Dynamic Regularization in UDA for Transformers in Multimodal Classification

Ivonne Monter-Aldana[*], **A. Pastor López-Monroy**[*], and **Fernando Sánchez-Vega**[*, †]
[*] Department of Computer Science,
Mathematics Research Center (CIMAT), Gto. Mexico.
[†] Consejo Nacional de Ciencia y Tecnología (CONACyT),
CDMX, México.
{ivonne.monter, pastor.lopez, fernando.sanchez}@cimat.mx

Abstract

Multimodal machine learning is a cutting-edge field that explores ways to incorporate information from multiple sources into models. As more multimodal data becomes available, this field has become increasingly relevant. This work focuses on two key challenges in multimodal machine learning. The first is finding efficient ways to combine information from different data types. The second is that often, one modality (e.g., text) is stronger and more relevant, making it difficult to identify meaningful patterns in the weaker modality (e.g., image). Our approach focuses on more effectively exploiting the weaker modality while dynamically regularizing the loss function. First, we introduce a new two-stream model called Multimodal BERT-ViT, which features a novel intra-CLS token fusion. Second, we utilize a dynamic adjustment that maintains a balance between specialization and generalization during the training to avoid overfitting, which we devised. We add this dynamic adjustment to the Unsupervised Data Augmentation (UDA) framework. We evaluate the effectiveness of these proposals on the task of multi-label movie genre classification using the Moviescope and MM-IMDb datasets. The evaluation revealed that our proposal offers substantial benefits, while simultaneously enabling us to harness the weaker modality without compromising the information provided by the stronger.

1 Introduction

Multimodal machine learning focuses on methods for modeling information from more than one modality, such as text, image, audio, and video. It is crucial for models to simultaneously analyze and organize diverse modalities. Recent advancements in multimodal applications have primarily been attributed to the availability of new large-scale multimodal datasets, increased computational capacity, and enhanced representations of individual modalities. This development encompasses a broad scope, for example, multimodal classification (Arevalo et al., 2017; Cascante-Bonilla et al., 2019), gesture recognition (Yu et al., 2021), and audio-video clustering (Alwassel et al., 2019).

Multimodal machine learning is a great challenge due to the multiple factors involved. In this work, we focus on multi-label movie genre classification utilizing two modalities: text and image. We focus on addressing four main difficulties: (i) Models trained on different modalities learn and generalize at different speeds (Wang et al., 2020), and the optimization process benefits one modality more than the other. The consequence is that during the training phase, the model gives more weight to the most relevant (stronger) modality and disregards the less informative (weaker) one. (ii) There is a lack of effective fusion since prediction by fusion modalities depends on the correlation between modalities and their representation. (iii) Complex deep neural architectures require a large computing capacity, and most multimodal models use complex modules for fusion that increment the computational cost. (iv) Multimodal models are prone to overfitting since they have more parameters (Wang et al., 2020).

To tackle these issues, our proposal focuses on model regularization and better use of weak modality information (e.g., image). Specifically, for (i) and (iii), we design a two-stream model called Multimodal BERT-VIT (MMBV). Our proposed MMBV consists of two Transformers (Vaswani et al., 2017), one for text and one for image, interconnected by the CLS token. Thereby, MMBV combines the modalities using self-attention without a complex fusion module, unlike most of the previous classification approaches that are based on large complex neural modules that increase computational cost and are prone to overfitting (Wang et al., 2020). MMBV

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 8700–8711
July 9-14, 2023 ©2023 Association for Computational Linguistics

is possible due to the recent creation of the Vision Transformer (ViT) (Dosovitskiy et al., 2021), a transformer pre-trained in a large image dataset that incorporates the CLS token for classification similar to text Transformers.

For (ii) and (iv), we design a ratio for the specialization and generalization of a model at each epoch to identify the onset of overfitting and correct it. We add the ratio to the consistency training framework Unsupervised Data Augmentation (UDA) (Xie et al., 2020). The intuitive idea is to obtain a score that automatically increases or decreases the contribution of the consistency loss in UDA during training. For multimodal classification, we can augment only one modality or the two modalities jointly. To reduce the breach between the learning speeds of different modalities, we experiment with augmenting only the strong modality.

We evaluate these ideas in the challenging scenario of multi-label classification of movie genre using two datasets (Cascante-Bonilla et al., 2019; Arevalo et al., 2017). Our results show that combining MMBV with dynamic UDA improves the performance of both image and text modalities by balancing between the weak and strong modalities. We also find that data augmentation in just one modality helps the other one. The data augmentation retards the training, which may reduce the difference between the learning rates of different modalities. Overall, the dynamic UDA improves the base model performance across all models tested.

2 Related work

2.1 Multimodal Fusion

Fusing modalities for multimodal prediction is a crucial challenge since it depends on the task, correlations between the modalities, and the input representation of the modalities. The study of merging modalities is an active research field due to the diverse characteristics of multimodal data sets. For example, modalities with little discriminative power, those that are contradictory, or those that are redundant and represent the same semantic concept. Models in the literature have used feature extractors and encoders for each modality to transform inputs into continuous numeric vectors that are ready for fusion. Methods include gated neural networks (Arevalo et al., 2017) to learn a linear combination of vector representations, projecting one modality's matrix representation onto another and concatenating (Kiela et al., 2020), and using self-attention or cross-attention transformer layers to interact between modalities (Kiela et al., 2020; Lu et al., 2019).

