$\mathcal{P}$owMix: A Versatile Regularizer for Multimodal Sentiment Analysis

Efthymios Georgiou, *Graduate Student Member, IEEE,* Yannis Avrithis, *Senior Member, IEEE,* and Alexandros Potamianos, *Fellow, IEEE*

***Abstract*—Multimodal sentiment analysis (MSA) leverages heterogeneous data sources to interpret the complex nature of human sentiments. Despite significant progress in multimodal architecture design, the field lacks comprehensive regularization methods. This paper introduces $\mathcal{P}$owMix, a versatile embedding space regularizer that builds upon the strengths of unimodal mixing-based regularization approaches and introduces novel algorithmic components that are specifically tailored to multimodal tasks. $\mathcal{P}$owMix is integrated before the fusion stage of multimodal architectures and facilitates intra-modal mixing, such as mixing text with text, to act as a regularizer. $\mathcal{P}$owMix consists of five components: 1) a varying number of generated mixed examples, 2) mixing factor reweighting, 3) anisotropic mixing, 4) dynamic mixing, and 5) cross-modal label mixing. Extensive experimentation across benchmark MSA datasets and a broad spectrum of diverse architectural designs demonstrate the efficacy of $\mathcal{P}$owMix, as evidenced by consistent performance improvements over baselines and existing mixing methods. An in-depth ablation study highlights the critical contribution of each $\mathcal{P}$owMix component and how they synergistically enhance performance. Furthermore, algorithmic analysis demonstrates how $\mathcal{P}$owMix behaves in different scenarios, particularly comparing early versus late fusion architectures. Notably, $\mathcal{P}$owMix enhances overall performance without sacrificing model robustness or magnifying text dominance. It also retains its strong performance in situations of limited data. Our findings position $\mathcal{P}$owMix as a promising versatile regularization strategy for MSA. Our code is made available[1].**

***Index Terms*—Multimodal Learning, Regularization, Multimodal Sentiment Analysis, intra-modal mixing**

I. Introduction

Multimodal Sentiment Analysis (MSA) is the task of enriching a computer system with affective understanding of real-world human-centric video segments. Interpreting sentiments from videos is very challenging due to the multifaceted nature of human communication through speech, facial expressions, linguistic content, etc. [1]. The practical applications of MSA are numerous in the digital era and vary from *human-computer interaction* (HCI) [2] and healthcare [3], to opinion mining in reviews [4] and education [5]. Despite the advancements in the MSA field [6]–[12], developing an end-to-end system that effectively analyzes the complex aspects of human sentiment remains an open research challenge.

Building on the idea that multimodal learning includes unimodal elements, such as decomposing multimodal predictions into separate unimodal contributions and multimodal interactions [13], we perceive multimodal tasks as being fundamentally more complex than unimodal. This suggests that challenges inherent in unimodal setups, such as overfitting, also exist in multimodal scenarios. Moreover, the presence of multiple dimensions of heterogeneity across various modalities, such as information and noise, the spectrum of internal structures from natural to symbolic signals, and the distributional gap between modalities, introduces significant complexities [14]. It is found, that learning multimodal representations poses unique challenges compared to the unimodal case [15], [16].

Despite expectations that multimodal networks would outperform their unimodal counterparts [17], this is not consistently observed. It is found that different inputs generalize at different rates, leading to unexpected performance degradation [15], as well as a tendency of networks to over-rely on dominant modalities [16], *e.g.*, text in MSA [18]. Furthermore, studies demonstrate that joint multimodal training tends to learn a limited spectrum of features and modalities [19], resulting in suboptimal solutions.

Considering the challenges in multimodal learning, it is reasonable to speculate that regularization and data augmentation methods may be beneficial, similar to unimodal tasks. Nevertheless, the existing literature on this topic remains relatively sparse and of limited scope. Some approaches like Wang et al. [15] and Wu et al. [16], propose to dynamically reweight unimodal loss terms within the overall learning objective, while Du et al. [20] suggest leveraging a unimodal teacher model to improve learning of unimodal features. However, these methods are tied with specific learning hypotheses and late fusion architectures, which constrains their broader applicability.

For more advanced models, like those employed for MSA, Liu et. al. [21] introduce a learnable auto-encoder for embedding augmentation within multimodal networks, and M3 [22] utilizes intense text masking in the latent unimodal space before fusion, acting as a regularizer. More closely aligned with our work, AV-MC [23] employs MixUp [24] independently for acoustic and visual streams, when labels are available for each modality, requiring three separate forward propagations,

Efthymios Georgiou is with the School of ECE, National Technical University of Athens, Athens, Greece, and the Instituite for Speech and Language Processing (ILSP), Athena Research Center, Athens, Greece (E-mail: efthygeo@mail.ntua.gr) Yannis Avrithis is with the Institute of Advanced Research on Artificial Intelligence (IARAI), Vienna, Austria (E-mail: yannis@avrithis.net) Alexandros Potamianos is with the School of ECE, National Technical University of Athens, Athens, Greece (E-mail: potam@central.ntua.gr)

[1] **https://github.com/efthymisgeo/powmix**

one for the original input and one for each unimodal set of mixed examples. However, all these approaches are bound to specific architectural designs, fusion strategies and learning assumptions, which restrict their applicability. Therefore, a broad-spectrum regularization framework that transcends such constraints is crucial for multimodal learning environments such as MSA.

In this work, we introduce $\mathcal{P}$owMix, a novel regularization method specifically crafted to improve regularization in multimodal scenarios and in particular MSA. Unlike methods designed to handle modality-specific challenges, $\mathcal{P}$owMix aims to offer a broad-spectrum solution applicable across a range of datasets and model architectures. As a member of the mixing algorithm family, it is inspired by methods like MixUp [24], TransMix [25] and MultiMix [26], known for their regularization capabilities. $\mathcal{P}$owMix is integrated before the fusion stage in the multimodal architecture and facilitates intra-modal mixing, *e.g.*, text with text and audio with audio.

What sets $\mathcal{P}$owMix apart is its novel components that, in synergy, render it suitable for multimodal contexts. In particular, the algorithm is characterized by five key features: 1) a *varying number of generated mixed examples*, 2) *mixing factor reweighting* to encapsulate the importance of each modality for each mini-batch example, 3) *anisotropic mixing* for independent mixing across unimodal spaces, 4) *dynamic mixing*, a novel element for representation mixing, and 5) *cross-modal label mixing*, a new way to aggregate mixed labels across modalities. $\mathcal{P}$owMix emerges as a versatile regularizer across MSA datasets and architectures. To the best of our knowledge, such an approach has not been previously reported in the literature.

To establish the effectiveness of $\mathcal{P}$owMix, we conduct experiments on three widely used MSA benchmark datasets: MOSI [27], MOSEI [28], and SIMS [29]. We also employ three different multimodal architectures: MulT [9], MISA [10], and Self-MM [11]. These models are chosen because they perform well and, most importantly, cover a wide range of architectural designs and learning approaches.

Our main contributions are summarized as follows:

1) We introduce $\mathcal{P}$owMix, a novel regularization method applied to MSA. It consists of five key components, two of which build upon existing ideas and the other three are entirely novel: anisotropic mixing, dynamic mixing, and cross-modal label mixing.
2) We experimentally validate the effectiveness of $\mathcal{P}$owMix across diverse MSA datasets and architectures, confirming superior performance over baselines and state-of-the-art mixing methods.
3) We conduct an in-depth ablation study of the features of $\mathcal{P}$owMix, demonstrating the contribution of each component to the overall performance. We also highlight the synergetic impact of these features.
4) We present a comprehensive algorithmic analysis, demonstrating the behavior of $\mathcal{P}$owMix across different fusion types, its robustness to noise and text dominance levels, as well as its efficacy under limited data scenarios.

The paper is structured as follows: section II covers related work and section III provides formulation and background for our study. Next, section IV details the $\mathcal{P}$owMix algorithm, while section V outlines our experimental setup. The core experimental results, as well as the ablation study and algorithmic analysis, are presented in section VI. Finally, section VII draws conclusions and discusses future research directions.

II. Related Work

This section provides an overview of the literature, beginning with an exploration of works in the MSA field, which is the core of our experimentation. We then discuss advancements in mixing techniques within the unimodal learning context, highlighting their components. Finally, we present existing multimodal regularizers and highlight their task and problem specific characteristics.

A. Multimodal sentiment analysis

MSA research mainly focuses on building better fusion schemes and utilizing diverse learning recipes to enhance representation learning for the task at hand. In particular, TFN [8] employs outer product of unimodal representations to capture cross-modal interactions. Poria *et al*. [30] and Gu *et al*. [31] implement multi-level and hierarchical attention to better contextualize information. DHF [32] applies a hierarchical fusion mechanism across different levels within the architecture.

Other types of neural structures employed in MSA include neural memory modules [28], capsule networks [33], and graph neural networks [34]. Tsai *et al*. [9] utilize transformers, where cross-attention blocks act as early fusion and concatenation serves as late fusion. Rahman *et al*. [35] fine-tune a pre-trained BERT [36] model by incorporating a multimodal shifting layer as early fusion and Guo *et al*. [37] build upon this idea and use a multimodal interaction layer prior to the language model. Wang *et al*. [38] use text-enhanced cross-attentive transformer blocks to promote the linguistic information and Zhang *et al*. [39] use language-guided fusion along with a fused hypermodality. In CENet [40] authors exploit a pretrained language model tailored towards sentiment analysis instead of BERT.

Another line of work utilizes more complex learning recipes such as canonical correlation analysis [12] and cycle-consistency loss [41] across modalities. Coupling different learning recipes with pre-trained models has been a popular choice among researchers. Yu *et al*. [11] introduce a unimodal pseudo-labeling module that backpropagates three additional losses. Hazarika *et al*. [10] augment the learning objective with feature reconstruction loss as well as attracting and repelling objectives and Yang *et al*. [42] learn a common space as well as a private space for each of the involved modalities.

A two-step hierarchical learning recipe based on mutual information maximization is proposed in [43], while Sun *et al*. [12] propose a meta-learning framework that learns each unimodal network and then adapts them for the MSA task. Sun *et al*. [44] propose a transformer architecture leveraging dual-level reconstruction loss and an attraction loss in a Siamese

setup between complete and incomplete data. NIAT [45] learns a unified joint representation between clean and noisy data by coupling masking-based feature augmentation with an adversarial training strategy. Hu *et al*. [46] employ a text encoder-decoder architecture, using T5 [47], and implement a contrastive loss among unimodal encoders. The decoder generates text sequences, which are decoded into MSA-related info such as polarity. Notably, none of the aforementioned approaches handles multimodal regularization.

B. Mixing in unimodal learning setups

Regularization[2] in unimodal learning setups has been extensively studied. We primarily focus on techniques that modify the learning process through mixing-based algorithms such as MixUp [24]. These algorithms are notable for their regularization benefits and a unique capability to mix representations in the latent space. This feature is particularly desired for the development of broad multimodal regularizers.

