IMPROVING MUSIC EMOTION RECOGNITION BY LEVERAGING HANDCRAFTED AND LEARNED FEATURES

Pedro Lima Louro[1] **Hugo Redinho**[1] **Ricardo Malheiro**[1,2]
Rui Pedro Paiva[3] **Renato Panda**[1,3]
[1] Centre for Informatics and Systems of the University of Coimbra (CISUC),
Department of Informatics Engineering, LASI, University of Coimbra, Portugal
[2] Polytechnic Institute of Leiria, School of Technology and Management, Portugal
[3] Ci2 — Smart Cities Research Center Polytechnic Institute of Tomar, Portugal
pedrolouro@dei.uc.pt, redinho@student.dei.uc.pt, {rsmal, ruipedro, panda}@dei.uc.pt

ABSTRACT

Music Emotion Recognition was dominated by classical machine learning, which relies on traditional classifiers and feature engineering (FE). Recently, deep learning approaches have been explored, aiming to remove the need for handcrafted features by automatic feature learning (FL), albeit at the expense of requiring large volumes of data to fully exploit their capabilities. A hybrid approach fusing information from handcrafted and learned features was previously proposed, outperforming separate FE and FL approaches on the 4QAED dataset (900 audio clips). The results suggested that, in smaller datasets, FE and FL could complement each other rather than act as competitors. In the present study, these experiments are extended to the larger MERGE dataset (3554 audio clips) to analyze the impact of the significant increase in data. The best-obtained results, 77.62% F1-score, continue to surpass the standalone FE and FL paradigms, reinforcing the potential of hybrid approaches.

1. INTRODUCTION

Recently, several Deep Learning (DL) approaches have been proposed to address research problems in Music Emotion Recognition (MER). These eliminate the necessity of feature engineering efforts since DL architectures can automatically learn relevant features from the input data.

However, as pointed out in a previous study [1], the current state-of-the-art DL approaches are still underperforming compared to classical MER approaches using handcrafted features. The lack of sizeable and quality MER datasets is part of the problem since DL architectures can only reach their full potential with a large, representative set of samples for the problem at hand, which usually takes hundreds of thousands of samples. Furthermore, the features learned by a neural network depend on the data provided. Emotionally-relevant patterns may be missed if they are rare in the dataset, unlike handcrafted features, which can target specific characteristics even if they appear infrequently, though this might not improve classification performance.

To take advantage of the strengths of both the classical and DL paradigms, a hybrid methodology was previously developed and validated on a set of small datasets (containing 1372 samples), showing promising results. This methodology surpassed all classical and neural network-based baselines [1].

In this article, we further extend the evaluation of this hybrid methodology to two new datasets proposed by Louro et al. [2], containing 3554 samples.

2. RELATED WORK

In this section, we briefly review the state-of-the-art approaches relevant to this study. Both classical ML and DL-based methodologies are discussed, concluding with a summary of the advantages and disadvantages of each.

Seminal works in MER follow a common pipeline. First, a set of songs is collected and manually annotated by subjects, followed by the extraction of features relevant to emotion, and finally, training and evaluating a classifier, such as Support Vector Machines (SVM) or Random Forest, to name a few. Feng et al. [3] presents such a pipeline, with a slight difference regarding the song annotations. Instead of manual annotations, these are automatically obtained through predefined intervals of the extracted features, e.g., an excerpt with legato as the predominant articulation and a slow tempo labeled as sad.

Subsequent works improve on the oversights of this approach and explore the problem in other directions. Some of these works include Lu et al. [4], which keep the single label approach but use a Gaussian Mixture Model to classify samples based on intensity, timbre and rhythm features, and Yang et al. [5], defining MER as a regression problem to mitigate the ambiguities inherent to the discrete labels from the previously mentioned approaches.

Recently, Panda et al. [6] proposed a set of new audio

Attribution: P. L. Louro, H. Redinho, R. Malheiro, R. P. Paiva, R. Panda, "Improving Music Emotion Recognition By Leveraging Handcrafted and Learned Features", in *Proc. of the 1st Latin American Music Information Retrieval Workshop,* Rio de Janeiro, Brazil, 2024.