2.2 Previous Text-Image Classification Models

We categorize multimodal models into two types: multimodal pre-training (Lu et al., 2019; Chen et al., 2020; Gan et al., 2020) and unimodal pre-training (Tsai et al., 2019; Kiela et al., 2020). The first type refers to models trained in agnostic multimodal tasks using both modalities to learn a joint representation of text and image. The second type defines models where each encoder is independently pre-trained in unimodal tasks to learn a representation of the individual modality.

Models with unimodal pre-training, such as (Tsai et al., 2019; Kiela et al., 2020), have the advantage of being able to easily replace encoders at no extra cost. However, their performance is typically lower due to a lack of joint representation learning. In this paper, we consider, adapt and study these models to perform fine-tuning and observe the behavior of learning speeds for each modality.

Despite the high computational costs, multimodal models typically perform better in most tasks (Lu et al., 2019; Chen et al., 2020; Gan et al., 2020). One major issue is that these models often have complex attention modules that require more memory and time during fine-tuning. Another limitation is that the models have a text-based architecture. Thus the image could be at a disadvantage concerning the text, resulting that the performance of the multimodal models being similar to the text-only model. Fortunately, recent advancements in adapting transformers to visual tasks have yielded important results (Dosovitskiy et al., 2021; Bao et al., 2021; Touvron et al., 2021). In this paper, we present a simple but effective adaptation of an image transformer to create a new multimodal model that achieves a better balance between modalities.

2.3 Regularization for multimodal models

A primary issue with multimodal networks is overfitting, as they usually have more parameters than analogous unimodal models. In a notable work, Wang et al. (2020) found that overfitting occurs because different modalities generalize and specialize at different rates. To address this issue, they computed an overfitting-to-generalization ratio (OGR) per modality to measure the quality of training between two model checkpoints.

The OGR between epochs N and $N+n$ is defined as:

$$OGR = \left| \frac{O_{N+n} - O_N}{L_N^V - L_{N+n}^V} \right| \quad (1)$$

where L_N^T is the model's average loss over the fixed train set, L_N^V is the validation loss and $O_N = L_N^V - L_N^T$. Finally, they proposed to minimize the OGR during training by using gradient-blending.

Another approach to reducing overfitting is adversarial training, Gan et al. (2020) propose an adversarial framework to avoid overfitting during pre-training and fine-tuning by adding adversarial perturbations to the embedding space of each modality. This improves the generalization of pre-trained models but is computationally expensive and time-consuming.

Model regularization is crucial for multimodal learning, and inspired by the concept of OGR, we propose a novel score to quantitatively measure overfitting. We integrate this overfitting score with the UDA consistency training framework, which was previously used only in unimodal contexts. We will discuss this method further in the next section.

3 CLS fusion model: MMBV

We introduce the Multimodal BERT-ViT (MMBV) model [1], a novel two-stream model for text-image classification that combines two transformers through the CLS token. See Figure 1 for an illustration of the architecture. We use as image encoder the novel pre-trained transformer for image ViT (Dosovitskiy et al., 2021), and for the text encoder, we use the transformer BERT (Devlin et al., 2019). In this way, both modalities have an encoder with a similar architecture, where the encoders are pre-trained in large datasets for each domain. Since both models have a token (CLS) that resumes the information of the input sequence, we do the fusion by interconnecting this token from one modality to the other.

3.1 CLS fusion

BERT and ViT models have a CLS hidden state at the same dimension, $h_{CLS}^{(i)}$, that resumes the information of the input at the end of the i transformer block. We use the CLS hidden states to connect the two models. At the end of the first block, we have r hidden states for the text input $H_{txt}^{(1)} = (h_{CLS_{txt}}^{(1)}, h_{1_{txt}}^{(1)}, \ldots, h_{r_{txt}}^{(1)})$ and s hidden states for the image input $H_{img}^{(1)} = (h_{CLS_{img}}^{(1)}, h_{1_{img}}^{(1)}, \ldots, h_{s_{img}}^{(1)})$. $H_{txt}^{(1)}$ is the output of the first self-attention block for the text and $H_{img}^{(1)}$ is the output of the first self-attention block for the image. These hidden states are the input for the next block and to add the information from the other modality at each model, we concatenate $h_{CLS_{img}}^{(1)}$ to $H_{txt}^{(1)}$ and $h_{CLS_{txt}}^{(1)}$ to $H_{img}^{(1)}$. The input for the next block of BERT is:

$$\hat{H}_{txt}^{(1)} = (h_{CLS_{txt}}^{(1)}, h_{1_{txt}}^{(1)}, \ldots, h_{r_{txt}}^{(1)}, h_{CLS_{img}}^{(1)}) \quad (2)$$

and the input for the next block of ViT is:

$$\hat{H}_{img}^{(1)} = (h_{CLS_{img}}^{(1)}, h_{1_{img}}^{(1)}, \ldots, h_{s_{img}}^{(1)}, h_{CLS_{txt}}^{(1)}) \quad (3)$$

The input for the second block has an extra hidden state. From this point, to build the input for the next transformer block, instead of concatenating, the last hidden state is replaced by the CLS hidden state of the other model. The inputs for the $i+1$ block of each model are:

$$\hat{H}_{txt}^{(i)} = (h_{CLS_{txt}}^{(i)}, h_{1_{txt}}^{(i)}, \ldots, h_{r_{txt}}^{(i)}, h_{CLS_{img}}^{(i)}) \quad (4)$$

$$\hat{H}_{img}^{(i)} = (h_{CLS_{img}}^{(i)}, h_{1_{img}}^{(i)}, \ldots, h_{s_{img}}^{(i)}, h_{CLS_{txt}}^{(i)}) \quad (5)$$

The cross-modal CLS token concatenation only happens in the output of the first transformer block. After that, the last token produced for each subsequent layer is replaced by the corresponding CLS token of the other modality built in the corresponding transformer block.