Input mixing: Algorithmic approaches in this category are usually attached to specific types of data. For *computer vision* (CV), the most studied field, options have evolved from fundamental transformations, *e.g*., translation and rotation [48], to more advanced methods. These include MixUp [24] and CutMix [49], which mix pairs of images in the pixel and label space, as well as AutoMix [50] that introduces an automatic mixing framework. For a unified study on vision mixing techniques, we refer to [51].

In *natural language processing* (NLP), mixing words in their raw format is not straightforward, leading to approaches like SSMix [52], which substitutes salient parts of a sentence with words from another. In the Audio and Speech domain, SpecMix [53] mixes two spectrogram representations w.r.t. the frequency domain. Our work shares a similar idea with TransMix [25], a state-of-the-art CV technique, proposing pixel-wise reweighting of mixing factors based on their attention map values.

Latent space mixing: Algorithms in this category focus on manipulating latent representations. Manifold MixUp [54] interpolates pairs of hidden representations along with their labels. Non-Linear MixUp [55] extends this concept with a non-linear interpolation scheme in the text embedding space. ReMix [56] favors the minority class during mixing. Further expanding on these ideas, SpeechMix [57] and MixUp-Transformer [58], are variants of Manifold MixUp for speech and NLP tasks respectively.

Closely related to our work is MultiMix [26], a state-of-the-art method that mixes all representations within a batch to generate many mixtures. MultiMix also incorporates the idea of reweighting interpolation factors prior to mixing. Similarly, $\mathcal{P}$owMix proposes a methodology to generate more mixed examples than the mini-batch size, yet by interpolating fewer examples than MultiMix. These algorithms, especially MultiMix, represent a methodological shift towards more abstract regularization methods and motivate the need for techniques beyond unimodal boundaries.

[2]In this work, we attribute regularization as any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error [48].

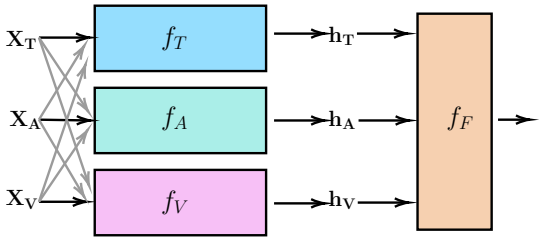

Fig. 1. *Abstract multimodal fusion scheme* for MSA architectures including MulT [9], MISA [10] and Self-MM [11]. Independent processing pathways for each modality, represented by *encoders* f_m, where $m \in \{T, A, V\}$ is the modality (T: text, A: acoustic, V: video). In models like MulT, encoders can incorporate other modalities as inputs too, *i.e*., early-fusion. The hidden representations $\mathbf{h}_m$ extracted by the encoders are fed to the *fusion network* f_F, which generates the final prediction. Depending on the architecture, f_F can manifest as a non-linear feedforward network (MulT), a single-layer transformer (MISA), a dual linear layer setup (Self-MM), *etc*. This scheme abstracts away components of the architecture not directly related to the prediction task. Mixing is performed directly on $\mathbf{h}_m$.

C. Regularization in multimodal setups

Here, we position the proposed $\mathcal{P}$owMix in the context of existing multimodal learning regularization techniques. Current methods often target specific challenges [15], [20] or are confined to particular inputs and domains. For example, MixGen [59] combines image and text data intra-modally but is primarily helpful for tasks like retrieval and captioning. Similarly, *cross modal CutMix* (CMC) [60] bridges unpaired image-text datasets cross-modally but has limited architecture and task flexibility. By contrast, $\mathcal{P}$owMix is designed to adapt across a wide range of supervised multimodal classification or regression problems without being limited to specific input types or architectures.

The embedding space augmentation technique proposed in [21] marks progress in multimodal regularization. However, its complexity limits its application, involving additional learnable parameters, doubled forward propagations, adversarial-like optimization, and manual output thresholding. Similarly, AV-MC [23], which utilizes MixUp for acoustic and visual streams independently, requires unimodal labels and three forward propagations. $\mathcal{P}$owMix, on the other hand, offers versatility and broad applicability by not posing any constraints on the learning setup, particularly in complex tasks like MSA.

III. Background

We formulate *multimodal sentiment analysis* (MSA) as a multimodal fusion task and present an abstract architecture scheme that encapsulates most existing approaches, as illustrated in Figure 1. We then describe unimodal mixing algorithms with a particular focus on Manifold MixUp [54] and MultiMix [26]. We also briefly outline the idea of *reweighting*, a mechanism used in different mixing algorithms [25].

A. Preliminaries and notation

Figure 1 shows an abstract multimodal fusion architecture used in MSA models, *e.g*., [8]–[11], [22]. We have omitted parts of the architecture not directly involved in the predictive

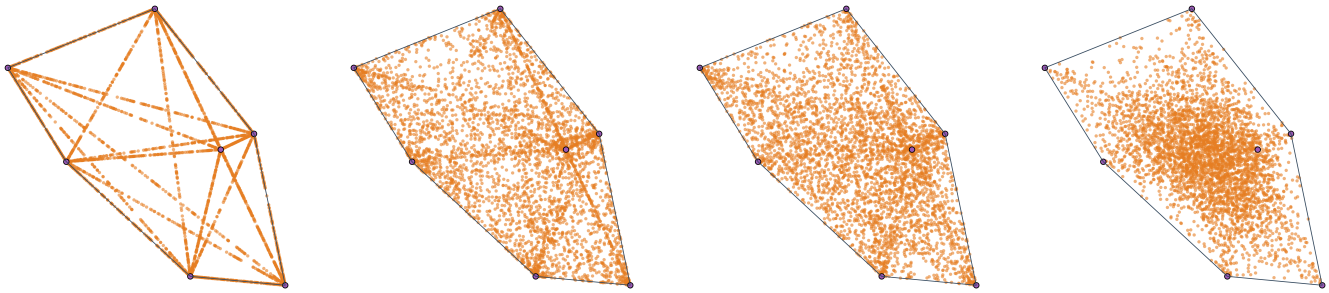

Fig. 2. Motivation of $\mathcal{P}$owMix. Given seven 2-dimensional points, we generate 2^{12} points, each as a convex combination of a random subset of n_I points. From left to right, we interpolate a) pairs, b) triplets, c) quadruplets, and d) all seven points. MixUp (a) interpolates two points at a time, while MultiMix (d) uses all seven points. By *dynamic mixing*, $\mathcal{P}$owMix randomly samples a subset of different cardinality for each generated point, hence can provide mixing instances between interpolation of pairs and all points. Empirically, we find that mixing from 2 to 4 points gives good performance. See Fig.3 for detailed results. ●: Original examples; ●: interpolated examples; ——: convex hull.

information flow, such as the decoder in MISA [10] and the unimodal label generation module in Self-MM [11]. Our focus is on the input modality encoders f_m and the fusion network f_F alone.

Each unimodal input space is denoted as $\mathcal{X}_m$, where m is the modality, from a set of indices $\mathcal{M} = \{1, \ldots, M\}$. The multimodal input space is denoted by the Cartesian product $\mathcal{X}_\mathcal{M} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_M$. The multimodal input data is given as the collection $\mathcal{D} = \{\mathbf{X}^{(i)}, y^{(i)}\}_{i=1}^N$, where $\mathbf{X}^{(i)} = (\mathbf{X}_1^{(i)}, \ldots, \mathbf{X}_M^{(i)})$ is a multimodal m-tuple with each $\mathbf{X}_m^{(i)} \in \mathcal{X}_m$, $y^{(i)} \in \mathbb{R}$ is the corresponding label for regression tasks and N is the number of multimodal m-tuples. When discussing unimodal concepts, such as Manifold MixUp, we omit the m subscript.

Each input modality encoder is a mapping $f_m : \mathcal{X}_\mathcal{M} \to \mathbb{R}^{d_m}$, where d_m is the hidden space dimension. Given multimodal input $\mathbf{X} \in \mathcal{X}_\mathcal{M}$, the output of each encoder is $\mathbf{h}_m = f_m(\mathbf{X}) \in \mathbb{R}^{d_m}$, allowing for early fusion in general. The fusion network is a mapping $f_F : \mathbb{R}^{d_1} \times \cdots \times \mathbb{R}^{d_M} \to \mathbb{R}$. The complete prediction scheme f is the composition of input modality mappings with the fusion map, *i.e.*, $f(\mathbf{X}) = f_F(\mathbf{h}_1, \ldots, \mathbf{h}_M) = f_F(f_1(\mathbf{X}), \ldots, f_M(\mathbf{X}))$. All mixing operations are applied to the hidden representations of an entire mini-batch, denoted by $\mathbf{H}_m \in \mathbb{R}^{B \times d_m}$, where B is the mini-batch size.

B. Manifold MixUp

Manifold MixUp [54] is a unimodal regularization method that interpolates pairs of hidden representations from different input examples using an interpolation factor $\lambda \in [0, 1]$. Given a pair of representations $\mathbf{h}^{(i)}, \mathbf{h}^{(j)}$ in latent unimodal space $\mathbb{R}^d$ and their labels $y^{(i)}, y^{(j)}$, mixing is performed by the convex combination

$$\widetilde{\mathbf{h}} = \lambda \mathbf{h}^{(i)} + (1 - \lambda)\mathbf{h}^{(j)} \tag{1}$$

$$\widetilde{y} = \lambda y^{(i)} + (1 - \lambda) y^{(j)}. \tag{2}$$

Following standard practice, λ is sampled from distribution $\mathrm{Beta}(\alpha)$, where $\alpha = 1$. As shown in 2a, this method results in mixed embeddings along linear segments between pairs of examples.

C. MultiMix

MutiMix [26], a state-of-the-art mixing method, is incorporated in our study for comparative analysis with our proposed $\mathcal{P}$owMix. Unlike Manifold MixUp, which interpolates pairs of examples, MultiMix interpolates all examples in a batch and generates a number of mixed examples that is independent of the mini-batch size. In particular, given a batch of size B, MultiMix mixes all B hidden representations, denoted by $\mathbf{H} \in \mathbb{R}^{B \times d}$, to generate n_O mixed examples. This involves randomly sampling n_O interpolation vectors in $\mathbb{R}^B$ from Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\alpha})$, resulting in an interpolation matrix $\boldsymbol{\Lambda} \in \mathbb{R}^{n_O \times B}$. Mixing is then performed by the convex combination

$$\widetilde{\mathbf{H}} = \boldsymbol{\Lambda}\mathbf{H} \in \mathbb{R}^{n_O \times d} \tag{3}$$

$$\widetilde{\mathbf{y}} = \boldsymbol{\Lambda}\mathbf{y} \in \mathbb{R}^{n_O}, \tag{4}$$

where $\mathbf{y} \in \mathbb{R}^B$ denotes the labels of the mini-batch. Parameter α is sampled from a uniform distribution $U[0.5, 2.0]$ by default, which produces mixed examples as shown in 2d. The number n_O of mixed examples is a tunable hyperparameter. In $\mathcal{P}$owMix, we adopt this idea of n_O being decoupled from the mini-batch size.

