

MERGE App: A Prototype Software for Multi-user Emotion-aware Music Management

Pedro Lima Louro¹^a, Guilherme Branco¹^b, Hugo Redinho¹^c, Ricardo Correia¹^d,
Ricardo Malheiro^{1,2}^e, Renato Panda^{1,3}^f, Rui Pedro Paiva¹^g

¹University of Coimbra, Centre for Informatics and Systems of the University of Coimbra (CISUC), Department of Informatics Engineering, and LASI

²Polytechnic Institute of Leiria School of Technology and Management

³Ci2 — Smart Cities Research Center, Polytechnic Institute of Tomar

pedrolouro@dei.uc.pt, guilherme.m.branco@tecnico.ulisboa.pt, redinho@student.dei.uc.pt, {ricardocorreia, rsmal, panda, ruipedro}@dei.uc.pt

Keywords: Music Information Retrieval; Music Emotion Recognition; Machine Learning; Deep Learning; Software

Abstract: We present a prototype software for multi-user music library management using the perceived emotional content of songs. The tool offers music playback features, song filtering by metadata, and automatic emotion prediction based on arousal and valence, with the possibility of personalizing the predictions by allowing each user to edit these values based on their own emotion assessment. This is an important feature for handling both classification errors and subjectivity issues, which are inherent aspects of emotion perception. A path-based playlist generation function is also implemented. A multi-modal audio-lyrics regression methodology is proposed for emotion prediction, with accompanying validation experiments on the MERGE dataset. The results obtained are promising, showing higher overall performance on train-validate-test splits (73.20% F1-score with the best dataset/split combination).

1 INTRODUCTION

The digital era has brought an unprecedented amount of music right at our fingertips through digital marketplaces and streaming services. With the sudden availability of millions of songs to users, the necessity to automatically organize and find relevant music emerged. Current recommendation systems provide personalized suggestions to users based on listening patterns and using tags, such as genre, style, etc. However, options are lacking when we consider recommendations based on the automatic analysis of the emotional content of songs.

The field of Music Emotion Recognition (MER) has seen considerable advances in recent years in terms of the more classical approaches. Panda et al.

(2020) proposed a new set of features that considerably increased the performance of these systems, achieving a 76.4% F1-score with the top 100 ranked features. Although the feature evaluation is limited to one dataset, the improvements are significant compared to the best results from similar systems that reached a glass ceiling Hu et al. (2008).

One drawback of audio-only methodologies is their shortcomings when differentiating valence. Various systems have been proposed using a bimodal approach leveraging both audio and lyrics, attaining considerable improvements when compared to systems using only one or the other Delbouys et al. (2018); Pyrovolakis et al. (2022). Such systems have also implemented Deep Learning (DL) architectures to skip the time-consuming feature engineering and extraction steps from the classical systems and considerably speed up the inference process of the overall system.

In this study, we present the MERGE¹ application,

^a <https://orcid.org/0000-0003-3201-6990>

^b <https://orcid.org/0000-0003-4073-1716>

^c <https://orcid.org/0009-0004-1547-2251>

^d <https://orcid.org/0000-0001-5663-7228>

^e <https://orcid.org/0000-0002-3010-2732>

^f <https://orcid.org/0000-0003-2539-5590>

^g <https://orcid.org/0000-0003-3215-3960>

¹MERGE is the acronym of "Music Emotion Recognition nExt Generation", a research project funded by the Portuguese Science Foundation.