Usually, the fusion between modalities is time-consuming and computationally expensive, but with this approach, we have effective fusion at a low cost. With the CLS fusion, each transformer has access to the resume of the input from the other transformer with only a token. Then, this compressed representation is combined with each token of the original sequence input by the self-attention mechanism.

4 Proposed Dynamic Unsupervised Data Augmentation

To extend data augmentation to the multimodal domain, we adapt the semi-supervised framework

[1] The code is available at https://github.com/IvonneMont/Dynamic-UDA-for-Transformers-in-Multimodal-Classification.git

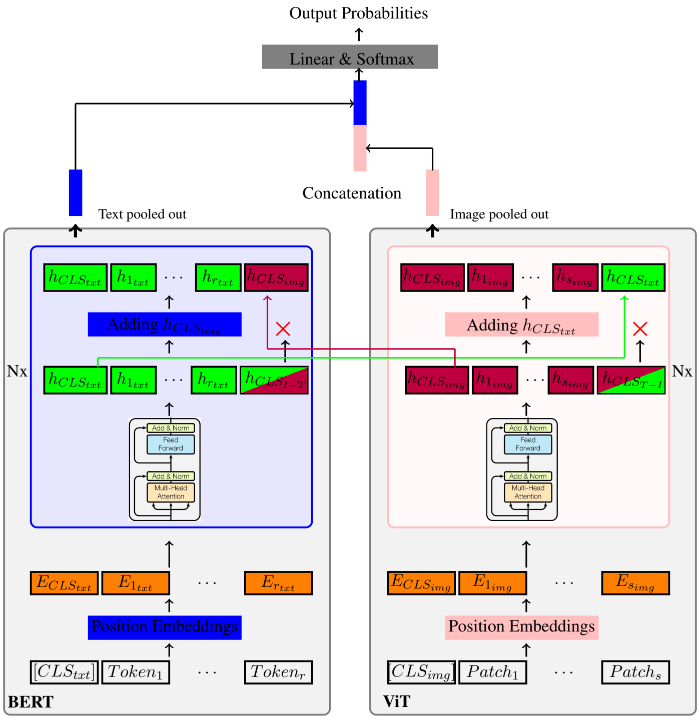


Figure 1: CLS fusion model: The proposed model MMBV is a model of two streams, the ViT transformer (Dosovitskiy et al., 2021) for image, and the BERT transformer (Devlin et al., 2019) for text. The two transformers are connected by the CLS embedding in each block. The tokens of the text ($Token_i$) are the input for BERT, and the patches of the image ($Patch_i$) are the input for ViT. The $h_{CLS_{I-T}}$ and $h_{CLS_{T-I}}$ hidden states are not used for the input of the next self-attention block; instead, they are replaced by the $h_{CLS_{img}}$ and $h_{CLS_{txt}}$ versions from the transformer of the other modality.

UDA (Xie et al., 2020) to supervised multimodal classification. UDA has not yet, to the best of our knowledge, been used for this task. The purpose of UDA is to force model consistency with realistic noise by applying data augmentation techniques. We propose a dynamic regularization of the loss function that is useful when there is not a large unsupervised data set and the computational resources are limited.

4.1 Standard UDA

UDA (Xie et al., 2020) computes a supervised loss over labeled data and a consistency loss over unlabeled data to gradually propagate label information from labeled examples to the unlabeled. A regularization parameter λ weights the consistency loss to control how much we want to consider the unlabeled data and allow for model flexibility.

They consider a model M that estimates a conditional distribution $P_\theta(y|x)$, for a given input x. To calculate the consistency loss, the model uses data augmentation. Given an input x from the unlabeled dataset, its noised version $\hat{x}$ is obtained from x applying a data augmentation transformation. Then, the consistency loss L_c is defined as:

$$L_c = \frac{1}{|U|}\sum_{x\in U}\mathcal{D}(p_{\hat{\theta}}(y|x) \parallel p_\theta(y|\hat{x})) \qquad (6)$$

where $\mathcal{D}(p_{\hat{\theta}}(y|x) \parallel p_\theta(y|\hat{x}))$ is a divergence metric between the two distributions $p_{\hat{\theta}}(y|x)$ and $p_\theta(y|\hat{x})$ and U is the unlabeled dataset. The objective is that the distribution of the augmented data is similar to the distribution of original data, therefore the gradient only propagates through $p_\theta(y|\hat{x})$. The latter is achieved by fixing the θ parameter in $p_{\hat{\theta}}(y|x)$, i.e $p_{\hat{\theta}}(y|x)$ is consider a constant.

Finally, the objective function is calculated as the sum of a supervised loss L_s and λ times the unsupervised consistency loss

$$L = L_s + \lambda L_c \qquad (7)$$

Experimentally, it has been found that large amounts of unlabeled data and considerably large batch sizes are necessary for the regularization to correctly work, which is computationally expensive and infeasible when only limited computational resources are available.