D. Mixing factor reweighting

Reweighting the mixing factors based on representations has proven an effective strategy in various MixUp variants, such as TransMix and MultiMix [25], [26]. This idea is founded on the principle that reweighting different elements by assigning them attention-like values can result in more effective mixtures. The key to this approach is a mapping g, which determines the attention weights of mixed elements. This function is based on attention maps obtained by transformer architectures in TransMix [25] and a cross-attention operation over dense features in MultiMix [26]. To keep our approach generic in terms of architecture, we rather use a simpler mapping g in $\mathcal{P}$owMix.

IV. $\mathcal{P}$owMix

Our proposed multimodal regularization method, $\mathcal{P}$owMix, is described here in detail. It consists of five key elements:

1) Generating a *varying number of mixed examples*.
2) *Mixing factor reweighting*, adjusting the contribution of each representation in a mixed example.
3) *Anisotropic mixing*, *i.e.*, sampling distinct mixing factors for each latent modality space.
4) *Dynamic mixing*, allowing the combination of a variable number of embeddings from the mini-batch.
5) *Cross-modal label mixing*, creating a unified multimodal label for each mixed multimodal tuple.

While the first two elements build upon established concepts in the literature (see subsection III-C and subsection III-D respectively), the other three are entirely novel contributions, specifically designed to make $\mathcal{P}$owMix perform best in a multimodal setup. In the following, we provide a detailed account of how $\mathcal{P}$owMix combines these novel elements in a multi-step algorithm, and then we discuss their motivation and impact. We give pseudo-code for $\mathcal{P}$owMix in algorithm 1.

Algorithm 1: $\mathcal{P}$owMix multimodal mixing.

Input: hidden representations $\mathbf{H}_m \in \mathbb{R}^{B \times d_m}$
Input: label vector $\mathbf{y} \in \mathbb{R}^B$
Input: number n_O of generated mixed examples
Output: mixed representations $\widetilde{\mathbf{H}}_m \in \mathbb{R}^{n_O \times d_m}$
Output: mixed label vector $\widetilde{\mathbf{y}} \in \mathbb{R}^{n_O}$

$\mathbf{a}_m \leftarrow g_A(\mathbf{H}_m)$ ▷ attention vector
$\boldsymbol{\alpha}_m \sim U(0.5, 2)$ ▷ interpolation hyperparameter
$\boldsymbol{\Lambda}_m \sim \mathrm{Dir}(\boldsymbol{\alpha}_m)$ ▷ interpolation matrix
$\mathbf{P} \sim U(2, 4)$ ▷ masking hyperparameter
$\mathbf{M} \sim \mathrm{Bern}(\mathbf{P}/B)$ ▷ mask sampling
$\widetilde{\boldsymbol{\Lambda}}_m \leftarrow \eta_1(\mathbf{a}_m^\top \odot \mathbf{M} \odot \boldsymbol{\Lambda}_m)$ ▷ normalization
$\widetilde{\mathbf{H}}_m \leftarrow \widetilde{\boldsymbol{\Lambda}}_m \mathbf{H}_m$ ▷ representation mixing
$\widetilde{\mathbf{y}}_m \leftarrow \widetilde{\boldsymbol{\Lambda}}_m \mathbf{y}_m$ ▷ label mixing
$\widetilde{\mathbf{y}} \leftarrow \frac{1}{M} \sum_m \widetilde{\mathbf{y}}_m$ ▷ cross modal label mixing

A. Algorithm

Given a multimodal example $\mathbf{X}^{(i)} = (\mathbf{X}_1^{(i)}, \ldots, \mathbf{X}_M^{(i)}) \in \mathcal{X}_\mathcal{M}$, we obtain a hidden representation $\mathbf{h}_m^{(i)} = f_m(\mathbf{X}^{(i)}) \in \mathbb{R}^{d_m}$ as the output of encoder f_m for each modality $m \in \mathcal{M}$. At the mini-batch level, let the matrix $\mathbf{H}_m \in \mathbb{R}^{B \times d_m}$ hold the hidden representations of all examples in its rows, where B is the mini-batch size. Let also vector $\mathbf{y} \in \mathbb{R}^B$ denote the labels $y^{(i)} \in \mathbb{R}$ for all examples of the mini-batch.

***Mixing factor reweighting**:* First we compute the *mixing factor attention weights* $\mathbf{a}_m \in \mathbb{R}^B$. Specifically, as a form of attention, we use average pooling over the feature dimension followed by ReLU and normalization across modalities:

$$\mathbf{a}_m = g(\mathbf{H}_m) := \frac{\sigma(\mathbf{H}_m \mathbf{1}_{d_m}/d_m)}{\sum_{m' \in \mathcal{M}} \sigma(\mathbf{H}_{m'} \mathbf{1}_{d_{m'}}/d_{m'})}, \tag{5}$$

where $\mathbf{1}_{d_m} \in \mathbb{R}^{d_m}$ is an all-ones vector, $\sigma(\cdot)$ is the ReLU function and division is performed element-wise. This operation is similar to transformer cross-attention between query $\mathbf{1}_{d_m}$ and key $\mathbf{H}_m$ but here normalization is performed across modalities. We call the use of weights $\mathbf{a}_m$ *mixing factor reweighting*. A baseline is to use a uniform $\mathbf{a} = \mathbf{1}_B/M$ for all modalities.

***Anisotropic mixing**:* For each modality, we then sample a different *mixing matrix* $\boldsymbol{\Lambda}_m \in \mathbb{R}^{n_O \times B}$. To do this, we sample n_O distinct B-dimensional interpolation coefficient vectors from a Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\alpha}_m)$, with its parameter $\boldsymbol{\alpha}_m \in \mathbb{R}^{n_O}$ drawn from a uniform distribution $U(0.5, 2.0)$, following [26]. Let's assume that the latent representation of each modality is embedded in separate subspaces of an overall multimodal latent representation space. Since we use a different mixing matrix per subspace, we call this approach *anisotropic mixing*. A simpler alternative is to use the same $\boldsymbol{\Lambda} \in \mathbb{R}^{n_O \times B}$ for all modalities.

***Dynamic mixing**:* Next, we randomly mask mixing factors to keep only a small number of nonzero elements per row of $\boldsymbol{\Lambda}_m$. This limits the interpolation to just a few examples. Specifically, we sample a binary mask $\mathbf{M} \sim \mathrm{Bern}(\mathbf{P}/B) \in \mathbb{R}^{n_O \times B}$ from a Bernoulli distribution and apply it element-wise to the mixing matrix $\boldsymbol{\Lambda}_m$.

The hyperparameter $\mathbf{P} \in \mathbb{R}^{n_O \times B}$ provides a different value for each element of $\mathbf{M}$ and dictates the proportion of mini-batch examples to interpolate in the representation space. In general, we sample $\mathbf{P}$ from $U(a, b)$, which means that we mix between a and b examples. We find empirically that $\mathbf{P} \sim U(2, 4)$ works well, corresponding to 2 to 4 nonzero elements on average per row of $\boldsymbol{\Lambda}_m$.

Since different examples are interpolated for each generated example, we call this process *dynamic mixing*. Importantly, the same binary mask $\mathbf{M}$ is used for all modalities m by default, a choice called *mask sharing*. An alternative is to use a different $\mathbf{M}_m \in \mathbb{R}^{n_O \times B}$ for each modality. The simplest approach would be not to use any masking, *i.e.* setting $\mathbf{M} = \mathbf{1}_{n_O \times B}$, which deactivates dynamic mixing.

***Interpolation operation**:* Given the mixing factor attention weights $\mathbf{a}_m \in \mathbb{R}^B$ and the binary mask $\mathbf{M} \in \mathbb{R}^{n_O \times B}$, we multiply element-wise with the mixing matrix $\boldsymbol{\Lambda}_m \in \mathbb{R}^{n_O \times B}$ and re-normalize over the mini-batch dimension to obtain the *reweighted mixing matrix*

$$\widetilde{\boldsymbol{\Lambda}}_m = \eta_1(\mathbf{a}_m^\top \odot \mathbf{M} \odot \boldsymbol{\Lambda}_m) \in \mathbb{R}^{n_O \times B}, \tag{6}$$

where η_1 here denotes ℓ_1-normalization of rows and $\odot$ is Hadamard product (with vectors broadcasting to matrices as needed). In this product, attention weights $\mathbf{a}_m$ are scaling the columns of the mixing matrix, while the binary mask $\mathbf{M}$ is selecting subsets from each row.

Now, using $\widetilde{\boldsymbol{\Lambda}}_m$, we generate n_O mixed examples per modality. In particular, given the hidden representations $\mathbf{H}_m$ and the labels $\mathbf{y}$, we perform intra-modal interpolation for modality m by

$$\widetilde{\mathbf{H}}_m = \widetilde{\boldsymbol{\Lambda}}_m \mathbf{H}_m \in \mathbb{R}^{n_O \times d_m} \tag{7}$$

$$\widetilde{\mathbf{y}}_m = \widetilde{\boldsymbol{\Lambda}}_m \mathbf{y} \in \mathbb{R}^{n_O}. \tag{8}$$

This is similar with (3), (4), but $\widetilde{\boldsymbol{\Lambda}}_m$ is reweighted, masked and unique for each modality. Formally, it expresses convex combinations of different input examples in the representation space and their labels, due to nonnegativity of $\widetilde{\boldsymbol{\Lambda}}_m$ and its rows summing to 1.

Cross-modal label mixing: Finally, we compute a single multimodal label $\widetilde{\mathbf{y}} \in \mathbb{R}^{n_O}$ for the mini-batch by averaging $\widetilde{\mathbf{y}}_m$ over modalities:

$$\widetilde{\mathbf{y}} = \frac{1}{M} \sum_m \widetilde{\mathbf{y}}_m. \tag{9}$$

This step, called *cross-modal label mixing*, is only meaningful when mixing factor reweighting and anisotropic mixing are used, in which case $\widetilde{\mathbf{y}}_m$ are different for each modality.

B. Discussion

We now discuss the key features of $\mathcal{P}$owMix, to better understand the underlying idea of each algorithmic component.

Varying number of mixed examples: $\mathcal{P}$owMix enables the generation of a variable number n_O of mixed examples. These mixtures lie in the convex hull of the mini-batch in the representation space. The value of n_O is a hyperparameter, which is decoupled from the mini-batch size and much larger in practice. As such, it generates a plethora of mixtures, leading to more loss terms being calculated per example during model training. As in MultiMix [26], it is hypothesized that this provides a better approximation of the expected risk integral.

Mixing factor reweighting: Prior works [25], [26] perform mixing factor reweighting on dense features of image patches, applied to a unimodal task. By contrast, our approach is inherently multimodal, reweighting the mixing factors of each mini-batch example by normalizing across modalities. Our intuition is that by reweighting each modality in a multimodal m-tuple according to the sum of its features we are able to reduce the impact of uninformative (near zero) unimodal instances in the representation space. Unlike prior work [25], [26], our approach is also agnostic to the architecture and pooling mechanism of the input modality encoders. We experimentally verify that it improves performance.