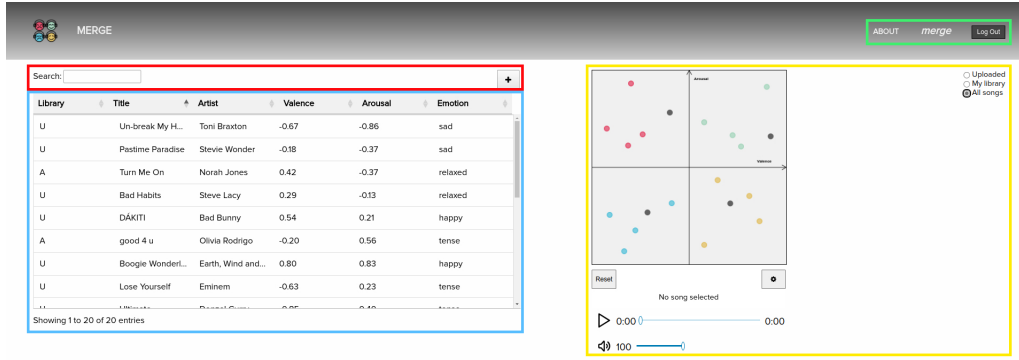


Figure 1: MERGE application interface. The AV plot, alongside music playback and song display controls, is seen in yellow. The table view is highlighted in blue with red highlight filtering search bar and button for adding music. Finally, green highlights the buttons for application information and user logout.

which automatically predicts the arousal and valence of songs based on Russell’s Circumplex Model Russell (1980). Two axes make up this model: arousal (Y-axis), which depicts whether the song has high or low energy, and valence (X-axis), which represents whether the emotion of the song has a negative or positive connotation.

The integrated model used for prediction is also presented in this study alongside validation experiments, which received both audio and lyrics information to map the song more accurately into the above mentioned model.

The MERGE application is a follow-up of the MOODetector application, previously created by our team Cardoso et al. (2011). The new MERGE app was built from scratch, with significant code refactoring and optimization, while keeping the overall user interface of the MOODetector app. In addition, significant novel features and improvements were implemented, namely: i) a bimodal app, which exploits the combination of audio and lyrics data for improved classification (unlike the single audio modality in the MOODetector app); ii) an improved classification model, training with the MERGE dataset Louro et al. (2024b) and following a deep learning approach Louro et al. (2024a); iii) and a shift from the monolithic single-user paradigm to the web-based multi-user paradigm.

2 MERGE APPLICATION

The MERGE application is implemented using JavaScript, with the addition of the jQuery library to handle AJAX, for its frontend, while the backend is served using the Express library on top of Node.js. The application interface is depicted in Figure 1.

2.1 Application Overview

The components can be broken down as: i) the Russell’s Circumplex model where all songs can be seen (highlighted in yellow); ii) a table view of the songs with various options for sorting (highlighted in blue); iii) options to filter and add new songs (seen in the red region, from left to right); iv) information about the application, the current user, and an option to logout (highlighted in green, also from left to right).

Songs are placed in the plot described in i) according to the estimated arousal and valence (AV) values in an interval of $[-1, 1]$ for each axis. The process for obtaining these values is described in Section 3. Beyond the AV positioning, each point on the plane is also color-coded depending on the quadrant: green (happy), red (tense), blue (sad), and yellow (relaxed).

The view of the graph can be switched between a) “Uploaded” to show only songs uploaded by the current user, b) “My Library” to display a user’s library, i.e., the songs uploaded by the current user, plus songs uploaded by other users added by the current user, and c) “All songs” to show all the songs available in the database. A note regarding the latter option is the differentiation of songs not added by the user, appearing as grey dots in the plot, also depicted in Figure 1.

Each user can change the song’s position directly by moving the point in the plane view, or by editing the AV values through the table view. These new AV values are unique to the user.

The application can be used as an audio playback software, thus offering usual features such as:

- Playback controls for mp3 files (play, pause, seek);
- Volume controls, including mute;
- Double-clicking a song to be played either in the plot or table view;