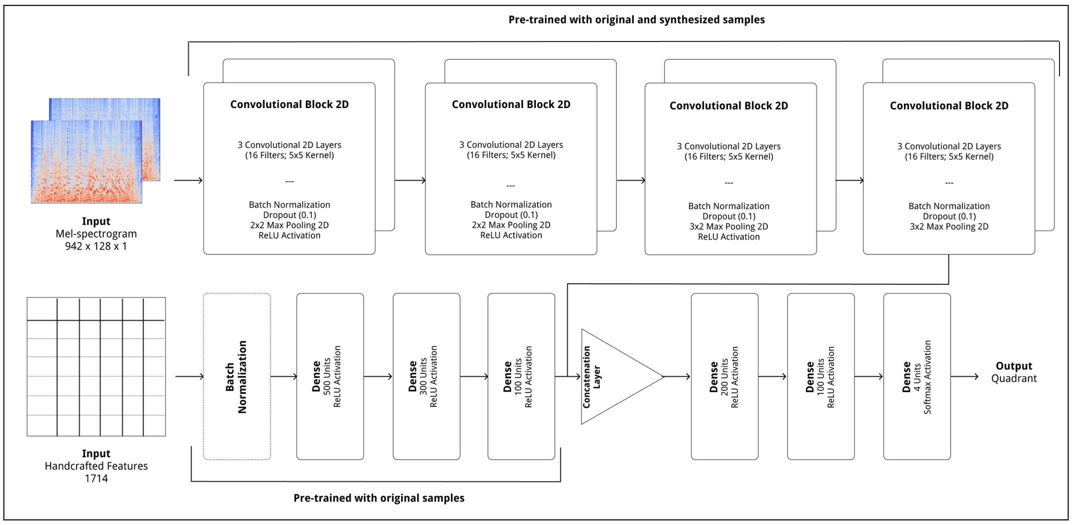

Figure 1. Hybrid augmented architecture. The architecture can be decomposed into the frontend, or the CNN and DNN portions where features are learned or processed, respectively, and the backend, further processing the concatenated features and outputting the predicted quadrant.

features alongside a small but thoroughly validated balanced dataset of 900 song excerpts and a state-of-the-art methodology focused on classical static MER. There, the 4 Quadrant Audio Emotion Detection (4QAED) dataset was built with the aid of a semi-automatic approach, building on user-generated labels from AllMusic, and manual validation. Features were extracted for each excerpt and ranked, from which only the top 100 were used to train an SVM classifier. The observed results are considerably higher than those previously reported, attaining a 76.4% F1-score.

As discussed previously, neural networks have the ability to automatically learn the most relevant features from the input data. Such an idea is very appealing to any Music Information Retrieval (MIR) problem, considering the hardship of developing and validating features by hand. To our knowledge, the first application of these models to MIR was presented by Choi et al. [7]. Here, the experiments used only convolutional layers, learning and processing the learned features from Mel-spectrogram representations of the considered datasets for validation. Later, the same authors presented a more complex network, consisting of a convolutional portion for feature learning, and a recurrent portion for processing time-related features and performing classification, referred to as Convolutional Recurrent Neural Network (CRNN) [8]. The final system, trained for multi-label classification, attained a 0.86 Area Under the ROC Curve (AUC), outperforming the other proposed architectures.

Several approaches built on this system, iterating mostly on certain aspects of its architecture. Some of these include musically-motivated filters applied to the convolutional portion of the network, focusing on finding timbral and temporal information [9], and end-to-end architectures, which aim to learn the most relevant features directly from the raw audio waveform [10].

Regarding MER specifically, many works build upon the previously described system's pipelines, experimenting with different data representations such as chromagram [11] and conchleogram [12], applying transfer learning from related tasks such as speech [13], and experimenting with smaller input sizes [14].

This work builds on Panda et al. [6] and Choi et al. [7] for the FE and FL portions respectively, given their impact in the field.

3. MATERIALS AND METHODOLOGY

This section describes the methodology followed in this study, starting with the pre-processing steps, then describing the architecture details, and ending with the optimization strategy.

Dataset	Q1	Q2	Q3	Q4	Total
MERGE Audio C	875	915	808	956	3554
MERGE Audio B	808	808	808	808	3232
MERGE Bimodal C	525	673	500	518	2216
MERGE Bimodal B	500	500	500	500	2000

Table 1. Datasets used and their distribution per quadrant.

3.1 Datasets

The methodology was evaluated using two datasets: MERGE Audio and MERGE Bimodal, which includes a complete and a balanced collection of samples for each dataset, detailed in [2]. MERGE Audio contains 3554 and 3232 samples, while the MERGE Bimodal comprises 2216 and 2000 samples. The quadrant distribution of each is detailed in Table 1.

Each dataset entry includes a 30-second audio clip of the most emotionally-representative part of the song. Samples were annotated into one of four emotion quadrants (happy, tense, sad, and relaxed), according to Russell's Circumplex model [15]. While the MERGE Bimodal dataset includes the full lyrics for each sample, this research does not explore the lyrical content.

