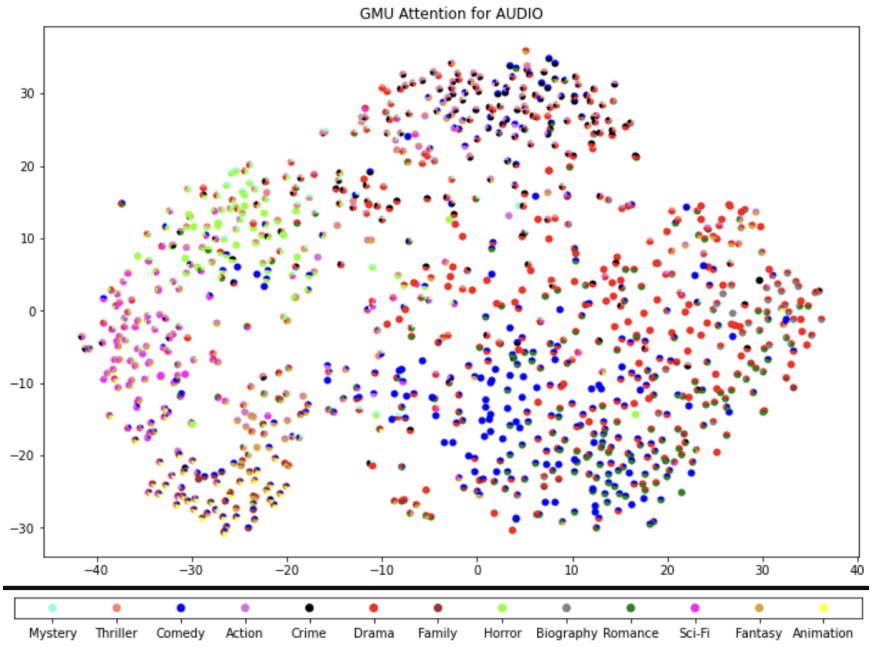



**Figure A.2:** TSNE study of the output of the last GMU for dynamic fusion of modalities in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. We are visualising the activations corresponding to the **audio** part. It is considered a **good** clusterization.

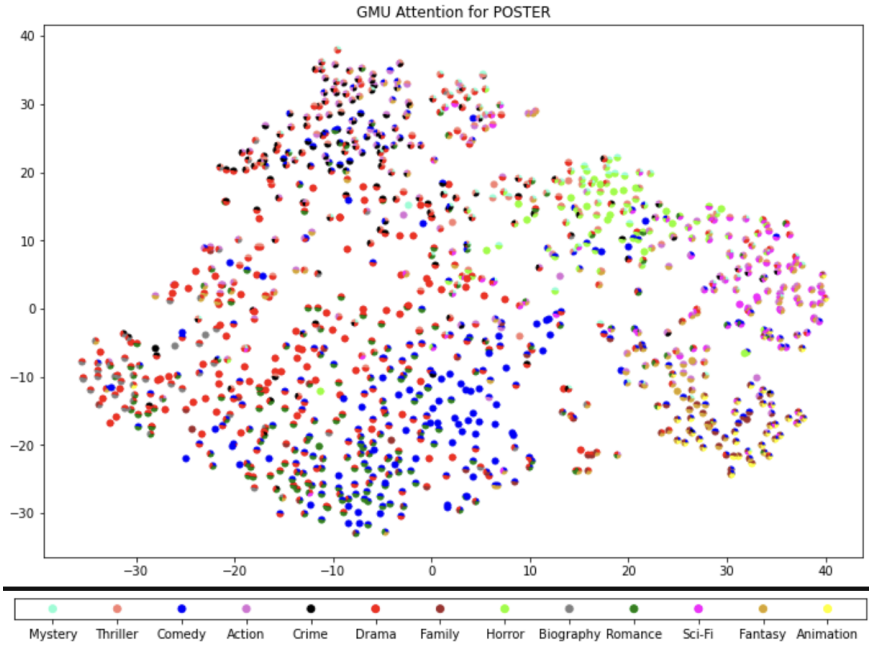



**Figure A.3:** TSNE study of the output of the last GMU for dynamic fusion of modalities in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. We are visualising the activations corresponding to the **poster** part. It is considered a **good** clusterization.