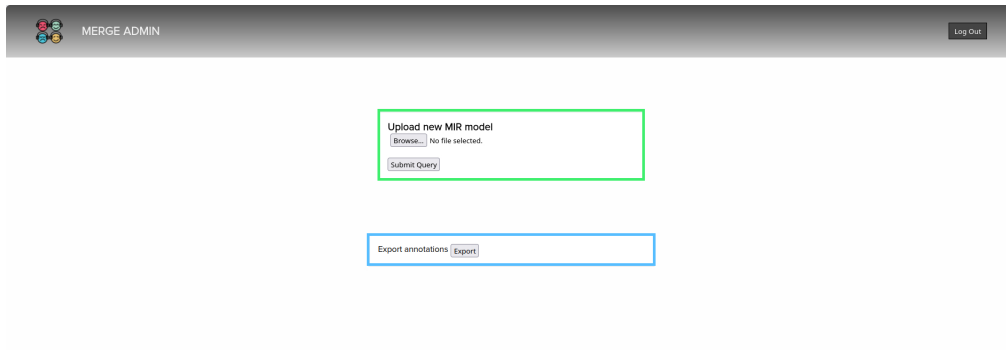


Figure 2: MERGE App Backoffice. Users with administrative privileges can overwrite the model used for AV values’ prediction (highlighted in green) and export a CSV file with information regarding the annotations for all users to each song in the database (highlighted in blue).

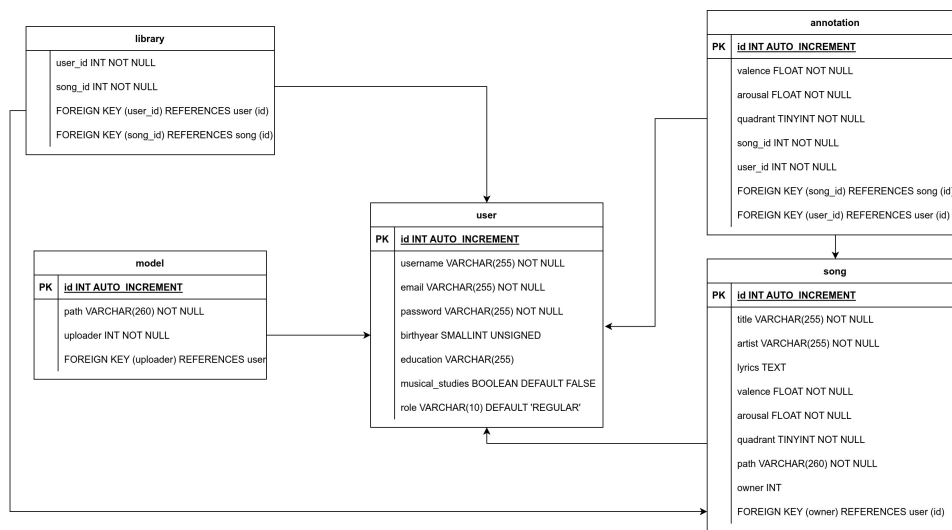


Figure 3: Entity relation diagram for the application’s database.

- Filtering and sorting by any of the available song properties (title, artist, valence, arousal, emotion);
- Adding and deleting songs from the user’s library.

The application also provides a backoffice for users with administrative privileges, pictured in Figure 2. After logging in, the user can perform one of two actions: upload and deploy a new model for AV prediction, and export a CSV with the existing user AV annotations. The latter option is designed to easily retrieve each user’s available annotations for songs in their respective libraries. In this way, the MERGE app can be used as a crowdsourcing data collection and annotation tool, promoting the creation of sizeable and quality MER datasets, a current key need in MER research Panda et al. (2020).

Regarding the database used to store all the relevant data from users and songs, the corresponding entity relationship diagram is depicted in Figure 3. The user table stores the user’s personal information,

as well as the user role, for access privileges purposes. The path for the model used for AV prediction is saved in the corresponding table and identifies the user that uploaded the currently deployed model. The song table stores all song-related information, including metadata, the AV values first predicted by the presently deployed model, the mapped quadrant in the plot view, the path for the uploaded audio clip, and the reference to the user who first added the song.

The user-specific annotations are stored in the annotation table, which stores the AV values defined per the user’s perception, the corresponding quadrant, and a reference to the song and user for that annotation. Finally, the library table stores all user libraries, containing only songs added by the user, either through uploading or from other users’ libraries.