4.2 New Dynamic UDA[2]

Since we work in the supervised scheme, we use only labeled data. Then, the consistency loss is calculated over the supervised dataset as per Ramesh et al. (2021). However, the model becomes susceptible to the choice of λ in Equation 7. For large values of λ, the model generalizes well but underfits. This means that the model learns the most general characteristics of the classes without obtaining discriminative patterns that can help to get high performance. Therefore, it tends to obtain a similar performance on the training and validation data set. On the contrary, for small values of λ, the model specializes well in the training dataset but overfits. This means that the model learns many detailed features of the training dataset without the appropriate generalization. It thus tends to misclassify validation instances.

To achieve a balance between generalization and specialization, we propose computing the coefficient λ based on the model's generalization and specialization at each epoch. The idea is to dynamically adjust λ , decreasing it when the model is generalizing well, and increasing it when the model is over-specializing. Therefore, it is essential to have a quantitative measure of the model's generalization and specialization. Inspired by the integration of the minimization of the overfitting-to-generalization ratio (OGR) into the objective function (see Wang et al. (2020)), we exploit the loss in validation $\mathcal{L}^V$ as an approximation of the loss over the target distribution.

First we define the next quotient at the epoch i:

$$q_i^V = \frac{1 + \mathcal{L}_{i-1}^V}{1 + \mathcal{L}_{i-2}^V} \tag{8}$$

$$q_i^T = \frac{1 + \mathcal{M}_{i-1}^T}{1 + \mathcal{M}_{i-2}^T} \tag{9}$$

where $\mathcal{L}_i^V$ is the loss in the validation set, $\mathcal{M}_i^T$ is the metric to improve as calculated for the training set (F1 score, accuracy, AP). Note that $q_i^V < 1$ if the validation loss decrease in the epoch $i - 1$, i.e. there was generalization. During training, it is expected that $\mathcal{M}_i^T$ will increase, but a rapid increase could result in overfitting, so we consider that $q_i^T >> 1$ could indicate overfitting.

Case	Explanation	Result
1. ${q_i}^V > 1, q_i^T > 1$	No generalization and overfitting	λ_i increase
2. $q_i^V < 1, (q_i^T)^K > \frac{1}{(q_i^V)^N}$	overfitting >> generalization	λ_i increase
3. $q_i^V < 1, (q_i^T)^K < \frac{1}{(q_i^V)^N}$	overfitting << generalization	λ_i decrease

Table 1: Behavior of λ_i in the three main cases.

Finally we defined the weighted factor λ_i as:

$$R = (q_i^V)^N (q_i^T)^K \tag{10}$$

$$\lambda_i = \lambda \cdot R \tag{11}$$

where $N, K \in \mathbb{N}$ and $\lambda \in \mathbb{R}$, and λ is the magnitude of the contribution of the consistency loss, N controls the flexibility of the model when there is generalization, and K controls the tolerance to the over-fitting. Since we are interested in improving the generalization we consider $N >> K$. The ratio R controls the magnitude of λ: when R is greater than one it is because specialization is greater than generalization and overfitting begins, therefore the contribution of consistency loss must be increased.

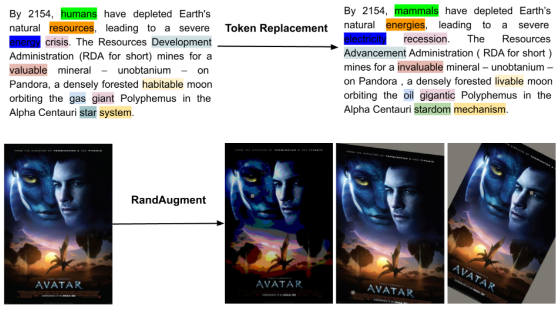
To analyze the behavior of λ_i we split it into three cases as shown in Table 1 . In case one, the learned weights improve the training metric, but the learned patterns do not generalize to the target distribution. Thus, we increase the value of λ to stop over-specialization in the training dataset. For case two, the learned patterns generalize the target distribution but not enough. Patterns in the training dataset are being learned very quickly, which could result in overfitting. In case three, the learned patterns generalize the target distribution well. The improvement in training and validation is similar. This is the ideal case during the optimization process: the training direction is correct. Thus, we reduce the value of λ to continue in that direction.

Our dynamic regularization approach automatically diagnoses and corrects overfitting. Moreover, it encourages generalization by using a measure of generalization and specialization to weigh the loss of consistency.

5 Dataset and Experimental Settings

Datasets: We used two multimodal datasets, Moviescope (Cascante-Bonilla et al., 2019) and Multimodal IMDb (MM-IMDb) (Arevalo et al., 2017). The task is the multi-label classification of a movie's genre based on its plot and image poster. The performance metrics are macro and micro F1

[2] The code is available at https://github.com/IvonneMont/Dynamic-UDA-for-Transformers-in-Multimodal-Classification.git



By 2154, humans have depleted Earth's natural resources, leading to a severe energy crisis. The Resources Development Administration (RDA for short) mines for a valuable mineral – unobtanium – on Pandora, a densely forested habitable moon orbiting the gas giant Polyphemus in the Alpha Centauri star system.