Figures A.4 and A.5 show the FGMU activation of the test set when data passes through the branch to get the video features. In the first projection (Figure A.4), the audio features of the text space have a good clustering. Then, the clustering is terrible when we do a biprojection to the video space (Figure A.5). Similarly, with the same branch of video features, if we make a projection from text to audio (Figure A.6) is good. If we do a biprojection to the video space (Figure A.7), clustering goes not interesting.

On the other hand, if we take now the branch of the audio features, we can see not the same case. The first projections (Figures A.8 and A.10) show that projecting from text to the video space results in a bad clustering, and the projection from video to text is a good option. Biprojection from the bad clustered (L—›V) projection to the audio space (Figure A.9) also results in a bad clusterization. In contrast, the well clustered (V—›L) projection results in a good enough clusterization with the biprojection to the audio space.

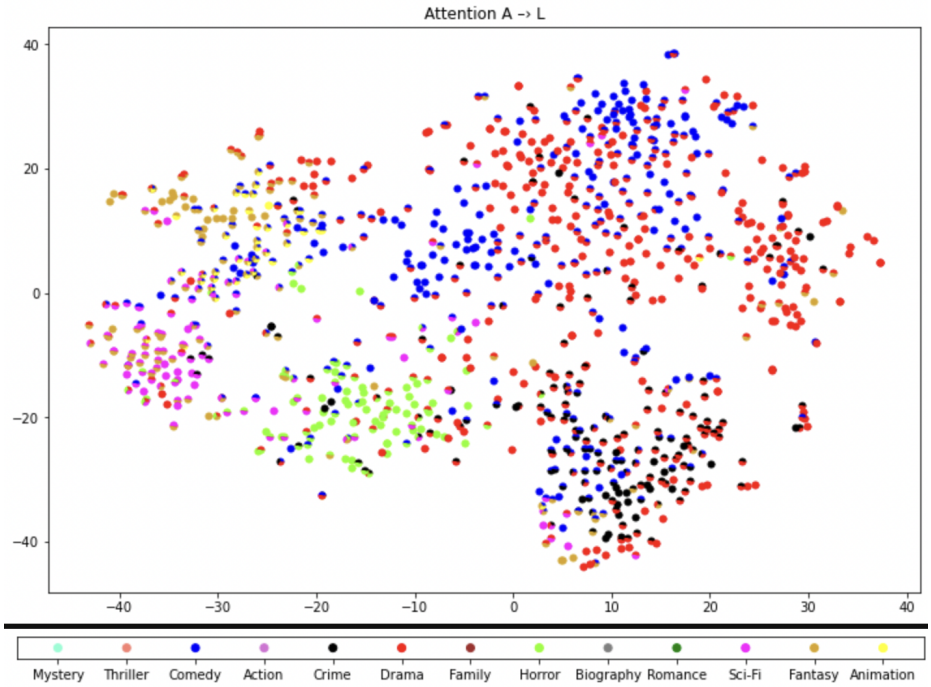



**Figure A.4:** TSNE study of the output of the crossmodal transformer (A—›L) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **good** clusterization.