Anisotropic mixing: Given m hidden tensors $\mathbf{H}_m \in \mathbb{R}^{B \times d_m}$, $\mathcal{P}$owMix samples a separate mixing matrix $\mathbf{\Lambda}_m \in \mathbb{R}^{n_O \times B}$ for each modality m. This enables modality-specific mixing, that is, the ability of the algorithm to exhibit different mixing strategies across modalities. This property is shown to be critical for $\mathcal{P}$owMix to perform well.

Dynamic mixing and power set: Considering the formation of the reweighted mixing matrix $\widetilde{\mathbf{\Lambda}}_m$ (6), we track two sources of randomness. The first is due to mixing factors in $\mathbf{\Lambda}_m$, sampled from the Dirichlet distribution, which is common in prior mixing methods, *e.g.* [26]. The second is due to the binary mask $\mathbf{M}$, sampled from the Bernoulli distribution, which is unique to our method. One may interpret the interpolation process as sampling a subset of examples from the *power set* $\mathcal{P}$ of the mini-batch in the representation space. The subset sampling is then followed by the formation of a convex combination based on the selected representations. This is a *dynamic mixing* process in the sense of using a different subset for each generated mixed example. The $\mathcal{P}$owMix acronym of the proposed method alludes to the use of the power set $\mathcal{P}$.

In practice, the effect of sampling a subset prior to forming a convex combination is that $\widetilde{\mathbf{\Lambda}}_m$ is *sparse*, having a small number of nonzero entries per row (2 to 4 on average according to our default settings). Thus, only few mini-batch examples are interpolated for each generated mixture. While it is possible to control the entropy of mixing factors by adjusting the hyperparameter $\boldsymbol{\alpha}_m$ of the Dirichlet distribution in sampling $\mathbf{\Lambda}_m$ itself, it is shown experimentally that true sparsity is superior in our multimodal problem. As shown in 2b and 2c, $\mathcal{P}$owMix can provide mixing instances between interpolation of *pairs*, as in Manifold Mixup [54], and *all* mini-batch examples, as in MultiMix [26]. Sharing the binary mask $\mathbf{M}$ across modalities is empirically shown to be essential for $\mathcal{P}$owMix to work well.

Cross-modal label mixing: $\mathcal{P}$owMix uses mixing factor reweighting and anisotropic mixing, which result in different reweighted mixing matrices $\widetilde{\mathbf{\Lambda}}_m$ (6) and thus different mixed labels $\widetilde{\mathbf{y}}_m$ (8) for each modality m. To unify these into a single label vector $\widetilde{\mathbf{y}}$, $\mathcal{P}$owMix averages the mixed labels $\widetilde{y}_m$ over modalities (9). This assumes equal contribution from each modality in label generation. While this assumption may not hold universally, empirical evidence has demonstrated its effectiveness in practice. The averaging operation is purely multimodal, intertwining label information across different modalities. We also experiment with a learnable variant, but it performs worse than the baseline since it places higher weight on the text modality and much smaller to other modalitites, undermining the multimodal learning process [15], [16], [20], [32].

V. Experimental setup

We evaluate $\mathcal{P}$owMix and other mixing techniques over three benchmark datasets for MSA and three distinct archetypal multimodal networks. In the following, we provide a detailed description of our experimental setup.

A. Benchmark datasets

MOSI: CMU-MOSI [27] is an English MSA benchmark dataset consisting of YouTube videos ($\approx$ 2.5h), featuring monologues where individuals express opinions, stories and reviews. These videos range from 2-5 minutes in length. CMU-MOSI contains 2199 utterance-level video segments from 93 videos and 89 distinct speakers (41 female and 48 male), with an average segment length of 4.2 sec. Each segment is manually transcribed and annotated with sentiment intensity scores ranging from -3 (strongly negative) to 3 (strongly positive).

MOSEI: CMU-MOSEI [28] is the largest MSA benchmark dataset ($\approx$ 66h). Compared to MOSI, it offers a more diverse range of samples, video topics and speakers. MOSEI contains 23,453 manually transcribed and annotated utterance-level video segments from 1000 distinct speakers and covers 250 topics. The average segment length is 7.28 sec, with segmentation based on punctuation from the high-quality manual transcriptions. Each segment is manually annotated in a Likert scale from -3 to 3 as in MOSI.

***SIMS**:* The CH-SIMS [29] is a Chinese MSA benchmark dataset, comparable in size to MOSI ($\approx$ 2.3h). It consists of 2281 utterance-level monologue video segments from 60 diverse videos, including movies, TV series, and variety shows. Each segment is manually segmented, transcribed, and annotated with sentiment intensity scores ranging from -1 (strongly negative) to 1 (strongly positive). While SIMS provides both multimodal and unimodal annotations, we only leverage the multimodal labels. The average length of each video segment is 3.67 sec.

B. Multimodal features

Processing raw multimodal video streams is computationally intensive and might also face copyright issues. Therefore, benchmarks in this field typically include a set of extracted features [27]–[29]. Since feature extraction for emotion and sentiment is a challenge with varied approaches [61], direct algorithm comparison can be problematic. In our study, we utilize the feature set provided in [61] for fair comparison across benchmarks.

***Text modality**:* Semantic word embeddings mainly rely on pretrained language models. Following [61], we use BERT [36] embeddings adopted from their open-source transformer implementations [62]. In particular, we use `bert-base-uncased` for English and `bert-base-chinese` for the Chinese language. Eventually, each word is tokenized and represented as a 768-dim word vector.

***Acoustic modality**:* The acoustic modality predominantly uses hand-crafted features. MOSI and MOSEI employ the COVAREP [63] framework to extract low-level descriptors (LLDs) like pitch and 12 Mel-freq cepstral coefficients (MFCCs), yielding a 74-dim frame-level feature. For SIMS, we use Librosa [64], resulting in a 33-dim acoustic representation per frame.

***Video modality**:* Standard video features for MSA tasks include facial landmarks, eye gaze, and facial action units. For MOSI and MOSEI, 35 facial action units are extracted using Facet [3], focusing on emotion-related movements. In SIMS, the OpenFace2.0 toolkit [65] extracts 68 facial landmarks, 17 facial action units, and other features, forming a 709-dim frame-level representation.

C. Evaluation metrics

***Performance metrics**:* Aligning with existing literature [9]–[11], [61], we evaluate MSA as a regression task using *mean absolute error* (MAE) and *Pearson correlation* (Corr). *Classification accuracy*, denoted as Acc-k for k classes, is also used by mapping regression scores to discrete categories. For binary metrics (Acc-2, $F1$), in line with [9], [61], we exclude neutral (zero-valued) predictions, focusing on positive versus negative values.

[3]https://imotions.com/platform

TABLE I
PROPERTIES OF MSA ARCHETYPAL MODELS. *FT-BERT*: BERT ENCODER FINE-TUNING; *Uni.Encoder*: UNIMODAL PROJECTOR OR ENCODER MODULE; *Objectives*: NUMBER OF OBJECTIVES IN LEARNING RECIPE. CA: CROSS-ATTENTION; SA: SELF-ATTENTION.

MODEL	FT-BERT	UNI.ENCODER	EARLY / LATE FUSION	OBJECTIVES
MulT		1D-CNN	CA / Concat	1
MISA	✓	LSTM	— / SA + Concat	4
Self-MM	✓	LSTM	— / Concat	4
ALMT	✓	Transformer	CA / CA	1

***Robustness metrics**:* Evaluating MSA model robustness, especially against textual modality dominance, is crucial [18], [66]. We assess model robustness under various noisy inputs using the same metrics as for performance. To quantify the robustness of each model under noisy inputs we employ the Area Under Indicators Line Chart (AUILC) metric [45]. In particular for increasing missing rates $\{r_1, \cdots r_n\}$ and their corresponding performance metrics $\{p_1, \cdots, p_n\}$ the AUILC is:

$$\sum_{i=1}^{n-1} \frac{p_i + p_{i+1}}{2}(r_{i+1} - r_i) \tag{10}$$

For non-error metrics, e.g., F1 and Corr, higher AUILC values indicate better models.

D. Competitors and MSA models

We compare our $\mathcal{P}$owMix multimodal regularization method against Manifold MixUp [54] and state-of-the-art MutiMix [26], as discussed in section III.

All comparisons are performed on three different archetypal MSA architectures, namely MulT [9], MISA [10] and Self-MM [11]. These models have demonstrated strong performance across the datasets we examine [61]. As shown in Table I, they represent a diverse range of architectural choices, including LSTM, CNNs, and transformers. They also employ various fusion strategies and unique learning recipes, including single-task and multi-task objectives. We reproduce those three models for fair comparison. We also report other models from the literature to provide a holistic performance overview.

***LF-DNN**:* The *late fusion deep neural network* (LF-DNN) [67] learns unimodal features separately for each modality, then concatenates them for multimodal prediction.

***TFN**:* The *tensor fusion network* (TFN) [8] employs LSTM for text and averages acoustic and visual features. Latent representations from DNN-processed modalities are concatenated, forming a high-dimensional multimodal space.

***MAG-BERT**:* In the MAG-BERT [35] model, a multimodal adaptation gate is introduced and integrated with a pretrained BERT encoder to handle multimodal information processing.

***MulT**:* The *multimodal transformer* (MulT) [9] employs a 1D-CNN as a unimodal projector for reducing the dimensionality of input features. MulT uses early fusion through *cross-attention* (CA) blocks. These blocks facilitate interaction and integration of information across different modalities. After this early fusion step, the model processes the combined

multimodal streams using *self-attention* (SA) mechanisms. The output of these processes is then concatenated for the final prediction. All mixing operations are integrated prior to concatenation. MulT is optimized based on a single task loss.

***MISA**:* The MISA model [10] uses LSTM networks to process audio and video modalities. For the text modality, it fine-tunes the BERT encoder. MISA is designed to embed unimodal representations into both a shared multimodal space and distinct unimodal spaces. This design promotes the extraction of common features across modalities while also preserving modality specific features.

In the final stages, MISA decodes all these six representations (from both common and individual spaces) in one branch for reconstruction, while in another branch, it independently merges them using SA to make the final prediction. MISA thus combines a task loss with reconstruction, repelling and attractive objectives. All mixing algorithms are employed before the SA block, treating the six representations as independent modalities.

***Self-MM**:* The Self-MM model [11] also relies on LSTM networks to process the audio and visual features. For the text modality, it fine-tunes the BERT encoder. Self-MM implements a pseudolabeling component called *unimodal label generation module* (ULGM), which generates unimodal labels from the multimodal label and the unimodal embeddings. These generated labels then influence the learning process through backpropagation. For the final prediction, Self-MM concatenates the unimodal representations and processes them through a dual linear layer setup. The model combines a task loss with an additional loss for each modality, derived from the pseudolabeling network. All mixing operations are integrated before concatenation.

