MixMAS: A Framework for Sampling-Based Mixer Architecture Search for Multimodal Fusion and Learning

Abdelmadjid Chergui
Higher School of Computer Science 8 Mai 1945
SBA, Algeria
a.chergui@esi-sba.dz

Grigor Bezirganyan
Aix-Marseille Univ, LIS, CNRS
Marseille, France
grigor.bezirganyan@univ-amu.fr

Sana Sellami
Aix-Marseille Univ, LIS, CNRS
Marseille, France
sana.sellami@univ-amu.fr

Laure Berti-Équille
IRD, ESPACE-DEV
Montpellier, France
laure.berti@ird.fr

Sébastien Fournier
Aix-Marseille Univ, LIS, CNRS
Marseille, France
sebastien.fournier@univ-amu.fr

***Abstract*—Choosing a suitable deep learning architecture for multimodal data fusion is a challenging task, as it requires the effective integration and processing of diverse data types, each with distinct structures and characteristics. In this paper, we introduce MixMAS, a novel framework for sampling-based mixer architecture search tailored to multimodal learning. Our approach automatically selects the optimal MLP-based architecture for a given multimodal machine learning (MML) task. Specifically, MixMAS utilizes a sampling-based micro-benchmarking strategy to explore various combinations of modality-specific encoders, fusion functions, and fusion networks, systematically identifying the architecture that best meets the task's performance metrics.**

***Index Terms*—Multimodal Deep Learning, Architecture Search, MLP-Mixer, Multimodal Fusion.**

I. Introduction

The increasing complexity and diversity of data in various domains require the use of multimodal learning, which can leverage and integrate information from different modalities including text, image, audio, video, time series, etc. [1]. The application of multimodal learning spans a wide array of fields, including but not limited to, text-to-image generation, text-to-video synthesis, robotics, and autonomous driving [7]. The essence of multimodal learning lies in its ability to provide a more holistic understanding of the data by harnessing the complementary nature of different data types. However, the fusion of multimodal data presents significant computational and theoretical challenges [7] that arise from the inherent heterogeneity of the data sources, which makes learning intermodal relationships and representations more difficult, as each modality often exhibits diverse qualities, structures, and relevance to the task at hand. Computationally, processing and fusing such diverse data types at scale is very hardware demanding. This highlights the increasing need for specialized architectures that can effectively handle multimodal data.

In response to these challenges, Multi-Layer Perceptron (MLP) based architectures have emerged as a promising solution [4, 9, 12]. These architectures offer a compelling alternative to transformer models, as achieve a favorable balance between performance and computational complexity [12]. The advantages of these architectures lie in being computationally more efficient [12], exhibiting a simpler architectural design [8], facilitating ease of implementation and modification, and robustness in handling a variety of data types and tasks [8]. Integrating these components into an automated search pipeline enables us to leverage their benefits and design architectures that address the unique requirements and constraints of each task, resulting in more effective and specialized models.

In this paper, we present MixMAS, an automated framework for solving the problem of choosing which MLP-based architecture to use for a given multimodal machine learning task (MML). Our contributions can be summarized as follows: 1) We propose an automated pipeline for selecting the optimal MLP-based architecture for multimodal tasks. This pipeline benchmarks various MLP-based encoders for each modality, identifies the most effective fusion function for integrating all modalities, and determines the most suitable fusion network. Although we selected MLP-based architectures for their computational and conceptual simplicity, the framework is flexible and can accommodate other models, such as transformers, CNNs, and more. 2) We propose to employ a sampling approach, where different modules are benchmarked only on a small sample of the dataset to reduce the computational cost compared to evaluating on the full dataset. 3) We experimentally validate that our proposed pipeline for optimizing multimodal MLP-based architectures improves accuracy compared to standard MLP-based multimodal networks. 4) We open-source the code for the proposed framework at https://github.com/Madjid-CH/auto-mixer.

II. Related Work

This section reviews the related research on MLP-based models and multimodal architecture search methods.

MLP-Mixers [12] introduced a paradigm shift in the field of deep learning by achieving competitive results on image classification benchmarks against the state-of-the-art models with comparable computational resources. The architecture is based exclusively on MLPs with two types of layers: MLPs applied independently to image patches, and MLPs applied across patches. Many follow-up works improve the MLP-Mixer architecture, such as: Region-aware MLP (**RaMLP**) [6], which addresses the limitation of fixed input sizes in previous MLP models and captures both local and global visual cues in a region-aware manner; the **HyperMixer** [9] introduces a token mixing mechanism called HyperMixing, which uses hypernetworks to dynamically generate the weights of the token mixing MLP based on the input. This allows HyperMixer to handle variable input lengths and ensures systematicity by modeling interactions between tokens with shared weights across positions; the **Monarch-Mixer** [4] uses Monarch matrices for efficient performance on GPUs and demonstrates comparable or superior results in tasks like language modeling or image classification, with fewer parameters. These models are conceptually simpler compared to other architectures like CNNs and Transformers. They also make a good trade-off between performance and computational efficiency.