By 2154, mammals have depleted Earth's natural energies, leading to a severe electricity recession. The Resources Advancement Administration (RDA for short) mines for a invaluable mineral – unobtanium – on Pandora , a densely forested livable moon orbiting the oil gigantic Polyphemus in the Alpha Centauri stardom mechanism.

Figure 2: Augmented examples using word replacement and RandAugment

for Moviescope and macro and micro average precision for MM-IMDb. Since both datasets have unbalanced labels, the most important metric is the micro. MM-IMDb is the largest dataset since its training partition is almost five times larger than the one for Moviescope. Moviescope has 13 movie genre categories, and MM-IMDb has 26 categories. These categories include genres such as action, animation, biography, comedy, crime, drama, family, fantasy, horror, and others.

Augmentation Strategies: For the text modality, we replace the 10% of tokens with the closest word using cosine similarity in W2V embeddings. For the image modality, we use the RandAugment technique. Rand Augment (Cubuk et al., 2020) applies N random transformations sequentially such as equalize, rotate, solarize, contrast, shear-x-y, translate-x-y, and others. We generate an augmented example for each instance in the training dataset as shown in Figure 2.

Implementation Details: We perform five experiments for each model, with the same hyperparameters but different seeds. We explore a learning rate of $1e-4$ and $5e-5$, batch size of 8,16, and 32, and a max number of epochs of 100 and 150. Also, we implement an early stopping with a patience number of 2 and 5. Finally, we use the best model selection on the validation partition to obtain the results in the test partition. We use two NVIDIA Tesla V100 32GB SXM2 cards for the experiments.

6 Results

This section presents the results obtained by the proposed model MMBV and the proposed framework dynamic UDA for the task of multi-label movie genre classification. We compare MMBV performance against its unimodal parts, ViT (Dosovitskiy et al., 2021) and BERT (Devlin et al., 2019). Also, we compare MMBV against Multimodal Bitransformer (MMBT) (Kiela et al., 2020), which reportedly holds the best-performing result in MM-IMDb dataset without a multimodal pertaining. For the second proposal, since the dynamic UDA framework does not depend on the model, we experiment with three unimodal and two multimodal models for the same task.

6.1 Evaluation proposed MMBV: CLS fusion

The purpose of the following experiments is to compare the performance of our MMBV model with the MMBT model. We also compare the performance of the unimodal models that compose each model to observe the individual performance of each modality.

Table 2 summarizes the results of MMBV applied on the datasets Moviescope and MM-IMDb. We compare with existing models ViT, ResNet 152, BERT, and MMBT. The evaluation metrics are micro/macro $F1$ for Moviescope and micro/macro AP for MM-IMDb. We show mean performance and standard deviation over five runs with different seeds. For the MM-IMDb dataset, we replicate the MMBT model used on Kiela et al. (2020) and obtain similar results.

The MMBV model outperforms MMBT by only +0.2. A possible cause is that the image model ViT has a lower performance on MM-IMDb than the ResNet 152. The convolutional network overcomes the vision transformer by 2.9%. However, when we fuse the modalities with MMBV, similar results are obtained for MMBT. This result indicates that the fusion of MMBV extracts useful information from the image more efficiently, without reducing the performance of text and increasing the global performance.

For the Moviescope dataset, there is no state-of-the-art method using only text and images. Since this dataset has more modalities, other authors have used other modalities such as audio and video, but we focus only on image and text only for this research. The MMBV model outperforms MMBT by +1.8. We find that ViT has a higher performance than ResNet 152. ViT improves over ResNet 152 by 6.4% on micro-F1. We observe that text is the dominant modality again, by almost 15% over the image. Another interesting observation is that MMBT is under its text part BERT by 0.2%. In this case, adding the image hurts global

Model	Modality	Moviescope $mAP/\mu AP$	MM-IMDB $mF1/\mu F1$
ResNet 152	Image	46.5±0.6/54.8±0.5	33.3±0.6/46.6±0.3
ViT	Image	52.2±0.3/61.2±0.4	32.2±0.4/43.7±0.3
BERT	Text	72.3±0.4/76.3±0.4	59.6±0.2/65.1±0.1
MMBT	Image & Text	72.1±0.5/76.1±0.4	61.4±0.3/66.6±0.3
MMBV	Image & Text	**74.0±0.2/77.9±0.4**	**61.7±0.5/66.8±0.4**

Table 2: Proposed MMBV results. Compared against ViT (Dosovitskiy et al., 2021), BERT (Devlin et al., 2019), and MMBT (Kiela et al., 2020). Moviescope is Macro-AP/Micro-AP; MM-IMDB is Macro-F1/Micro-F1.

performance.

The computational costs of MMBV and MMBT are almost the same. Both models have the same encoder for text but differ in their image encoder and fusion types. MMBT employs the ResNet 152 (He et al., 2015) encoder for image, which has fewer parameters than the ViT encoder used in MMBV. However, during fusion, the MMBV model requires no additional parameters, whereas the MMBT model modifies the dimensions and adds new positional image embeddings to represent the image.

6.2 Evaluation Dynamic Unsupervised Data Augmentation

The purpose of the following experiments is to compare the performance of our Dynamic UDA with the base model and the fixed (regular) UDA. Since UDA can be applied to any classification model, we compare the performance of the unimodal models and multimodal models.