A 70-15-15 train-validate-test (TVT) split was used as our validation strategy, as recommended in [2].

3.2 Pre-processing Steps

Initially, the audio samples are converted into WAV format. To obtain Mel-spectrograms, these samples are downsampled from 22.5 kHz to 16kHz, as per the methodology in [1].

The handcrafted features are extracted from all samples using MIRToolbox [16], Marsyas [17], and PsySound3 [18] audio frameworks, complemented by the novel features proposed by Panda et al. [6]. A final set of 1714 features is obtained after performing feature decorrelation, i.e., eliminating redundant features that would not contribute to increasing the model's performance.

As for the input data for the CNN portion, the librosa library [19] is used to obtain Mel-spectrogram representations. The library's default settings for the Fast Fourier Transform window length (2048) and the hop size (512) are used to generate the spectral representations.

Data augmentation is also performed on the train set of each dataset when optimizing the CNN portion. This is done by applying time shifting (shifts the start or the end of the audio clip by a maximum of 5 seconds), pitch shifting (increases or decreases the pitch by a maximum of 2 semitones), time stretching (speeds up or slows down an audio clip by a maximum of 50%), and power shifting (increases or decreases amplitude by a maximum of 10 dB) to each audio clip. Since each transformation is applied individually, the train set essentially increases five-fold.

3.3 Architecture Details

The architecture, illustrated in Figure 1, comprises a CNN and DNN portion for feature learning and processing, respectively, and a smaller DNN portion for classification.

The CNN portion, based on the mentioned work by Choi et al. [7], comprises four convolutional blocks, each containing a sequence of Batch Normalization, Dropout, and Max Pooling layers, ending with a ReLU activation layer. The last convolutional block does not contain the Dropout layer. The previously discussed Mel-spectrograms are fed as input to this portion.

The resulting output of both is concatenated at the feature level before being fed to the classifier, a set of three Dense layers, with the last one outputting one of the four quadrants of Russell's Circumplex model [15]. This way, the classifier could pick the set of patterns that are most relevant to the problem at hand.

The CNN portion's training phase includes synthesized samples to improve its performance. Therefore, this and the DNN for feature processing are pre-trained separately, freezing their weights before training the classification portion.

	Best Hyperparameters		
Dataset	Batch	Optimizer	Learning
	Size		Rate
MERGE	32	SGD	1e-2
Audio	128	SGD	1e-2
Complete	16	Adam	1e-2
MERGE	32	SGD	1e-2
Audio	128	Adam	1e-4
Balanced	32	SGD	1e-3
MERGE	32	SGD	1e-2
Bimodal	128	SGD	1e-2
Complete	64	SGD	1e-4
MERGE	32	Adam	1e-3
Bimodal	32	SGD	1e-2
Balanced	64	Adam	1e-3

Table 2. Optimal hyperparameters per dataset. For each, the optimal values for standalone CNN and DNN portions are shown, followed by the final classifier optimization.

3.4 Optimization Strategy

The model optimization was carried out with the Bayesian optimization approach provided by the Keras Tuner library [20]. This technique searches for the optimal combination of hyperparameters within predefined ranges, aiming to maximize or minimize a specific objective function defined by the user.

The tuner's objective is to maximize the validation set's accuracy. The optimal values for each hyperparameter, including batch size, optimizer, and the respective learning rate, are detailed in Table 2.

The process involves running ten trials, beginning at the lower end of the specified intervals. For every trial, the model undergoes training up to a maximum of 200 epochs. However, an early stopping mechanism fires if the validation accuracy does not improve for 15 straight epochs or if the training accuracy exceeds 90%. This approach greatly decreases the time required for the optimization by reducing the time spent on hyperparameters that show poor performance, also preventing overfitting.

We used the 70-15-15 train-validate-test (TVT) split defined in [2] as our validation strategy. The resulting models for each trial are backed up for later usage, including the evaluation phase, which is discussed next.

In the TVT strategy, the optimization function uses both training and validation sets to identify the best hyperparameters. Once the model is trained using these, it undergoes evaluation on the test set. This evaluation involves calculating the F1-score, Precision, and Recall by comparing the actual values with the model's predictions for each category and assessing the model's overall performance.