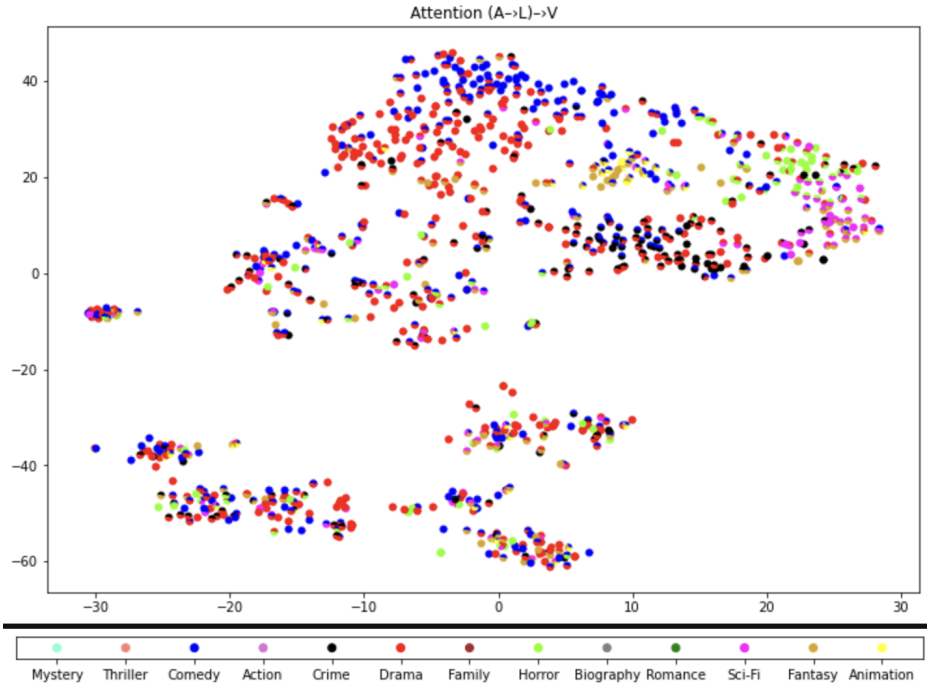



**Figure A.5:** TSNE study of the output of the crossmodal transformer (A—›L—›V) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **bad** clusterization.

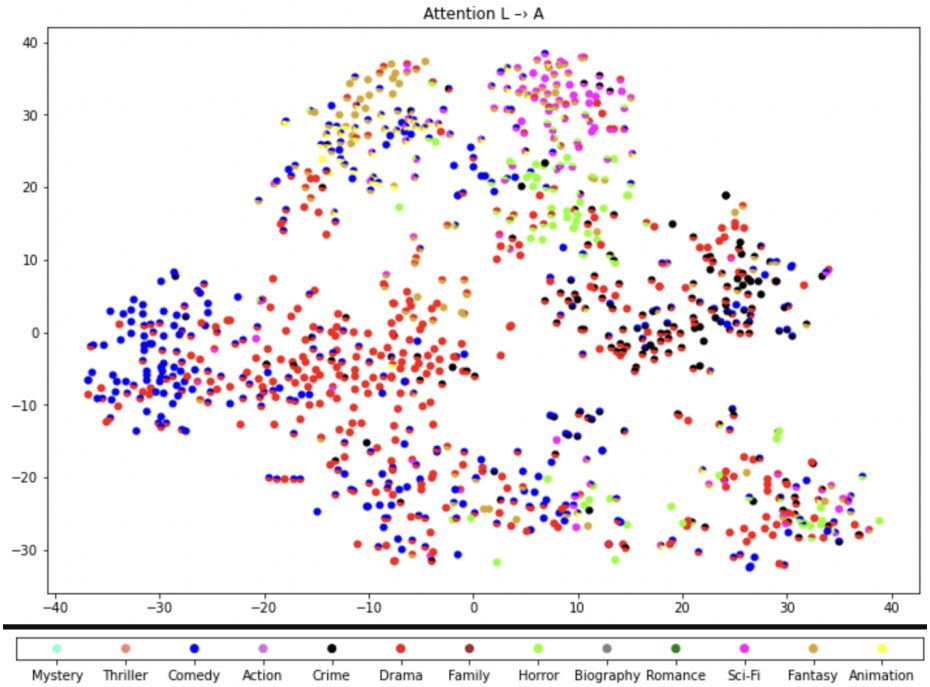



**Figure A.6:** TSNE study of the output of the crossmodal transformer (L—›A) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **good** clusterization.