The M2-Mixer [2] architecture has been proposed for multimodal classification, leveraging the simplicity and efficiency of MLP-Mixers. It employs a multi-head loss function to address optimization imbalance, ensuring that no single modality dominates the learning process. This results in a conceptually and computationally simple model that outperforms baseline models on benchmark multimodal datasets, achieving higher accuracy and significantly reducing training time. However, MLP-Mixers are not universally effective across all modalities, and selecting the right MLP-based architecture to optimize performance can be challenging. To address this, our proposed pipeline systematically identifies the most suitable MLP-based network for each specific dataset, ensuring optimal performance and compatibility with the data characteristics.

Multimodal NAS: Neural Architecture Search (NAS) methods try to automate the process of finding the optimal architecture for given tasks and datasets. Several multimodal NAS approaches were presented in the literature using different search algorithm and search space [10, 13, 14, 15]. These approaches are often complex to implement and train, and adjusting certain parts of the architecture typically requires rerunning the entire NAS process. In contrast, our proposed framework is simple to implement and allows for the retention of existing micro-benchmarks, enabling quick updates to the architecture when new modules or components are added to the search space. Additionally, the modular design of the pipeline permits micro-benchmarking to be applied selectively to specific parts of the architecture, streamlining the process.

III. Our Framework

We propose **MixMAS**, a sampling-based framework for automatic mixer architecture search in multimodal learning. MixMAS efficiently selects optimal MLP-based architectures for various multimodal tasks, leveraging modularity and extensibility for adaptability. This pipeline focuses on efficient MLP architectures composed solely of matrix multiplications, transformations, and activation functions, streamlining the search process for suitable configurations in multimodal learning.

As illustrated in Figure 1, our pipeline can be conceptually divided into four main stages: 1) Sampling, 2) Encoder selection, 3) Fusion function selection, 4) Fusion Network selection.

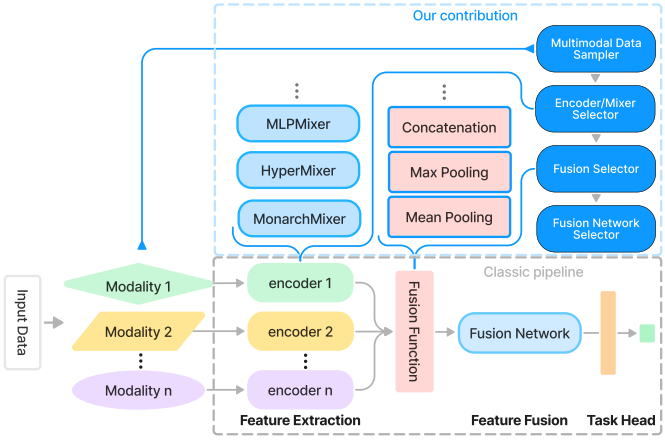

Fig. 1. MixMAS sampling based architecture search pipeline

Sampling: The sampling module selects a subset of the dataset for comparing the performance of different modules at each stage. It ensures that the sampled subset is representative of the entire dataset, which is essential for obtaining accurate and reliable performance metrics to guide the selection process. The number of samples is computed as in (1) using the sample size determination formula [5]:

$$N' = \frac{n}{1 + \frac{z^2 \times \hat{p}(1-\hat{p})}{\varepsilon^2 N}}, \quad n = \frac{z^2 \times \hat{p}(1-\hat{p})}{\varepsilon^2}, \tag{1}$$

where N is the size of the dataset, z is the z-score, $\hat{p}$ is the estimated proportion of the population that has the attribute of interest, and ε is the margin of error (i.e., 1%). We use random sampling to approximate the class distribution of the original dataset. To validate this approach, we calculate the distance between the class proportions in the original and sampled datasets, ensuring it remains below 0.05. In future work, we aim to implement uncertainty-based sampling to obtain more informative samples.

Encoder Selection involves benchmarking the performance of various MLP-based encoders on the sampled dataset for each modality. The choice of encoders depends on the dataset's modalities, ensuring all options are MLP-based. Users can customize the evaluation metrics within the framework to suit the problem's specific nature. For example, in a balanced

binary classification problem, accuracy might be preferred, while for imbalanced classes, metrics like precision, recall, or F1 score may be more appropriate. While MLP based encoders were selected for this work, the framework allows to incorporate arbitrary encoders, such as transformers, CNNs, RNNs, etc. The best encoder for each modality is then selected based on the benchmarking results to be used in the next stage.