***ALMT**:* The ALMT model [39] uses transformer layers to process features from every modality and also fine-tunes the BERT encoder. The encoded features are processed by an Adaptive Hyper-modality module consisting of self-attention layers for the language modality, cross-attention layers between text, and the concatenation of acoustic, visual, and learnable hypermodality. The output of this module is language-based features and fused features, which are combined via cross-attention fusion blocks. $\mathcal{P}$owMix is applied to the output of the Adaptive Hyper-Modality block.

E. Implementation details

We employ the M-SENA framework [61] for MSA model evaluation, implementing all models in PyTorch [68] and conducting all experiments on a single NVIDIA RTX 3090. We use the Adam [69] optimizer with early stopping and set hyperparameters per M-SENA's guidelines[4]. Results for MulT, MISA and Self-MM are reproduced from open-source implementations for fair comparison. Results for ALMT are based on (unofficial) open-source implementation[5].

Mixing is integrated before the late fusion stage in training and excluded during inference. For Self-MM model, following official recommendations [11], we initiate mixing only after the first two epochs for SIMS and MOSEI, and after one epoch for MOSI. Metrics are averaged over at least five independent runs, while robustness assessments use 15 runs, following [61], [70].

For $\mathcal{P}$owMix, we sample the masking probability as $\mathbf{P} \sim U(2,4)$ and the interpolation hyperparameter as $\boldsymbol{\alpha} \sim U(0.5, 2.0)$. For MultiMix, we use the default hyperparameters from [26], while Manifold MixUp performs best with $\alpha = 1.0$. When tuning hyperparameters such as the probability p_{mix} of applying the mixing algorithm, as well as the number n_O of generated mixed examples for MultiMix and $\mathcal{P}$owMix, we employ the following strategy. Initially, n_O is set to 256, and p_{mix} is optimized. Subsequently, n_O is optimized based on the optimal p_{mix} value. The process may be repeated if the results are not satisfying. For Manifold MixUp, because of the small batch size, we process batches twice the baseline size, split each batch in half and perform mixing between the two halves.

VI. Experimental results

We evaluate $\mathcal{P}$owMix against competitors over a diverse set of multimodal networks across different MSA benchmark datasets. All latent feature regularizers are applied before the late fusion part of each architecture.

A. Comparison with the state of the art

To ensure a fair comparison between different mixing methods and consistency across all evaluation metrics, we reproduce MulT, MISA, Self-MM and ALMT baseline models. Additionally, we present results from established frameworks, which are generally comparable with our reproduced results. We primarily compare our results to the reproduced ones.

Table II evaluates $\mathcal{P}$owMix against the state of the art. These results clearly show that integrating $\mathcal{P}$owMix leads to consistent performance improvements across all metrics and datasets over the reproduced baseline models. This result clearly illustrates the width of applicability and consistency of the proposed method across various architectures, fusion schemes and learning recipes. Notably, in the vast majority of the examined metrics (72%), Manifold MixUp and MultiMix fail to improve or even harm performance compared to the baseline. By contrast, all models are benefited by multimodal regularization across all setups. Next, we take a closer look at individual datasets.

***MOSI**:* By using $\mathcal{P}$owMix, Self-MM outperforms all examined models, both reproduced and original. For MulT, improves by 0.75 Acc-2 and 0.023 MAE.

***MOSEI**:* By using $\mathcal{P}$owMix, MISA improves by 0.95 Acc-5 and 1.04 Acc-7. Self-MM improves by 0.86 for binary metrics, outperforming its original results by 1.0 on average and even outperforming MISA, which has better baseline performance on these metrics. Moreover, ALMT achieves the best performance in binary metrics, i.e., Acc-2 and F1.

***SIMS**:* $\mathcal{P}$owMix significantly boosts all models. By using $\mathcal{P}$owMix, MulT outperforms the stronger baseline Self-MM model across three metrics, which clearly confirms the benefits of regularization in multimodal architectures.

[4]https://github.com/thuiar/MMSA/blob/master/src/MMSA/config/config_regression.json

[5]https://github.com/Haoyu-ha/ALMT

TABLE II
State of the art comparisons. M.MIXUP: MANIFOLD MIXUP. †: RESULTS REPORTED IN [61]; ‡: RESULTS REPORTED IN [39]; ALL OTHER RESULTS ARE REPRODUCED. ↑ / ↓: HIGHER/LOWER IS BETTER. RED: WORSE THAN THE BASELINE; BOLD: BEST FOR EACH MSA MODEL.

MODEL	MOSI						MOSEI						SIMS			
	ACC2↑	F1↑	MAE↓	CORR↑	ACC5↑	Acc7↑	ACC2↑	F1↑	MAE↓	CORR↑	ACC5↑	Acc7↑	ACC2↑	F1↑	MAE↓	CORR↑
LF-DNN†	79.39	79.45	0.945	0.675	-	-	82.78	82.38	0.558	0.731	-	-	76.68	76.48	0.446	0.567
TFN†	78.02	78.09	0.971	0.652	-	-	82.23	81.47	0.573	0.718	-	-	77.07	76.94	0.437	0.582
MAG-BERT†	83.41	83.47	0.761	0.772	-	-	84.87	84.85	0.539	0.764	-	-	74.44	71.75	0.492	0.399
MulT†	80.21	80.22	0.912	0.695	-	-	84.63	84.52	0.559	0.733	-	-	78.56	79.66	0.453	0.564
MISA†	82.96	82.98	0.761	0.772	-	-	84.63	84.52	0.559	0.733	-	-	76.54	76.59	0.447	0.563
Self-MM†	84.30	84.31	0.720	0.793	-	-	84.06	84.12	0.531	0.766	-	-	80.04	80.44	0.425	0.595
ALMT‡	86.43	86.47	0.683	0.805	56.41	49.42	86.79	86.86	0.526	0.779	55.96	54.28	81.19	81.57	0.404	0.619
MulT	80.26	80.32	0.927	0.689	40.10	34.71	84.07	83.93	0.564	0.731	53.97	52.56	77.77	77.99	0.442	0.584
+ M.MixUp	80.41	80.36	0.928	0.686	39.14	34.26	84.02	83.92	0.563	0.729	54.19	52.50	78.09	77.95	0.445	0.576
+ MultiMix	80.46	80.49	0.911	0.688	39.33	34.96	84.08	84.01	0.563	0.733	53.99	52.39	78.09	77.87	0.445	0.575
+ $\mathcal{P}$owMix	**81.01**	**80.99**	**0.904**	**0.696**	**40.65**	**35.00**	**84.44**	**84.38**	**0.559**	**0.738**	**54.26**	**52.75**	**79.04**	**78.51**	**0.437**	**0.595**
MISA	82.93	82.95	0.772	0.774	47.55	42.10	84.51	84.47	0.549	0.759	53.57	51.96	76.59	76.20	0.457	0.550
+ M.MixUp	83.08	83.12	0.783	0.770	46.94	42.10	84.50	84.32	0.551	0.755	53.61	52.10	75.60	75.47	0.460	0.549
+ MultiMix	82.82	82.86	0.780	0.778	47.06	41.80	84.55	84.47	0.551	0.757	53.94	52.30	76.67	76.15	0.455	0.547
+ $\mathcal{P}$owMix	**83.49**	**83.50**	**0.761**	**0.780**	**48.02**	**42.65**	**84.97**	**84.86**	**0.543**	**0.762**	**54.52**	**53.00**	**77.35**	**76.97**	**0.441**	**0.569**
Self-MM	84.22	84.23	0.724	0.791	52.22	45.64	84.26	84.24	0.532	0.765	55.52	53.85	78.16	78.15	0.417	0.592
+ M.MixUp	84.38	84.37	0.722	0.792	53.50	46.33	84.24	84.23	0.532	0.765	55.57	53.82	78.56	78.58	0.414	0.594
+ MultiMix	84.35	84.38	0.723	0.792	52.45	45.89	84.17	84.16	0.547	0.751	54.53	52.84	77.62	77.77	0.426	0.576
+ $\mathcal{P}$owMix	**84.76**	**84.78**	**0.712**	**0.795**	**53.86**	**46.88**	**85.11**	**85.10**	**0.528**	**0.770**	**55.87**	**54.25**	**79.02**	**78.94**	**0.412**	**0.599**
ALMT	83.90	83.89	0.746	0.784	48.77	43.58	85.23	85.32	0.539	0.766	54.64	53.05	78.16	78.16	0.433	0.575
+ $\mathcal{P}$owMix	**84.30**	**84.30**	**0.741**	**0.788**	**49.24**	**44.25**	**85.85**	**85.94**	**0.535**	**0.770**	**54.89**	**53.29**	**78.91**	**79.13**	**0.429**	**0.580**

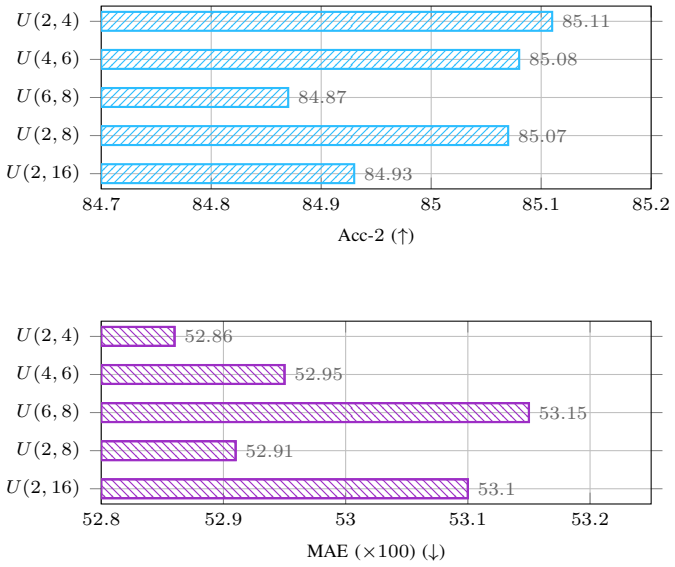

Fig. 3. The effect of *subset sampling* in $\mathcal{P}$owMix, as controlled by the uniform distribution $U(a,b)$ used to sample the hyperparameter $\mathbf{P} \in \mathbb{R}^{n_O \times B}$ of the Bernoulli distribution from which we sample the binary mask $\mathbf{M} \in \mathbb{R}^{n_O \times B}$. Using Self-MM model on MOSEI. ↑ / ↓: higher/lower is better.

TABLE III
EFFECT OF ALGORITHMIC COMPONENTS OF $\mathcal{P}$OWMIX: ANISOTROPIC MIXING (ANISO.), MIXING FACTOR REWEIGHTING (REWEIGHT), CROSS-MODAL MASK SHARING (M. SHARE), AND DYNAMIC MIXING (D.MIX). USING SELF-MM MODEL ON MOSEI. ↑ / ↓: HIGHER/LOWER IS BETTER.

ANISO.	REWEIGHT	M.SHARE	D.MIX	ACC2↑	ACC5↑	MAE↓
✓	✓	✓	✓	**85.11**	**55.87**	**0.528**
	✓	✓	✓	84.90	55.60	0.529
✓		✓	✓	85.00	55.77	**0.528**
✓	✓		✓	84.99	55.30	0.531
✓	✓			84.62	55.21	0.535

B. Ablation study

To investigate the effect of each hyperparameter and algorithmic component in $\mathcal{P}$owMix, we conduct extensive experiments on MOSEI (largest benchmark) with Self-MM (best performing model).