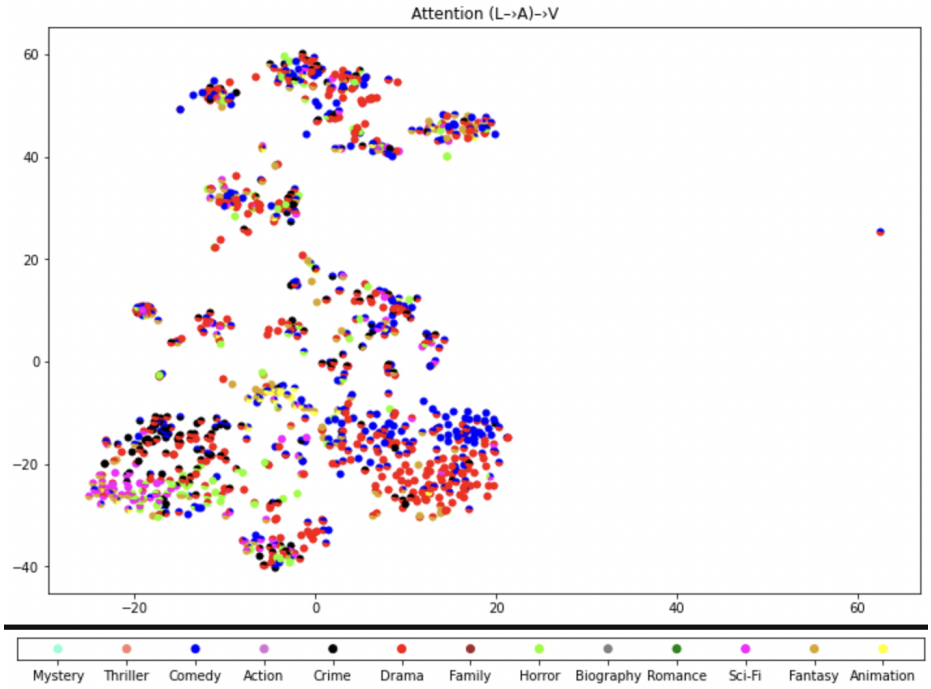



**Figure A.7:** TSNE study of the output of the crossmodal transformer (L—›A—›V) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **bad** clusterization.