Table 3 shows the results of the dynamic regularization for UDA, compared against the base model (Base), the original UDA (fixed), and standard augmentation (Aug). We consider three variants for the multimodal models, one where the text and image are augmented simultaneously (UDA dynamic), a second where only the image is augmented (UDA dynamic Image), and a third where only the text is augmented (UDA dynamic Text). The purpose of these variants is to observe how UDA affects the training speed of each modality.

Our Dynamic UDA obtained the best results when applied to all models across all the benchmarks. Specifically, Dynamic UDA outperforms ResNet 152-base by +0.9/0.1 on Moviescope and +1.2/2.3 on MM-IMDb; ViT-base by +1.2/1.0 on Moviescope and +2.1/3.8 on MM-IMDb; BERT by +0.4/0.3 on Moviescope and +1.0/1.4 on MM-IMDb; MMBT-base by +2.2/1.8 on Moviescope and +1.1/1.2 on MM-IMDb; MMBV-base by +1.5/1.5 on Moviescope and +1.0/1.4 on MM-IMDb. We observe that using the fixed UDA in some cases had lower performance than the base model. However, when we use the dynamic UDA, we obtain the best results. The dynamic framework has better performance than the fixed one.

	Model	Moviescope	MM-IMDb
ResNet 152	Base	50.1±0.6/58.9±0.5	33.3±0.6/46.6±0.3
	Aug	50.2±0.2/58.4±0.5	33.8±0.6/46.4±0.4
	UDA (fixed)	50.5±0.8/58.5±1.0	34.6±0.7/48.0±0.4
	UDA dynamic Image	**51.0±0.5/59.0±0.4**	**34.5±0.4/48.9±0.4**
ViT	Base	52.2±0.3/61.2±0.4	35.9±0.4/47.4±0.3
	Aug	47.8±2.6/55.3±0.7	36.3±0.3/48.1±0.3
	UDA (fixed)	47.1±1.3/54.5±0.7	37.8±0.5/48.2±0.7
	UDA dynamic Image	**53.4±0.5/62.2±0.6**	**38.0±0.7/51.2±0.2**
BERT	Base	72.3±0.4/76.3±0.4	59.6±0.2/65.1±0.1
	Aug	71.8±0.3/75.8±0.3	59.6±0.2/65.1±0.0
	UDA (fixed)	72.1±0.7/74.9±0.6	59.2±0.2/64.5±0.2
	UDA dynamic Text	**72.7±0.2/76.6±0.4**	**60.6±0.3/66.5±0.1**
MMBT	Base	72.1±0.5/76.1±0.4	61.4±0.3/66.6±0.3
	Aug	73.7±0.3/77.2±0.6	61.5±0.2/66.7±0.1
	UDA (fixed)	73.6±0.7/76.9±0.5	61.8±0.4/66.9±0.2
	UDA dynamic	73.6±0.2/77.4±0.3	62.0±0.2/67.5±0.2
	UDA dynamic Image	73.5±0.4/77.7±0.4	**62.5±0.2/67.8±0.1**
	UDA dynamic Text	**74.3±0.6/77.9±0.5**	62.2±0.2/67.6±0.2
MMBV	Base	74.0±0.2/77.9±0.4	61.7±0.5/66.8±0.4
	Aug	74.2±0.4/78.0±0.5	61.0±0.1/66.5±0.2
	UDA (fixed)	72.6±1.5/75.0±2.3	61.9±0.6/66.7±0.8
	UDA dynamic	**75.5±0.1/79.4±0.3**	**62.7±0.1/68.2±0.2**
	UDA dynamic Image	75.2±0.4/79.1±0.3	62.8±0.2/68.0±0.2
	UDA dynamic Text	75.3±0.8/79.2±0.5	62.7±0.3/67.9±0.3

Table 3: Supervised dynamic UDA results. Compared against the base model (Base), the original UDA (fixed), and standard augmentation (Aug). Three variants for the multimodal models: Dynamic UDA with image and text augmented (Dynamic UDA); with just the image augmented (UDA Dynamic Image); with just the text augmented(UDA dynamic text). Moviescope is Macro-AP/Micro-AP; MM-IMDb is Macro-F1/Micro-F1.

6.3 Effect of the regularization parameter λ

The regularization parameter λ controls the contribution of the consistency loss to the final loss. When the contribution of consistency loss is close to zero, the total loss function is almost equal to the supervised loss. So the performance is similar to the base model. The model is specialized in the training dataset but has low performance in the validation dataset, as Figure 3a shows. On the contrary, if the contribution of the consistency loss is big, the performance between the training dataset and validation is similar, but both have a low performance, as Figure 3b shows. Finally, Figure 3c shows the effect of using the proposed dynamic parameter. There is a balance between the specialization in the training dataset and the generalization for the validation dataset. The performance in the training dataset is high, and the overfitting is diminished.

We specifically compare the performance of multimodal models, MMBT and MMBV, using the base training and adding the framework dynamic UDA. Figure 4 shows the graph of the performance metric Micro AP during training. The dotted lines correspond to the base model and the continuous lines to the UDA dynamic framework. We observe that the gap between training and validation is smaller with Dynamic UDA. At first, the dynamic UDA keeps the performance of training and validation closer for more epochs. Dynamic UDA also takes more epochs to converge to the solution in comparison with the base model.