1) Subset sampling: In this experiment, we vary the number of interpolated examples within a mini-batch for forming convex combinations in $\mathcal{P}$owMix. This is controlled by a uniform distribution $U(a,b)$ used to sample the hyperparameter $\mathbf{P} \in \mathbb{R}^{n_O \times B}$ of the Bernoulli distribution from which we sample the binary mask $\mathbf{M} \in \mathbb{R}^{n_O \times B}$. This uniform distribution means that there are from a to b nonzero entries on average in each row of the mask $\mathbf{M}$, thus also each row of the reweighted mixing matrix $\hat{\mathbf{\Lambda}}_m$. In turn, this means that we are interpolating from a to b mini-batch examples on average.

Figure 3 shows the results for a variety of choices for a, b. According to both metrics, the intervals are ranked by decreasing performance as $U(2,4)$, $U(2,8)$, $U(4,6)$, $U(2,16)$, and $U(6,8)$. This highlights the importance of sampling a small subset of mini-batch examples (both a and b being small) and the sparsity of $\hat{\mathbf{\Lambda}}_m$, justifying our dynamic mixing process. Notably, even the least effective choice, $U(6,8)$, still outperforms the baseline Self-MM model. We choose $U(2,4)$ by default in $\mathcal{P}$owMix based on these results.

2) Number of generated mixed examples: Figure 4 shows the effect of the number n_O of generated mixed examples. We observe that a smaller number of generated mixed examples,

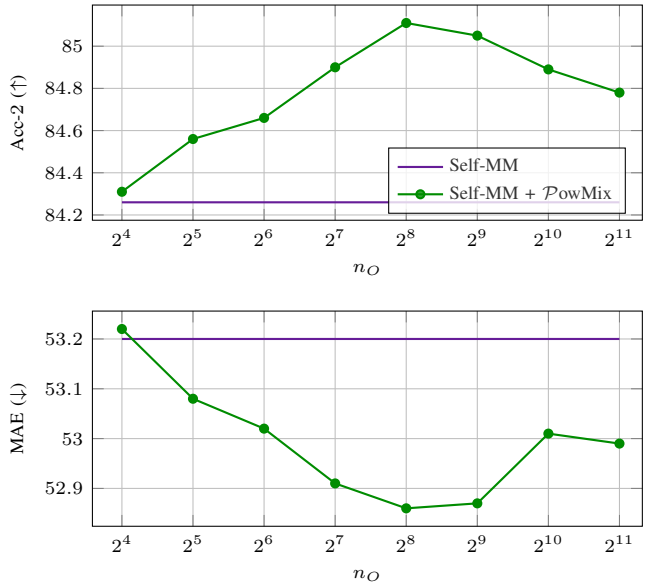

Fig. 4. Impact of the number n_O of *generated mixed examples* on MOSEI. ↑ / ↓: higher/lower is better.

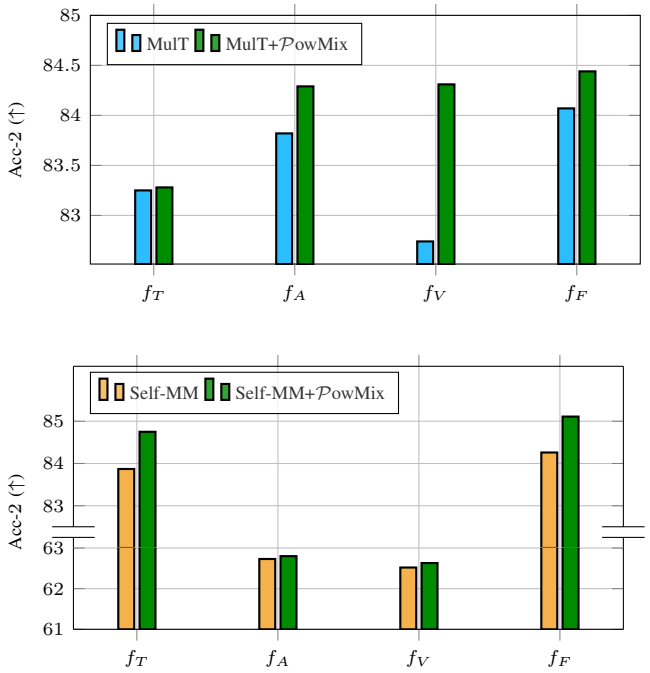

Fig. 5. *Unimodal evaluation* analysis on MOSEI. Effect of PowMix on performance of encoders f_m of individual modality $m \in \{T, A, V\}$ and fusion network f_F for different MSA architectures. f_m evaluated by training a linear head on each modality representations ($\mathbf{h}_m$), while keeping f_m frozen. f_F evaluated based on Table II. T: text; A: acoustic; V: video. MulT: early fusion; Self-MM: late fusion. ↑ / ↓: higher/lower is better.

such as 2^4 or 2^5, does not significantly benefit the model, whereas a very large number, like 2^{11}, seems suboptimal. The best-performing values are found among $\{2^7, 2^8, 2^9\}$, with 2^8 performing best. We underline that PowMix is effective over a very wide range of n_O values, outperforming the baseline Self-MM model for all values tested.

3) Algorithmic components: By turning four critical algorithmic components on/off during training, we study their effect on PowMix. In paricular, we examine anisotropic mixing, mixing factor reweighting, cross-modal mask sharing and dynamic mixing. It is important to note that cross-modal label mixing (9) is linked with other components. When turning off mixing factor reweighting and anisotropic mixing, we also turn off cross-modal label mixing, since in this case $\widetilde{\mathbf{y}}_m$ (8) are the same across modalities. When turning dynamic mixing off, i.e., use unit mask in (6), we assume that the mask sharing feature is also deactivated.

Table III shows that dynamic mixing, *i.e.*, using sparse binary masks, is the most important component of PowMix. Morever, mask sharing, *i.e.*, masking the same examples across modalities, is also a crucial component. Therefore, we use a sparse shared cross-modal mask in all our experiments. Anisotropic mixing is also essential, as its absence lowers performance. The benefit of reweighting is also clear, supporting findings from other studies [25]. These findings underscore the synergetic impact of the features of PowMix; removing any of them leads to a drop in performance.

C. Analysis

To better understand how PowMix works and affects the models, we conduct analysis experiments on MOSEI.

1) Fusion Representation evaluation: In this experiment, we explore how PowMix influences each representation *prior to* and *after* the final fusion operation (f_F). Referring to Figure 1, we denote by f_m each (stream) module prior to fusion operation. Note here that in some architectures, e.g., MulT, f_m act as early fusion operators, thereby processing multimodal information. In the Self-MM architecture, on the other hand, the f_m modules process unimodal information. The index $m \in \{T, A, V\}$ (T: text, A: acoustic, V: video) refers to each stream encoder without being restricted to unimodal operations. We train the models with and without PowMix. After training, we assess the performance of each encoded stream by training a (separate) linear head on every representation $\mathbf{h}_m$ while keeping f_m frozen. The fusion network f_F is evaluated based on Table II. We repeat each experiment three times and report the average performance.

The results are shown in Figure 5. All modalities perform well in the MulT architecture (f_m), which uses early fusion. Applying PowMix improves f_A by 0.47 Acc-2 and f_V by 1.57, which is significant, while f_T shows minimal improvement (0.03). The fusion network f_F improves notably by 0.37 Acc-2, though this is less than the per-modality improvement.

By contrast, the Self-MM architecture, which uses late fusion, shows different trends. Here, the acoustic and visual modalities perform substantially worse ($\approx$ 63 Acc-2) than the dominant text modality (83.87). With PowMix, the audio and visual streams show minor improvements ($\approx$ 0.07), but the text modality f_T improves significantly by 0.88 Acc-2. Interestingly, this gain is directly transferred to the fusion network f_F, improving it by 0.85 Acc-2.

These findings are mutually informative. We interpret the results based on architectural differences and unimodal feature analysis from [20]. In MulT, the substantial per-modality

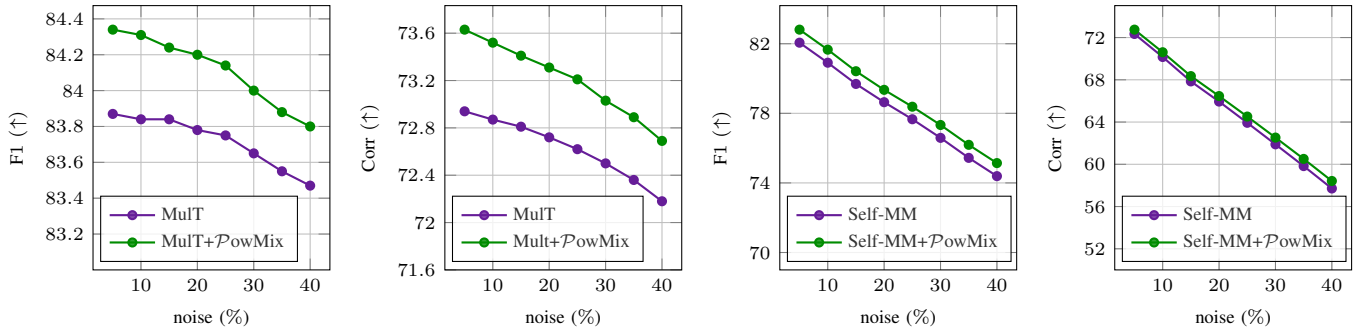

Fig. 6. *Robustness to noise* analysis on MOSEI. Input frames randomly dropped with probability ranging from 5% to 40%. Average metrics reported over dropping being aligned across modalities and independent. ↑ / ↓: higher/lower is better.

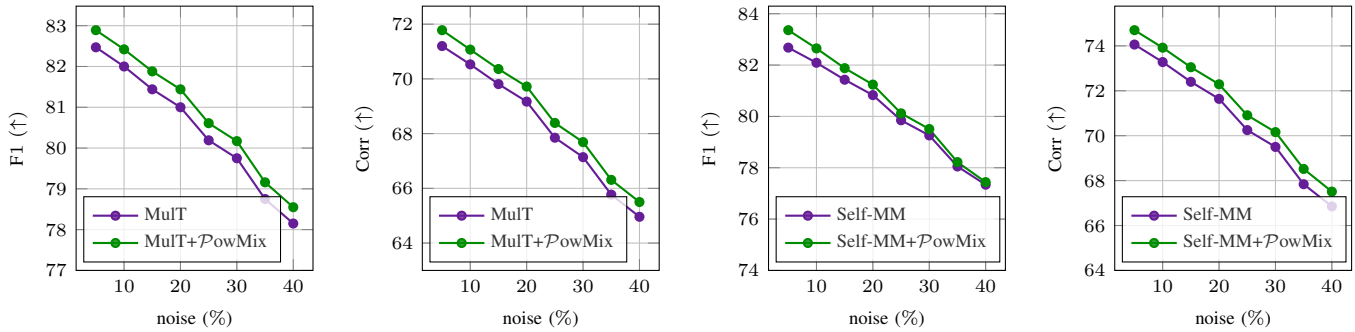

Fig. 7. *Modality dominance* analysis on MOSEI. Text modality input completely dropped with probability ranging from 5% to 40% or replaced with a mean representation over the training set. Average metrics reported over the two scenarios. ↑ / ↓: higher/lower is better.

improvements do not translate to similar multimodal predictive gains. Due to its early fusion approach, the gain of each modality is inherently multimodal, limiting the margin for additional gain at the final fusion stage. Conversely, in Self-MM, the significant boost in the text stream sufficiently enhances the performance of the final fusion layer, which aligns with the concept that unimodal improvements directly benefit the fusion process.