4. RESULTS AND DISCUSSION

The results and gathered insights are presented in this section. We begin by highlighting the most relevant results from the previously presented metrics, followed by a discussion on the improvements and drawbacks of applying

Dataset	F1-score	Precision	Recall
MERGE Audio Complete	68.84%	69.52%	68.80%
MERGE Audio Balanced	**77.62%**	**78.11%**	**77.89%**
MERGE Bimodal Complete	73.13%	75.45%	74.40%
MERGE Bimodal Balanced	70.00%	69.99%	70.33%

Table 3. TVT 70-15-15 results for the mentioned datasets

		Predicted			
		Q1	Q2	Q3	Q4
Actual	Q1	**70.2%**	10.9%	9.4%	9.4%
	Q2	7.1%	**92.1%**	0.8%	0.0%
	Q3	3.9%	2.6%	**55.3%**	38.2%
	Q4	18.3%	0.9%	21.7%	**59.13%**

Table 4. Confusion matrix for MERGE Audio Complete

		Predicted			
		Q1	Q2	Q3	Q4
Actual	Q1	**77.9%**	7.6%	3.8%	10.9%
	Q2	6.0%	**93.2%**	0.9%	0.0%
	Q3	2.8%	1.4%	**68.8%**	27.0%
	Q4	8.4%	0.0%	18.9%	**72.6%**

Table 5. Confusion matrix for MERGE Audio Balanced

		Predicted			
		Q1	Q2	Q3	Q4
Actual	Q1	**73.9%**	5.7%	6.8%	13.6%
	Q2	7.1%	**92.9%**	0.0%	0.0%
	Q3	5.7%	0.9%	**58.5%**	34.9%
	Q4	5.1%	0.0%	23.1%	**71.8%**

Table 6. Confusion matrix for MERGE Bimodal Complete

		Predicted			
		Q1	Q2	Q3	Q4
Actual	Q1	**74.7%**	5.3%	8.0%	12.0%
	Q2	9.1%	**90.9%**	0.0%	0.0%
	Q3	3.5%	1.2%	**58.8%**	36.5%
	Q4	14.3%	0.0%	30.2%	**55.6%**

Table 7. Confusion matrix for MERGE Bimodal Balanced

the hybrid methodology to the MERGE Audio and Bimodal datasets.

The best F1-score attained was 77.62% on the MERGE Audio Balanced dataset, as seen in Table 3. Again, the model's performance is shown to be particularly susceptible to quadrant balancing since the lowest result is observed when using the largest but most unbalanced of the validation datasets.

From previous experiments and according to the literature, one of the biggest challenges of audio-only approaches is to accurately differentiate valence when arousal is low, i.e., confusion between the third and fourth quadrants of Russell's Circumplex model. As observed in Table 4, this is still present in this model when considering MERGE Audio Complete, also with some considerable confusion between the first and fourth quadrants. Using the balanced counterpart, as seen in Table 5, the confusion is reduced considerably in the third quadrant. This improvement is very significant given that it is the quadrant that produces the most confusion, even for human annotators. The fourth quadrant also improves significantly, a consequence of less confusion with the first quadrant.

There are some caveats to consider, such as the overall higher results for MERGE Bimodal Complete against Bimodal Balanced. Although this contradicts the previous idea that quadrant distribution is essential for this model, this could be explained by less disparity between the number of samples of the third and fourth quadrants compared to MERGE Audio Complete. This is further corroborated by the confusion matrices in Tables 6 and 7, as the most significant difference is the performance of the fourth quadrant.

5. CONCLUSION AND FUTURE WORK

The Hybrid Augmented methodology is further experimented with in the present study. Due to the promising results of the fusion of handcrafted and learned features, we conducted further experiments on larger datasets, namely the complete and balanced versions of MERGE Audio and MERGE Bimodal. Each portion of the architecture is trained independently, first pre-training the CNN portion for feature learning and the DNN portion for feature processing, with additional synthesized samples added to the optimization phase of the former. The optimal weights for each portion are frozen, finally optimizing the classification portion.

The best result from these datasets is a 77.62% F1-score, attained with MERGE Audio Balanced. This was expected since previously reported results indicate the importance of large datasets with even distribution between quadrants for optimal performance. The confusion matrices for the MERGE Audio datasets further corroborate this conclusion, as low arousal quadrants are more easily distinguished in the balanced version of these. There are some inconsistencies, such as the higher results in the complete version of the MERGE Bimodal datasets, which may be due to a smaller gap between the number of samples of the third and fourth quadrants.

Regarding the methodology, it would be beneficial to analyze further the impact of new data augmentation techniques applied to the CNN portion of the model. It would also be beneficial to experiment with optimizing the DNN portion of the network with the same synthesized data of the CNN counterpart. Finally, the classifier could be further enhanced by including recurrent layers, such as in the CRNN architecture, to process time-related features from the previously processed information.