To observe how the ratio R from equation 10 controls the overfitting, we graph the cases described in Table 1 during training (see Figure 5). Case one is marked with red, case two with yellow, and case three with green. In this example, we observe that the beginning of the training corresponds to case two, where the patterns in the training dataset are learning very quickly, then the value of R is greater than one. By increasing the value of R it is possible to smooth the training curve for the following epochs (green zone). Later in epoch 10, there is a large gap between the validation and training curves, corresponding to case one. Thus in the following epochs, the ratio R is greater than one to control overfitting.

6.4 Multimodal models with a missing modality

To understand the behavior of the multimodal models for each modality, we evaluate the previously trained models in Moviescope, using only one modality. This means that for the multimodal models, the input for validation is only one modality, either only text or only image, although the training has been done with the two modalities. Table 4 shows the results. For the case of the MMBT model, we observe similar results when we use only the text as when we use both modalities. When we use only the image, the performance is poor. This means that the image information has a small contribution to the model MMBT. However, when we add the framework UDA the performance using only the image increases by 8.8%, and the performance using only text remains similar. This indicates that the improvement in models with both modalities relies on the better use of the image information.

For the MMBV model, with standard training, the performance drops when we use only one modality, either only text or only image. MMBV uses the joint information to make the prediction. When we add the Dynamic UDA framework, the model is more robust to the missing of one modality, especially when the missing modality is the image. We conclude that the MMBV model using Dynamic UDA improves the information learned from both modalities.

7 Conclusions

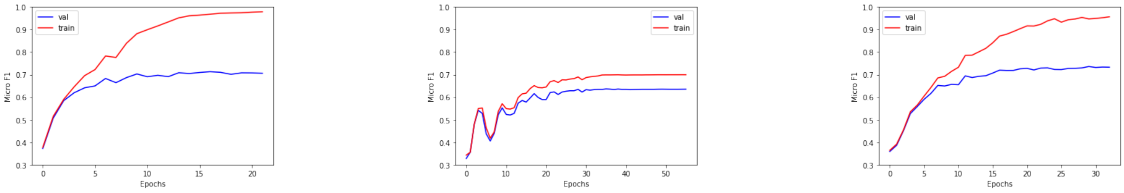
We developed two strategies to tackle key issues in multimodal learning for the text-image classification task. First, to address the lack of effective fusion and excessive computational complexity, we designed a text-image model MMBV for classification. MMBV has shown an effective fusion due to a similar use of the modalities and the CLS fusion. The second proposal, Dynamic UDA, focuses on reducing the gap between the learning rates of the different modalities and having an effective regularization that automatically identifies the overfitting and corrects it. We successfully modified the UDA framework by extending it to a multimodal supervised domain and overcoming the sensitivity of UDA to the choice of regulation parameter. Future work includes extending the model to other image-text tasks and more than two modalities and applying the Dynamic UDA to other general classification tasks. Since Dynamic UDA can be used to extract more information from the weaker modality in analogous scenarios, our work could have an impact on virtually any multimodal or multichannel application.

Limitations

The main limitation of the presented work is the need for significant computing resources to train multimodal models using Dynamic UDA. It should be noted that the proposed methods, MMBV and Dynamic UDA, require fewer computational resources than the original version of UDA.

Ethics Statement

While predicting film genre may seem similar to the task of age audience rating recommendation (e.g. MPAA film rating system), as some genres such as family, adventure, fantasy, animation, horror, or crime often have a similar rating, using this

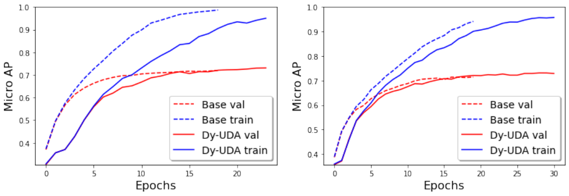


(a) $\lambda << 1$ overfitting (b) $\lambda >> 1$ underfitting (c) Dynamic λ balance

Figure 3: Visualization of the effect of λ in UDA

Dataset	Model	Modality	Base	Dynamic UDA	Dynamic UDA Text	Dynamic UDA Image
Moviescope	MMBT	Image	29.2±0.8/29.0±3.7	33.8±2.7/32.9±3.7	35.5±1.7/37.8±2.2	34.7±1.6/37.0±3.9
		Text	71.4±0.3/75.6±0.4	72.1±0.4/75.7±0.7	72.1±0.5/75.3±1.4	72.6±0.5/76.4±0.6
		Image-Text	72.1±0.5/76.1±0.4	73.6±0.2/77.4±0.3	**74.3±0.3/78.1±0.4**	73.8±0.1/77.9±0.3
	MMBV	Image	43.4±0.9/50.3±2.7	44.3±1.8/52.2±2.0	45.6±2.4/53.4±2.0	43.1±1.5/50.2±2.8
		Text	62.3±4.6/54.7±4.4	65.6±1.9/62.5±5.1	68.8±1.2/65.3±3.8	67.4±1.8/65.1±5.3
		Image-Text	74.0±0.2/77.9±0.4	**75.5±0.3/79.5±0.3**	75.3±0.8/79.2±0.5	75.2±0.4/79.1±0.3
MM-IMDb	MMBT	Image	16.5±1.9/29.4±1.0	16.7±1.5/32.9±1.0	22.5±1.9/35.7±1.8	18.6±0.5/33.6±1.59
		Text	56.6±0.7/62.8±1.4	59.9±0.3/65.3±0.5	58.0±0.6/64.3±1.5	59.2±0.5/65.2±0.8
		Image-Text	61.4±0.3/66.6±0.3	62.0±0.2/67.5±0.2	62.2±0.2/67.6±0.2	**62.5±0.2/67.8±0.1**
	MMBV	Image	22.8±2.6/39.4±1.8	21.9±1.1/39.9±1.4	23.6±2.9/39.9±2.1	22.6±2.5/38.9±1.4
		Text	36.2±9.7/45.0±8.2	47.8±3.3/55.5±2.6	42.1±4.1/49.0±3.8	43.8±6.8/52.2±5.3
		Image-Text	62.7±0.3/67.9±0.3	**62.7±0.1/68.2±0.2**	62.7±0.3/67.9±0.3	62.8±0.2/68.0±0.2