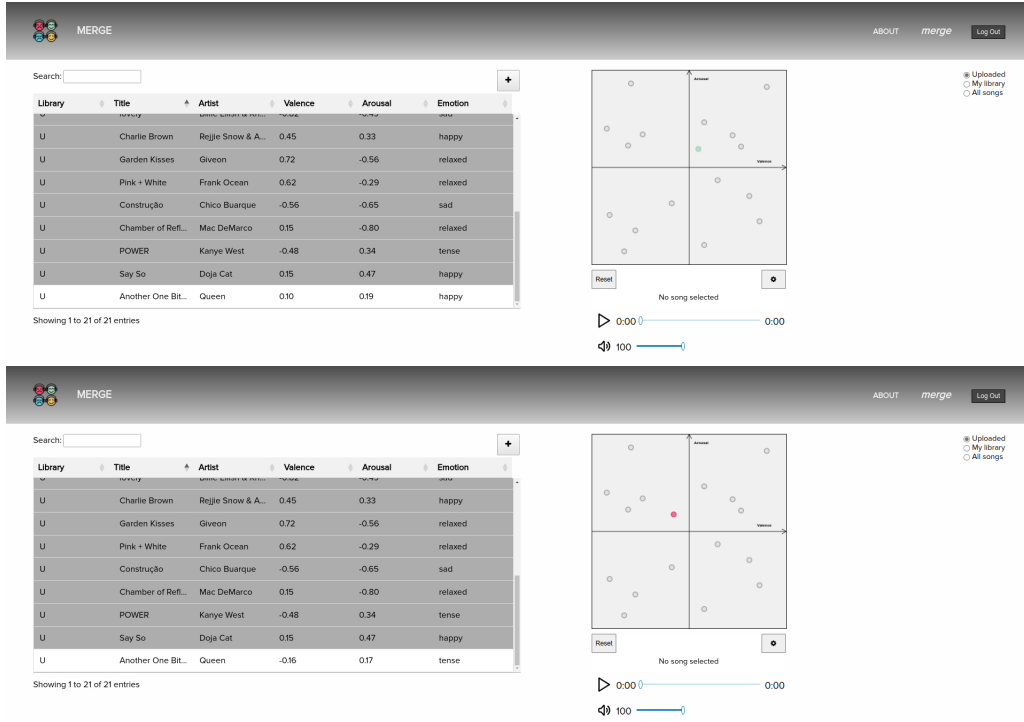


Figure 4: At the top, "Another One Bites The Dust" by Queen is added to the library and placed in the plane according to the predicted AV values. At the bottom, the point representing the song is moved to a more accurate position, according to the user.

2.2 Building An Emotionally-aware Library

After adding a new song from an available MP3 file, AV values are automatically predicted, and a point is added to the plot alongside a new entry on the table. Should the user disagree with the predicted values, these can be easily changed by editing the entry from the table view or moving the point in the plot. An example of the initially predicted position for a newly added song and the final position after adjustment can be seen in Figure 4. This personalization mechanism provides users with the ability to address the intrinsic subjectivity in MER. However, tackling this issue continues to present a significant challenge.

2.3 Path-based Playlist Generation

The ability to generate a playlist based on a user-drawn path is currently implemented, as depicted in Figure 5. This feature allows users to freely create an emotionally-varying playlist.

This is done by computing the distance of the user-defined N closest songs to the reference points that make up the drawn path. The user may also configure how far the songs can be from the path to be consid-

ered into the calculations. This threshold is defined as a decimal number between the plane interval $([-1, 1])$.

3 SONG EMOTION PREDICTION

In this section, we discuss the methodology used to predict AV values for a given song. First, the DL model's architecture is presented, followed by the pre-processing steps for each modality, a description of the optimization used, and the evaluation conducted.

3.1 Model Architecture

The proposed architecture, depicted in Figure 6, is based on the one by Delbouys et al. Delbouys et al. (2018). Distinct audio and lyrics branches receive Mel-spectrogram representations and word embeddings, respectively. The learned features for each modality are then fused and further processed by a small Dense Neural Network (DNN), finally outputting the AV values prediction.