Fusion Function Selection entails choosing the best fusion function to combine features from each modality. We use intermediate fusion, as raw fusion is often impractical with differing data structures, and late fusion may be suboptimal for highly correlated modalities [11]. The fusion function is evaluated based on classification performance.

Fusion Network Selection is responsible for selecting the appropriate network for encoding cross-modal information and preparing the last embedding for the task head. Similar to the Encoder Selection, multiple MLP-based networks will be benchmarked. The encoders for each modality and the fusion function are the ones fixed from the previous stages.

A recommended approach is to retain the micro-benchmarking scores after identifying the final architecture. By avoiding the need to re-benchmark previously evaluated architectures, this strategy streamlines the process when incorporating new components or making adjustments.

IV. Experiments

A. Experimental Setup

In this section, we describe the datasets we used in the experiments and the experimental setup.

MM-IMDB [1] is a multimodal dataset with images (movie posters) and text (plots) for genre classification. We used BERT [3] for text embeddings. **AV-MINST** [2] combines MNIST [3] images with FSDD [4] (digit pronunciations). **MIMIC-III** [5] is a clinical dataset with time-series (12 hourly medical measurements over 24 hours) and tabular data.

For micro benchmarks, we use a learning rate of 0.001, training with sampled data from the sampler for 10 epochs. For full training, we start with a learning rate of 0.001 on MM-IMDB, using a scheduler that reduces it by a factor of 10 if validation loss shows no improvement for 2 epochs. For AV-MINST and MIMIC-III, we follow the training setup of M2-Mixer [2]. For the MM-IMDB dataset, we compute the weighted F1 score due to the imbalanced labels. The other datasets were evaluated conducted using the accuracy metric.

We use MLP-Mixers, RaMLP [6], HyperMixer [9] and MonarchMixer [9] as candidates for encoder function, and HyperMixer and MLPMixer as candidates for Fusion Network. In MIMIC we opted for a fixed, simple feed-forward MLP as an encoder for the tabular modality. We compare MixMAS's performance against M2-Mixer.

[1]https://github.com/johnarevalo/gmu-mmimdb
[2]https://github.com/slyviacassell/_MFAS/tree/master
[3]https://yann.lecun.com/exdb/mnist/
[4]https://github.com/Jakobovski/free-spoken-digit-dataset
[5]https://physionet.org/content/mimiciii/1.4/

We utilized two internal clusters with NVIDIA GeForce RTX 3090, GeForce RTX 2080, A40 and V100 GPUS. The total runtime of all experiments was 144 hours.

B. Results

Table I summarizes the micro-benchmarking results, where the pipeline selects the highest-scoring modules to construct the final architecture. The results show that there is no universal solution for modality encoders or fusion networks, as different datasets and modalities benefit from distinct modules. This underscores the strength of the MixMAS pipeline in tailoring the architecture and fusion function to the task. Notably, ConcatFusion is consistently selected during the Fusion Function stage, validating our assumption that concatenation preserves more information from the modalities compared to mean or max pooling, improving overall model performance.

Table II presents the results of training the final model on full datasets. On MM-IMDB, MixMAS surpasses M2-Mixer, achieving an average F1-weighted score of 49.58% compared to M2-Mixer's 42.3%, with fewer parameters (10.37 million vs. 16.7 million). For AV-MNIST, MixMAS also outperforms M2-Mixer, achieving 75.79% average accuracy versus 73.2%. Results for MIMIC-III show similar performance between MixMAS and M2-Mixer. We hypothesize that the tabular modality's incompatibility with MLP-Mixers, alongside fixing a simple MLP for this modality, reduces the search space.

V. Conclusion

In this paper, we introduce MixMAS, a framework for selecting optimal MLP-based architectures using sampling and micro-benchmarking. Our approach builds on the simplicity and efficiency of MLP-Mixers, extending them to multimodal learning. Experiments confirm the framework's effectiveness on bi-modal datasets, and future work includes testing on datasets with more modalities. We also plan to explore alternative sampling methods, like uncertainty and diversity sampling, and expand the search to include a broader range of modules.

References

[1] T. Baltrušaitis, C. Ahuja, and L.-P. Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41 (2):423–443, 2018.

[2] G. Bezirganyan, S. Sellami, L. Berti-ÉQuille, and S. Fournier. M2-mixer: A multimodal mixer with multi-head loss for classification from multimodal data. In *2023 IEEE International Conference on Big Data (BigData)*, pages 1052–1058. IEEE, 2023.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2. Minneapolis, Minnesota, 2019.