Table 4: Multimodal models evaluated over only one modality. Moviescope is Macro-AP/Micro-AP. MM-IMDb is Macro-F1/Micro-F1



(a) MMBT (b) MMBV

Figure 4: A comparison between the base model and Dynamic UDA framework during training. The gap between training and validation is smaller with Dynamic UDA, as it takes more epochs to converge to the solution in comparison with the base model.

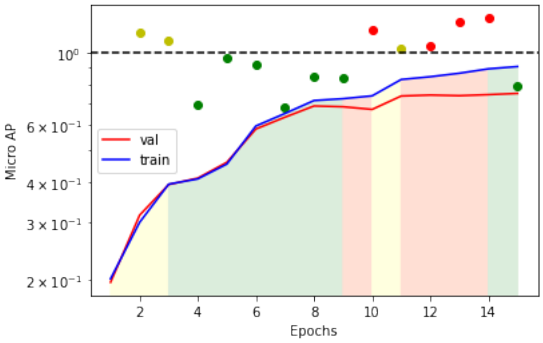


Figure 5: Effect of the ratio R (equation 10) according to the cases in Table 1. Case one is marked with red color, case two with yellow, and case three with green. The R value is represented in the colored dots.

model to directly guide the rating task may not be appropriate as genre and rating are not always correlated.

Additionally, note that the model has been trained primarily on the Western film industry (mainly American), and our data lacks representation from other significant industries, such as those in Asia. However, with appropriate training data, these models could perform well in these cultural contexts. Additionally, we encourage the ethical use of these models in other multimodal tasks with sensitive contexts.

Acknowledgements

This research was funded by *Consejo Nacional de Ciencia y Tecnología* (CONACyT) master's degree grant #1076977. The authors thank to CONACyT, CIMAT and *Instituto Nacional de Astrofísica, Óptica y Electrónica* (INAOE) for the computer resources provided through the INAOE Supercomputing Laboratory's Deep Learning Platform for Language Technologies (*Plataforma de aprendizaje profundo para tecnologías del lenguaje*) and CIMAT Bajio Supercomputing Laboratory (#300832). Sanchez-Vega acknowledges CONACyT for its support through the program *"Investigadoras e Investigadores por México"* (Project ID.11989, No.1311). Finally, we thank Asa Ferguson from Reed College for re-

viewing our work.

References

Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. 2019. Self-supervised learning by cross-modal audio-video clustering.

John Arevalo, Thamar Solorio, Manuel Montes y Gómez, and Fabio A. González. 2017. Gated multimodal units for information fusion.

Hangbo Bao, Li Dong, and Furu Wei. 2021. Beit: Bert pre-training of image transformers.

Paola Cascante-Bonilla, Kalpathy Sitaraman, Mengjia Luo, and Vicente Ordonez. 2019. Moviescope: Large-scale analysis of movies using multiple modalities.

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

Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. 2020. Randaugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems*, volume 33, pages 18613–18624. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale.

Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. 2020. Large-scale adversarial training for vision-and-language representation learning.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition.

Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, Ethan Perez, and Davide Testuggine. 2020. Supervised multimodal bitransformers for classifying images and text.

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

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2021. Training data-efficient image transformers & distillation through attention.

Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J. Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Multimodal transformer for unaligned multimodal language sequences.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Weiyao Wang, Du Tran, and Matt Feiszli. 2020. What makes training multi-modal classification networks hard?

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. 2020. Unsupervised data augmentation for consistency training.

Zitong Yu, Benjia Zhou, Jun Wan, Pichao Wang, Haoyu Chen, Xin Liu, Stan Z. Li, and Guoying Zhao. 2021. Searching multi-rate and multi-modal temporal enhanced networks for gesture recognition. *IEEE Transactions on Image Processing*, 30:5626–5640.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
In Limitations section

☑ A2. Did you discuss any potential risks of your work?
In Ethics Statement section

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1 Introduction

☑ A4. Have you used AI writing assistants when working on this paper?
Google engine, Google docs, grammarly, google translate, chatgpt to get sinomis or rephrase. All assistants were used for "Assistance purely with the language of the paper", strictly used for polish the author's original content.

B ☑ Did you use or create scientific artifacts?

2 proposed methods, 2 previos proposed methods, 2 previos colleted data

☑ B1. Did you cite the creators of artifacts you used?
section 2,3,4 and 5

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Ethics Statement section

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
the conditions of use are the same and these are established in their licensing sheets in their repositories

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
section 5

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

sections 5 and 6

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
The exact number is not mentioned but it compares with previous models.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
sections 5 and 6

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
sections 6 (the range of the standard deviation of the evaluation metric in 5 runs)

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Sections 2, 3, 4, 5 and 6

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.