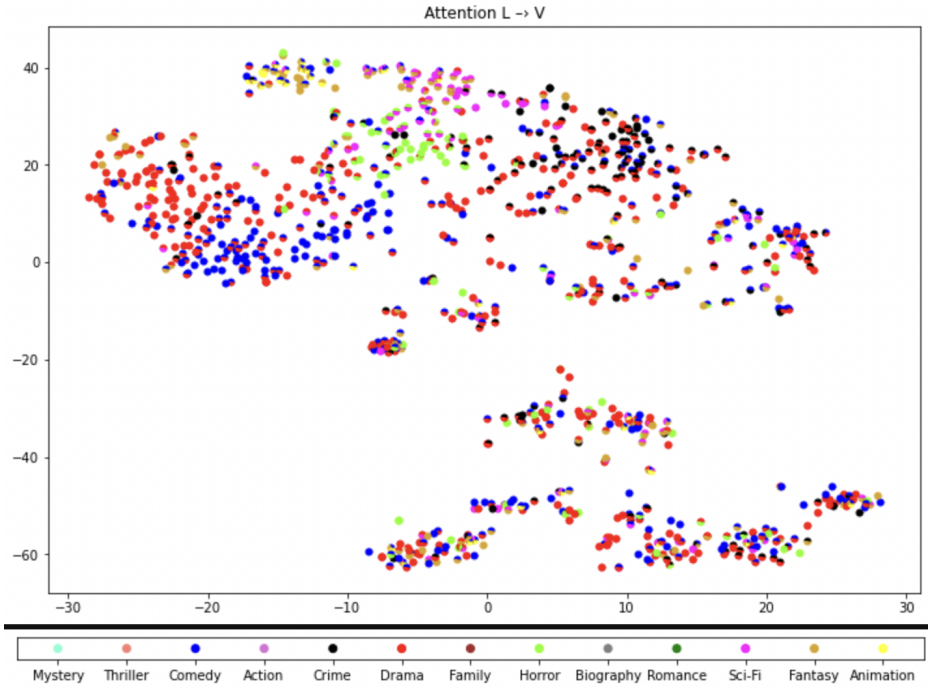



**Figure A.8:** TSNE study of the output of the crossmodal transformer (L—›V) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **bad** clusterization.

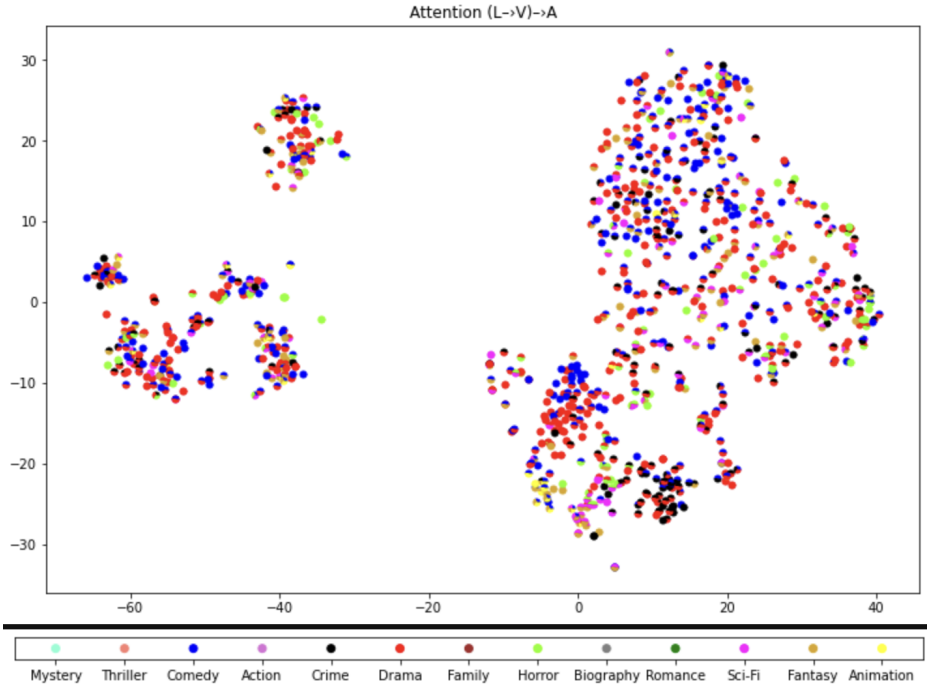



**Figure A.9:** TSNE study of the output of the crossmodal transformer (L—›V—›A) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **bad** clusterization.