We opted for a bimodal audio-lyrics approach considering that both modalities have relevant information for the different axes of Russell's Circumplex Model. Audio has been shown to better predict

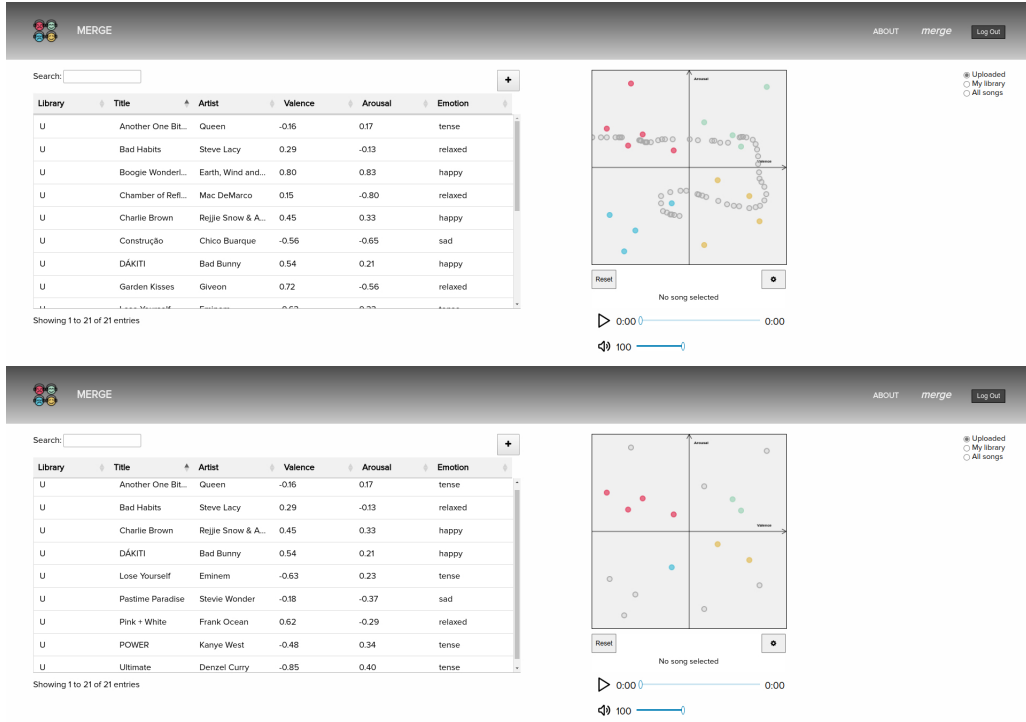


Figure 5: The user can be seen drawing a path to generate a playlist with the desired emotional trajectory at the top. The result of the path-based playlist generation is presented at the bottom.

arousal, while lyrical information is more relevant for valence prediction Louro et al. (2024b).

Starting in the audio branch, Mel-spectrogram representations of each sample are fed to the feature learning portion of the baseline architecture presented in Louro et al. Louro et al. (2024a). It is composed of four convolutional blocks, composed of a 2D Convolutional layer, followed by a Batch Normalization, Dropout, and Max Pooling layer, finishing with ReLU activation. As for the lyrics branch, the word embeddings of lyrics are also fed to four convolutional blocks, each comprising a 1D Convolutional layer, followed by Max Pooling and a ReLU activation layers. To balance the information from each modality, we significantly reduce the overwhelming amount of learned features from lyrics using a Dense layer before merging the learned features from both branches.

The classification portion of the model is composed of alternating Dropout and Dense layers, which reduce and further process the set of features respectively, finally outputting one of Russell’s Circumplex model’s four quadrants.

3.2 Pre-processing Steps

A set of pre-processing steps is necessary to obtain the data representations used for each branch of the architecture detailed above.