2) Robustness to noise: Next, we investigate the impact of $\mathcal{P}$owMix on model robustness. We train MulT and Self-MM models with and without $\mathcal{P}$owMix using clean data and then evaluate them in noisy conditions. In particular, we randomly drop input frames from each modality (temporal drop) with a probability p determining the noise intensity and ranging from 5% to 40%. This drop occurs either in a correlated fashion (simultaneously across all modalities) or independently. We average the results over the two noise types.

Figure 6 shows that both MulT and Self-MM are impacted by noise. Interestingly, integrating $\mathcal{P}$owMix does not significantly alter the effect of noise, as indicated by the slope of the curves. Models trained with $\mathcal{P}$owMix maintain their gain over baselines in noisy conditions. Notably, MulT exhibits greater noise robustness than Self-MM, as indicated by a smaller drop of performance, a result also verified by the AUILC metric in Table IV.

3) Modality dominance: For dominance analysis, we inject noise solely into the text modality to assess the conditional dependence of the model on language to make predictions. The noise is applied in two forms: completely dropping the text modality input with probability p or replacing it with a mean representation over the training set [16]. We average the results across these two noise variants.

TABLE IV
AUILC scores. *Robustness*: RANDOMLY DROP FEATURES (TIMESTEPS) FROM THE AVAILABLE MODALITIES. *Dominance*: RANDOMLY DROP FEATURES FROM THE TEXT MODALITY. ↑: HIGHER IS BETTER; BOLD: BEST FOR EACH MSA MODEL.

MODEL	ROBUSTNESS		DOMINANCE	
	F1↑	CORR↑	F1↑	CORR↑
MulT	0.293	0.254	0.281	0.239
+ $\mathcal{P}$owMix	**0.295**	**0.256**	**0.283**	**0.241**
Self-MM	0.273	0.227	0.280	0.247
+ $\mathcal{P}$owMix	**0.276**	**0.229**	**0.282**	**0.250**

Figure 7 shows the results for both MulT and Self-MM models. It reveals that text-only noise significantly affects both models, confirming the text dominance in MSA models as observed in prior work [22], [71]. Importantly, $\mathcal{P}$owMix-trained models consistently outperform the baselines across all examined noise levels, as illustrated in Table IV.

4) Limited data: This experiment investigates the effectiveness of $\mathcal{P}$owMix in limited data scenarios. We opt for the MulT model for its simple training process and lower

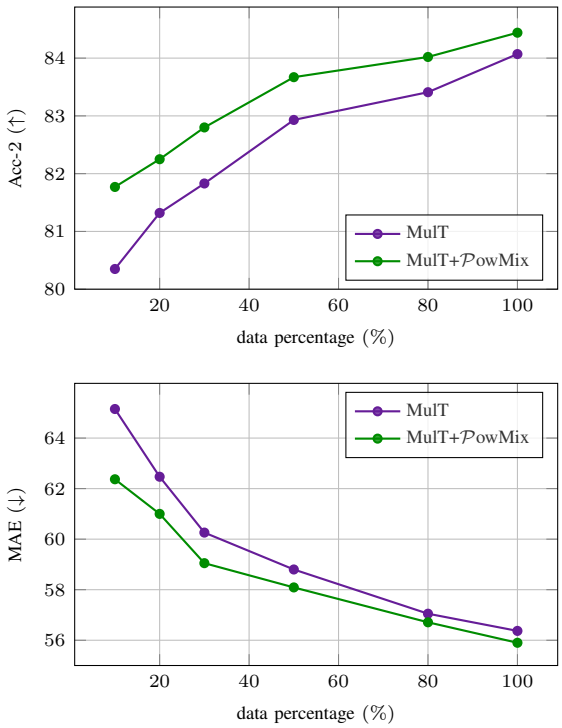

Fig. 8. *Limited data* analysis. Model trained on progressively larger portions of MOSEI. ↑ / ↓: higher/lower is better.

parameter usage. The model is trained on progressively larger portions of MOSEI. As shown in Figure 8, the model trained with $\mathcal{P}$owMix consistently outperforms the baseline across all data sizes. This performance enhancement is most pronounced in the lower data regime, specifically between $10-20\%$ of the data, where we observe an improvement of ≈ 1.21 Acc-2 and ≈ 0.022 MAE. This finding underlines the similarity of $\mathcal{P}$owMix to other regularization techniques, often showing greater improvements in scenarios with limited data.

5) FLOPs Analysis: Since most MSA models utilize the standard BERT architecture [10], [11] as a language encoder, we compare $\mathcal{P}$owMix's FLOPs with that of BERT. We calculate the FLOPs for a standard scenario of a maximum language token length of 50 for BERT. For $\mathcal{P}$owMix, we assume three modalities of hidden dimension 128, which is typical for MSA models [10], and vary the number of generated examples n_O in the range $\{256, 512, 1024\}$. We illustrate our findings in Table V and observe that the amount of GFLOPs (GF) required by $\mathcal{P}$owMix is a tiny fraction compared to the GFLOPs of BERT, making its overhead negligible in practice.

TABLE V
FLPOs Analysis. FLOP CALCULATION OF $\mathcal{P}$OWMIX FOR DIFFERENT NUMBER OF GENERATED MIXTURES n_0 COMPARED TO BERT. ALL RESULTS ARE IN GIGAFLOPS (GF).

MODULE	n_O		
	256	512	1024
$\mathcal{P}$owMix	0.203 (GF)	0.406 (GF)	0.811 (GF)
BERT	275.12 (GF)	275.12 (GF)	275.12 (GF)
%-BERT	0.07 %	0.15 %	0.30 %

VII. CONCLUSIONS

The increasing complexity of neural networks, especially in multimodal scenarios, underscores the critical need for effective regularization techniques. With the focus of MSA research on developing advanced architectures and diverse learning strategies, there is a clear demand for versatile multimodal regularization methods. To address this, we have introduced $\mathcal{P}$owMix, a novel approach specifically tailored for multimodal tasks. $\mathcal{P}$owMix incorporates five key elements: 1) generating a varying number of mixed examples, 2) mixing factor reweighting, 3) anisotropic mixing, 4) dynamic mixing, and 5) cross-modal label mixing. These elements collectively form an algorithm that improves training in multimodal contexts.

Our extensive experimentats across various MSA datasets and models demonstrate the broad applicability and consistent performance improvements of $\mathcal{P}$owMix. Detailed ablation studies uncover the synergistic nature of its components, emphasizing that the full set of components is essential for its effective operation. Removing any algorithmic component results in performance degradation. Moreover, the ablation highlights the robustness of the algorithm to several hyperparameter choices, a valuable quality of the algorithm. Extented analysis shows that $\mathcal{P}$owMix operates differently yet effectively in both early and late fusion architectures. Moreover, we find that the integration of $\mathcal{P}$owMix into MSA models preserves robustness without sacrificing performance or enhancing text dominance and offers consistent gains across different scales of data.

Future research on $\mathcal{P}$owMix is promising. One important direction is the application of $\mathcal{P}$owMix in other multimodal tasks beyond MSA, to further validate its versatility and efficacy in diverse environments. Investigating the integration of $\mathcal{P}$owMix with more and diverse neural network architectures could further establish it as a generic multimodal regularizer. Another potential direction is to examine how $\mathcal{P}$owMix performs under contrastive unsupervised learning scenarios. This could contribute to the development of more generalized multimodal learning frameworks, capable of handling a wider spectrum of real-world tasks. Further in-depth analysis of each component of $\mathcal{P}$owMix could provide a clearer understanding of both the individual as well as the collective contributions to the learning process. Such insights could lead to the development of more refined, targeted and even explicit regularization methods. Moreover, making the Cross-modal label mixing component of the algorithm learnable by considering the text dominance constraint is a fruitful research direction which would manage each multimodal example in a different fashion without undermining the multimodal learning process or increasing text dominance. Finally, the exploration of learnable mixing strategies in general, could further advance the state of the current multimodal regularization arsenal.

VIII. ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their feedback. The authors thank G.Bastas and K.Kritsis (Athena RC) for fruitful discussions and providing feedback.

This work has been funded by the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101061303 (PREMIERE).

REFERENCES

[1] S. Narayanan and P. G. Georgiou, "Behavioral signal processing: Deriving human behavioral informatics from speech and language," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1203–1233, 2013.

[2] P. P. Filntisis, N. Efthymiou, P. Koutras, G. Potamianos, and P. Maragos, "Fusing body posture with facial expressions for joint recognition of affect in child–robot interaction," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4011–4018, 2019.

[3] S. Poria, E. Cambria, R. Bajpai, and A. Hussain, "A review of affective computing: From unimodal analysis to multimodal fusion," *Information Fusion*, vol. 37, pp. 98–125, 2017.

[4] L. Stappen, A. Baird, L. Schumann, and S. Bjorn, "The multimodal sentiment analysis in car reviews (muse-car) dataset: Collection, insights and improvements," *IEEE Transactions on Affective Computing*, 2021.

[5] E. Yadegaridehkordi, N. F. B. M. Noor, M. N. B. Ayub, H. B. Affal, and N. B. Hussin, "Affective computing in education: A systematic review and future research," *Computers & Education*, vol. 142, p. 103649, 2019.

[6] A. Metallinou, M. Wollmer, A. Katsamanis, F. Eyben, B. Schuller, and S. Narayanan, "Context-sensitive learning for enhanced audiovisual emotion classification," *IEEE Transactions on Affective Computing*, vol. 3, no. 2, pp. 184–198, 2012.

[7] M. Soleymani, D. Garcia, B. Jou, B. Schuller, S.-F. Chang, and M. Pantic, "A survey of multimodal sentiment analysis," *Image and Vision Computing*, vol. 65, pp. 3–14, 2017.

[8] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.-P. Morency, "Tensor fusion network for multimodal sentiment analysis," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 1103–1114.

[9] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, "Multimodal transformer for unaligned multimodal language sequences," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 6558–6569.

[10] D. Hazarika, R. Zimmermann, and S. Poria, "Misa: Modality-invariant and-specific representations for multimodal sentiment analysis," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1122–1131.

[11] W. Yu, H. Xu, Z. Yuan, and J. Wu, "Learning modality-specific representations with self-supervised multi-task learning for multimodal sentiment analysis," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, 2021, pp. 10 790–10 797.