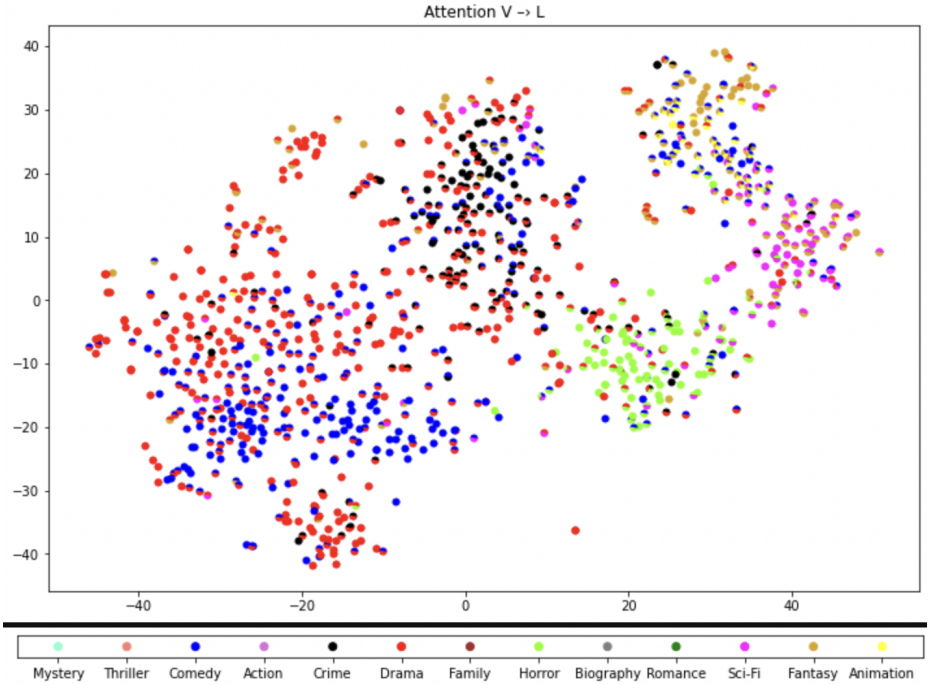



**Figure A.10:** TSNE study of the output of the crossmodal transformer (V—›L) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **good** clusterization.

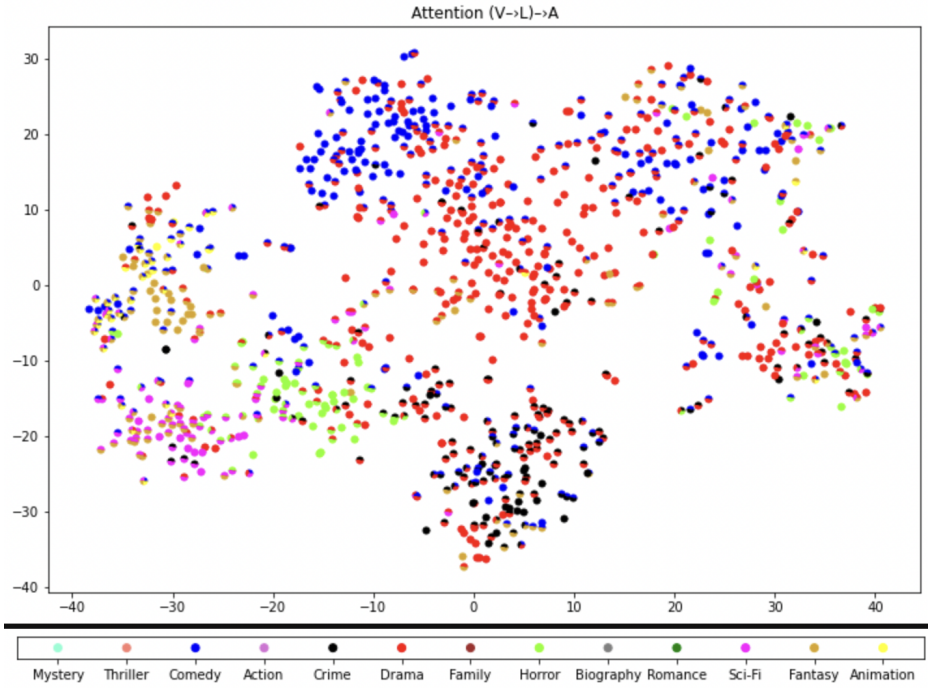



**Figure A.11:** TSNE study of the output of the crossmodal transformer (V—›L—›A) in the BPMulT model. Each color corresponds to a genre and data is multi-labeled. It is considered a **good** clusterization.