The librosa library McFee et al. (2015) is used to obtain the Mel-spectrogram representation for the audio branch. The audio samples, provided as mp3 files, are first converted to waveforms (.wav) and downsampled from 22.5 to 16kHz. This is done to reduce the complexity of the model, along with the computational cost for optimization. The downsampling has been shown to provide similar results to higher sampling rates, showing the robustness of DL approaches Pyrovolakis et al. (2022). The spectral representations are then generated using default parameters for the length of the Fast Fourier Transform window (2048) as well as the hop size (512).

As for word embeddings, the Sentence Transformer library from Hugging Face was used, specifically, the all-roberta-large-v1 pre-trained model. The embedder receives a context of up to 512 tokens and outputs a 1024 embedded vector. Given that the best results were provided by using the full context window, some of the lyrics had to be cut off at some point. After some simple tokenization steps, namely removing new line characters and converting all text to lowercase, the embeddings were obtained up to the already mentioned context size.

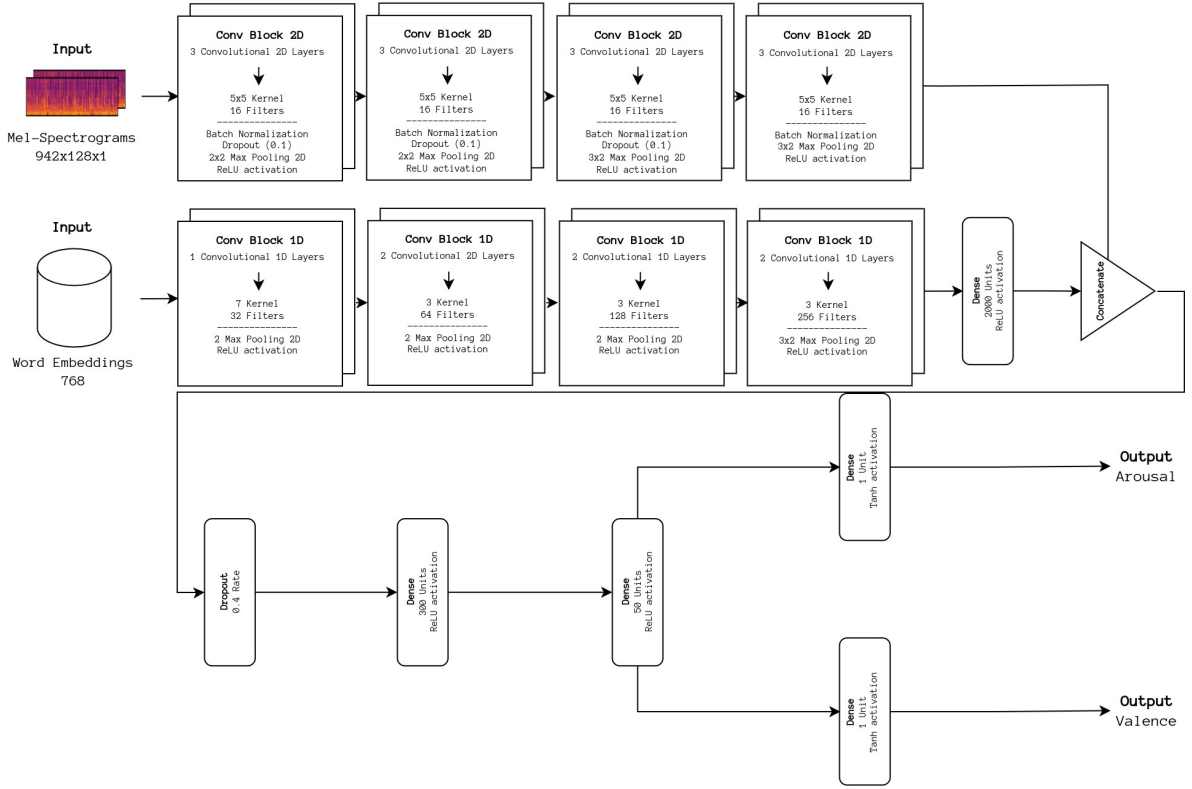


Figure 6: The multi-modal audio-lyrics regression model. Emotionally-relevant features are learned for both the audio representation in the Mel-spectrogram-receiving branch and the lyrics representation in the branch receiving the previously generated word embeddings. AV values are predicted after concatenating and processing the learned features from both branches.