[12] Y. Sun, S. Mai, and H. Hu, "Learning to learn better unimodal representations via adaptive multimodal meta-learning," *IEEE Transactions on Affective Computing*, 2022.

[13] J. Hessel and L. Lee, "Does my multimodal model learn cross-modal interactions? it's harder to tell than you might think!" in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 861–877.

[14] P. P. Liang, A. Zadeh, and L.-P. Morency, "Foundations and trends in multimodal machine learning: Principles, challenges, and open questions," Sep. 2022.

[15] W. Wang, D. Tran, and M. Feiszli, "What makes training multi-modal classification networks hard?" in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 12 695–12 705.

[16] N. Wu, S. Jastrzebski, K. Cho, and K. J. Geras, "Characterizing and overcoming the greedy nature of learning in multi-modal deep neural networks," in *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 2022, pp. 24 043–24 055.

[17] Y. Huang, C. Du, Z. Xue, X. Chen, H. Zhao, and L. Huang, "What makes multi-modal learning better than single (provably)," *Advances in Neural Information Processing Systems*, vol. 34, pp. 10 944–10 956, 2021.

[18] D. Gkoumas, Q. Li, C. Lioma, Y. Yu, and D. Song, "What makes the difference? an empirical comparison of fusion strategies for multimodal language analysis," *Information Fusion*, vol. 66, pp. 184–197, 2021.

[19] Y. Huang, J. Lin, C. Zhou, H. Yang, and L. Huang, "Modality competition: What makes joint training of multi-modal network fail in deep learning? (Provably)," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 17–23 Jul 2022, pp. 9226–9259.

[20] C. Du, J. Teng, T. Li, Y. Liu, T. Yuan, Y. Wang, Y. Yuan, and H. Zhao, "On uni-modal feature learning in supervised multi-modal learning," in *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, ser. Proceedings of Machine Learning Research, vol. 202. PMLR, 2023, pp. 8632–8656.

[21] Z. Liu, Z. Tang, X. Shi, A. Zhang, M. Li, A. Shrivastava, and A. G. Wilson, "Learning multimodal data augmentation in feature space," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[22] E. Georgiou, G. Paraskevopoulos, and A. Potamianos, "M3: MultiModal Masking Applied to Sentiment Analysis," in *Proc. Interspeech 2021*, 2021, pp. 2876–2880.

[23] Y. Liu, Z. Yuan, H. Mao, Z. Liang, W. Yang, Y. Qiu, T. Cheng, X. Li, H. Xu, and K. Gao, "Make acoustic and visual cues matter: Ch-sims v2. 0 dataset and av-mixup consistent module," in *Proceedings of the 2022 International Conference on Multimodal Interaction*, 2022, pp. 247–258.

[24] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *6th International Conference on Learning Representations, ICLR 2018*. OpenReview.net, 2018.

[25] J.-N. Chen, S. Sun, J. He, P. Torr, A. Yuille, and S. Bai, "Transmix: Attend to mix for vision transformers," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

[26] S. Venkataramanan, E. Kijak, L. Amsaleg, and Y. Avrithis, "Embedding space interpolation beyond mini-batch, beyond pairs and beyond examples," in *Advances in neural information processing systems*, 2023.

[27] A. Zadeh, R. Zellers, E. Pincus, and L.-P. Morency, "Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages," *IEEE Intelligent Systems*, vol. 31, no. 6, pp. 82–88, 2016.

[28] A. Zadeh and P. Pu, "Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Long Papers)*, 2018.

[29] W. Yu, H. Xu, F. Meng, Y. Zhu, Y. Ma, J. Wu, J. Zou, and K. Yang, "Ch-sims: A chinese multimodal sentiment analysis dataset with fine-grained annotation of modality," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 3718–3727.

[30] S. Poria, E. Cambria, D. Hazarika, N. Mazumder, A. Zadeh, and L.-P. Morency, "Multi-level multiple attentions for contextual multimodal sentiment analysis," in *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2017, pp. 1033–1038.

[31] Y. Gu, K. Yang, S. Fu, S. Chen, X. Li, and I. Marsic, "Multimodal affective analysis using hierarchical attention strategy with word-level alignment," in *Proceedings of the conference. Association for Computational Linguistics. Meeting*, vol. 2018. NIH Public Access, 2018, p. 2225.

[32] E. Georgiou, C. Papaioannou, and A. Potamianos, "Deep hierarchical fusion with application in sentiment analysis," *Interspeech 2019*, 2019.

[33] Y. H. Tsai, M. Ma, M. Yang, R. Salakhutdinov, and L. Morency, "Multimodal routing: Improving local and global interpretability of multimodal language analysis," in *Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP*, 2020, pp. 1823–1833.

[34] A. Joshi, A. Bhat, A. Jain, A. Singh, and A. Modi, "COGMEN: COntextualized GNN based multimodal emotion recognitioN," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2022, pp. 4148–4164.

[35] W. Rahman, M. K. Hasan, S. Lee, A. B. Zadeh, C. Mao, L.-P. Morency, and E. Hoque, "Integrating multimodal information in large pretrained transformers," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 2359–2369.

[36] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[37] J. Guo, J. Tang, W. Dai, Y. Ding, and W. Kong, "Dynamically adjust word representations using unaligned multimodal information," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 3394–3402.

[38] D. Wang, X. Guo, Y. Tian, J. Liu, L. He, and X. Luo, "Tetfn: A text enhanced transformer fusion network for multimodal sentiment analysis," vol. 136, 2023, p. 109259.

[39] H. Zhang, Y. Wang, G. Yin, K. Liu, Y. Liu, and T. Yu, "Learning language-guided adaptive hyper-modality representation for multimodal sentiment analysis," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2023, pp. 756–767.

[40] D. Wang, S. Liu, Q. Wang, Y. Tian, L. He, and X. Gao, "Cross-modal enhancement network for multimodal sentiment analysis," vol. 25, 2023, pp. 4909–4921.

[41] H. Pham, P. P. Liang, T. Manzini, L.-P. Morency, and B. Póczos, "Found in translation: Learning robust joint representations by cyclic translations between modalities," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 6892–6899.

[42] D. Yang, S. Huang, H. Kuang, Y. Du, and L. Zhang, "Disentangled representation learning for multimodal emotion recognition," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 1642–1651.

[43] W. Han, H. Chen, and S. Poria, "Improving multimodal fusion with hierarchical mutual information maximization for multimodal sentiment analysis," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 9180–9192.

[44] L. Sun, Z. Lian, B. Liu, and J. Tao, "Efficient multimodal transformer with dual-level feature restoration for robust multimodal sentiment analysis," *IEEE Transactions on Affective Computing*, 2023.

[45] Z. Yuan, Y. Liu, H. Xu, and K. Gao, "Noise imitation based adversarial training for robust multimodal sentiment analysis," vol. 26, 2024, pp. 529–539.

[46] G. Hu, T.-E. Lin, Y. Zhao, G. Lu, Y. Wu, and Y. Li, "UniMSE: Towards unified multimodal sentiment analysis and emotion recognition," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Dec. 2022, pp. 7837–7851.

[47] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[48] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[49] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 6022–6031.

[50] Z. Liu, S. Li, D. Wu, Z. Chen, L. Wu, J. Guo, and S. Z. Li, "Automix: Unveiling the power of mixup for stronger classifiers," in *European Conference on Computer Vision*, 2022, pp. 441–458.

[51] S. Li, Z. Wang, Z. Liu, D. Wu, and S. Z. Li, "Openmixup: Open mixup toolbox and benchmark for visual representation learning," *arXiv preprint arXiv:2209.04851*, 2022.

[52] S. Yoon, G. Kim, and K. Park, "Ssmix: Saliency-based span mixup for text classification," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 3225–3234.

[53] G. Kim, D. K. Han, and H. Ko, "SpecMix : A Mixed Sample Data Augmentation Method for Training with Time-Frequency Domain Features," in *Proc. Interspeech 2021*, 2021, pp. 546–550.

[54] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *International conference on machine learning*. PMLR, 2019, pp. 6438–6447.

[55] H. Guo, "Nonlinear mixup: Out-of-manifold data augmentation for text classification," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 4044–4051, Apr. 2020.

[56] H.-P. Chou, S.-C. Chang, J.-Y. Pan, W. Wei, and D.-C. Juan, "Remix: Rebalanced mixup," in *Computer Vision – ECCV 2020 Workshops, 2020, Proceedings, Part VI*, 2020, p. 95–110.

[57] A. Jindal, N. E. Ranganatha, A. Didolkar, A. G. Chowdhury, D. Jin, R. Sawhney, and R. R. Shah, "SpeechMix — Augmenting Deep Sound Recognition Using Hidden Space Interpolations," in *Proc. Interspeech 2020*, 2020, pp. 861–865.

[58] L. Sun, C. Xia, W. Yin, T. Liang, P. Yu, and L. He, "Mixup-transformer: Dynamic data augmentation for NLP tasks," in *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, 2020, pp. 3436–3440.

[59] X. Hao, Y. Zhu, S. Appalaraju, A. Zhang, W. Zhang, B. Li, and M. Li, "Mixgen: A new multi-modal data augmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, January 2023, pp. 379–389.

[60] T. Wang, W. Jiang, Z. Lu, F. Zheng, R. Cheng, C. Yin, and P. Luo, "VLMixer: Unpaired vision-language pre-training via cross-modal CutMix," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 17–23 Jul 2022, pp. 22 680–22 690.

[61] H. Mao, Z. Yuan, H. Xu, W. Yu, Y. Liu, and K. Gao, "M-SENA: An integrated platform for multimodal sentiment analysis," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, 2022, pp. 204–213.

[62] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, 2020, pp. 38–45.

[63] G. Degottex, J. Kane, T. Drugman, T. Raitio, and S. Scherer, "Covarep—a collaborative voice analysis repository for speech technologies," in *2014 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2014, pp. 960–964.

[64] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8. Citeseer, 2015, pp. 18–25.

[65] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, "Openface 2.0: Facial behavior analysis toolkit," in *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*. IEEE, 2018, pp. 59–66.

[66] C. Jin, C. Luo, M. Yan, G. Zhao, G. Zhang, and S. Zhang, "Weakening the dominant role of text: Cmosi dataset and multimodal semantic enhancement network," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2023.

[67] E. Cambria, D. Hazarika, S. Poria, A. Hussain, and R. Subramanyam, "Benchmarking multimodal sentiment analysis," in *Computational Linguistics and Intelligent Text Processing: 18th International Conference, CICLing 2017, Budapest, Hungary, April 17–23, 2017, Revised Selected Papers, Part II 18*. Springer, 2018, pp. 166–179.

[68] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[69] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[70] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 11, pp. 4037–4058, 2020.

[71] D. Hazarika, Y. Li, B. Cheng, S. Zhao, R. Zimmermann, and S. Poria, "Analyzing modality robustness in multimodal sentiment analysis," *arXiv preprint arXiv:2205.15465*, 2022.