[4] D. Y. Fu, S. Arora, J. Grogan, I. Johnson, E. S. Eyuboglu, A. W. Thomas, B. Spector, M. Poli, A. Rudra, and C. Ré. Monarch mixer: A simple sub-quadratic gemm-based architecture. In A. Oh, T. Naumann, A. Globerson,

TABLE I
MICRO BENCHMARK RESULTS FOR EACH STAGE. THE MODULE WITH THE HIGHEST SCORE WILL BE SELECTED.

MM-IMDB		**AV-MNIST**		**MIMIC-III**	
Sampling(%)	23%	12%		21%	
Module	Score F1-w(%)	Module	Score Acc(%)	Module	Score Acc(%)
Image Encoder Selection		**Image Encoder Selection**		**Time-Series Encoder Selection**	
MLPMixer	**24.02**	MLPMixer	44.27	MLPMixer	40.77
HyperMixer	16.89	**HyperMixer**	**56.15**	**HyperMixer**	**45.36**
RaMLP	14.44	RaMLP	47.52	MonarchMixer	44.38
Text Encoder Selection		**Audio Encoder Selection**		**Tabular Encoder Selection**	
MLPMixer	9.20	MLPMixer	27.40	—	—
HyperMixer	15.07	**HyperMixer**	**29.16**	—	—
MonarchMixer	**28.55**	MonarchMixer	28.49	—	—
Fusion Function Selection		**Fusion Function Selection**		**Fusion Function Selection**	
ConcatFusion	**19.56**	**ConcatFusion**	**18.38**	**ConcatFusion**	**28.55**
MeanFusion	10.20	MeanFusion	9.61	MeanFusion	4.28
MaxFusion	9.07	MaxFusion	6.20	MaxFusion	6.73
Fusion Network Selection		**Fusion Network Selection**		**Fusion Network Selection**	
HyperMixer	**29.0**	**HyperMixer**	**53.47**	**HyperMixer**	**38.15**
MLPMixer	25.97	MLPMixer	42.17	MLPMixer	34.14

TABLE II
RESULTS ON **MM-IMDB**, **AV-MNIST** AND **MIMIC-III** DATASETS.

Architecture	**MM-IMDB**		**AV-MNIST**		**MIMIC-III**	
	F1-w. (%) (avg)	Training Params (M)	Acc. (%) (avg)	Training Params (M)	Acc. (%) (avg)	Training Params (M)
M2-Mixer	46.66 ± 0.44	16.7	73.20 ± 0.2	8.3	78.32 ± 0.3	0.029
MixMAS	$\mathbf{49.58 \pm 0.5}$	10.37	$\mathbf{75.79 \pm 0.3}$	9.33	78.3 ± 0.73	0.033

K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems 36*, 2023.

[5] R. Hogg, E. Tanis, and D. Zimmerman. *Probability and Statistical Inference*. Pearson, 2013. ISBN 9780321923271.

[6] S. Lai, X. Du, J. Guo, and K. Zhang. Ramlp: Vision mlp via region-aware mixing. In *IJCAI*, pages 999–1007, 2023.

[7] P. P. Liang, A. Zadeh, and L.-P. Morency. Foundations and recent trends in multimodal machine learning: Principles, challenges, and open questions. *CoRR*, abs/2209.03430, 2022.

[8] R. Liu, Y. Li, L. Tao, D. Liang, and H. Zheng. Are we ready for a new paradigm shift? a survey on visual deep mlp. *Patterns (N Y)*, 3(7):100520, 2022. doi: 10.1016/j.patter.2022.100520.

[9] F. Mai, A. Pannatier, F. Fehr, H. Chen, F. Marelli, F. Fleuret, and J. Henderson. Hypermixer: An mlp-based low cost alternative to transformers, 2023.

[10] J.-M. Pérez-Rúa, V. Vielzeuf, S. Pateux, M. Baccouche, and F. Jurie. Mfas: Multimodal fusion architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6966–6975, 2019.

[11] S. R. Stahlschmidt, B. Ulfenborg, and J. Synnergren. Multimodal deep learning for biomedical data fusion: a review. *Briefings Bioinform.*, 23(2), 2022.

[12] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, and A. Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *Advances in Neural Information Processing Systems 34*, pages 24261–24272, 2021.

[13] Z. Xu, D. R. So, and A. M. Dai. Mufasa: Multimodal fusion architecture search for electronic health records. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10532–10540, 2021.

[14] Y. Yin, S. Huang, and X. Zhang. Bm-nas: Bilevel multimodal neural architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8901–8909, 2022.

[15] Z. Yu, Y. Cui, J. Yu, M. Wang, D. Tao, and Q. Tian. Deep multimodal neural architecture search. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3743–3752, 2020.