6. ACKNOWLEDGMENTS

This work is funded by FCT - Foundation for Science and Technology, I.P., within the scope of the projects: MERGE - DOI: 10.54499/PTDC/CCI-COM/3171/2021 financed with national funds (PIDDAC) via the Portuguese State Budget; and project CISUC - UID/CEC/00326/2020 with funds from the European Social Fund, through the Regional Operational Program Centro 2020. Renato Panda was supported by Ci2 - FCT UIDP/05567/2020.

7. REFERENCES

[1] P. L. Louro, H. Redinho, R. Malheiro, R. P. Paiva, and R. Panda, "A Comparison Study of Deep Learning Methodologies for Music Emotion Recognition," *Sensors*, vol. 24, no. 7, p. 2201, 2024.

[2] P. L. Louro, H. Redinho, R. Santos, R. Malheiro, R. Panda, and R. P. Paiva, "MERGE – A Bimodal Dataset for Static Music Emotion Recognition," Jul. 2024.

[3] Y. Feng, Y. Zhuang, and Y. Pan, "Popular music retrieval by detecting mood," in *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2003, pp. 375–376.

[4] L. Lu, D. Liu, and H.-J. Zhang, "Automatic mood detection and tracking of music audio signals." *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, pp. 5–18, 2006.

[5] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A Regression Approach to Music Emotion Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 448–457, 2008.

[6] R. Panda, R. Malheiro, and R. P. Paiva, "Novel Audio Features for Music Emotion Recognition," *IEEE Transactions on Affective Computing*, vol. 11, no. 4, pp. 614–626, Oct. 2020.

[7] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, 2016, pp. 805–811.

[8] K. Choi, G. Fazekas, M. Sandler, and K. Cho, "Convolutional recurrent neural networks for music classification," in *Proceedings of the 2017 International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 2392–2396.

[9] J. Pons, O. Nieto, M. Prockup, E. M. Schmidt, A. F. Ehmann, and X. Serra, "End-to-end learning for music audio tagging at scale," in *Proceedings of the 19th International Society for Music Information Retrieval Conference*, 2018, pp. 637–644.

[10] J. Lee, J. Park, K. Kim, and J. Nam, "SampleCNN: End-to-End Deep Convolutional Neural Networks Using Very Small Filters for Music Classification," *Applied Sciences*, vol. 8, no. 1, p. 150, Jan. 2018.

[11] M. Bilal Er and I. B. Aydilek, "Music Emotion Recognition by Using Chroma Spectrogram and Deep Visual Features:," *International Journal of Computational Intelligence Systems*, vol. 12, no. 2, p. 1622, 2019.

[12] M. Russo, L. Kraljević, M. Stella, and M. Sikora, "Cochleogram-based approach for detecting perceived emotions in music," *Information Processing & Management*, vol. 57, no. 5, p. 102270, Sep. 2020.

[13] J. S. Gomez Canon, E. Cano, P. Herrera, and E. Gomez, "Transfer learning from speech to music: Towards language-sensitive emotion recognition models," in *2020 28th European Signal Processing Conference (EUSIPCO)*. Amsterdam, Netherlands: IEEE, Jan. 2021, pp. 136–140.

[14] S. Hizlisoy, S. Yildirim, and Z. Tufekci, "Music emotion recognition using convolutional long short term memory deep neural networks," *Engineering Science and Technology, an International Journal*, vol. 24, no. 3, pp. 760–767, Jun. 2021.

[15] J. A. Russell, "A circumplex model of affect." *Journal of Personality and Social Psychology*, vol. 39, no. 6, pp. 1161–1178, Dec. 1980.

[16] O. Lartillot, "MIR Toolbox 1.8.1 User's Manual," 2021.

[17] D. Bogdanov, N. Wack, E. Gomez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. Zapata, and X. Serra, "Essentia: An Audio Analysis Library for Music Information Retrieval," in *14th International Society for Music Information Retrieval Conference*, 2013.

[18] D. Cabrera, S. Ferguson, and E. Schubert, "'PsySound3': Software for Acoustical and Psychoacoustical Analysis of Sound Recordings," in *G. P. Scavone (Ed.), 13th International Conference on Auditory Display*, 2007, pp. 356–363.

[19] B. McFee, C. Raffel, D. Liang, D. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "Librosa: Audio and Music Signal Analysis in Python," in *Python in Science Conference*, Austin, Texas, 2015, pp. 18–24.

[20] T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi *et al.*, "Keras Tuner," https://github.com/keras-team/keras-tuner, 2019.