3.3 Model Optimization

Model optimization was conducted using the Bayesian optimization implementation of the Keras Tuner library O’Malley et al. (2019). This method finds the best combination of hyperparameters in previously defined intervals for each, either maximizing or minimizing an objective function defined by the user.

Since our methodology is based on a regression task to predict arousal and valence for a given sample, the objective is defined as minimizing the sum of the mean squared error (MSE) for both. This ensures that none is prioritized, leveraging both audio’s better predictability in terms of arousal and the same for lyrics’ predictability of valence. The intervals for each considered hyperparameter, namely, batch size, optimizer, and corresponding learning rate, are presented in Table 1.

Table 1: Optimal Hyperparameters For Each Dataset

Best Hyperparameters		
Batch Size	Optimizer	Learning Rate
64	SGD	1e-2

The optimization process is run over ten trials, per the library’s default, starting at the lower end of each interval. For each trial, the model is trained to a maximum of 200 epochs, with an early stopping strategy defined to check for no improvements to the validation loss for 15 consecutive epochs. This considerably reduces the time needed to conduct the full optimization phase since less time is spent on underperforming sets of hyperparameters. We used a 70-15-15 train-validate-test (TVT) split as our validation strategy, as defined in Louro et al. (2024b). The resulting models for each trial are backed up for later usage, including the evaluation phase, which is discussed next.

3.4 Data and Evaluation

The MERGE Bimodal Complete dataset was used for validating our approach. Proposed in Louro et al. (2024b), it comprises a set of 2216 bimodal samples (audio clips and corresponding lyrics). For each sample, the dataset provides a 30-second audio excerpt of the most representative part of the song, links to the full lyrics, labels corresponding to each of the quadrants in Russell’s Circumplex model, and AV values, used to obtain the previously mentioned labels, cal-

Table 2: TVT 70-15-15 Results For MERGE Audio Complete

F1-score	Precision	Recall	R2 (A/V)	RMSE (A/V)
73.20%	74.53%	73.49%	0.454 0.506	0.133 0.339

culated based on the extracted emotion-related tags available in AllMusic ².

The above-mentioned AV values are obtained through the following process. First, the available tags for each song in the dataset are obtained from the All Music platform. Using Warriner’s Adjective Dictionary Warriner et al. (2013), the existing tags are translated to arousal and valence values. Finally, The values are then averaged across all tags corresponding to a specific song, obtaining its final mapping on Russell’s Circumplex model.

For the TVT strategy, both the training and validation sets are used in the optimization function. The set of optimal hyperparameters is found using the latter. After training the model for each dataset, the following metrics are computed between the actual and predicted AV values in the test set for each class as well as for the overall performance: F1-score, Precision, Recall, R^2 (squared Pearson’s correlation), and Root Mean Squared Error (RMSE).

Before computing these metrics, the predicted and real AV values were mapped to Russell’s Circumplex model to obtain classes for calculating Precision, Recall, and F1-score.

4 EXPERIMENTAL RESULTS AND DISCUSSION

Tables 2 and 3 show the overall results for the discussed methodology. The arousal and valence standalone results for the R^2 and RMSE metrics are presented in consecutive lines in the order displayed in the tables.

The obtained results for both datasets are lower than those obtained in previous studies focused on static MER as a categorical problem Louro et al. (2024a). The best result attained is a 73.20% F1-score, which is around 6% lower than the results obtained for the same dataset and evaluation strategy in the mentioned article. The lower results are mostly due to the semi-automatic approach to obtain AV values (see Section 3.4, considering that the tags available on All Music are user-generated and its curation is unknown.

²<https://www.allmusic.com/>

Table 3: TVT 70-15-15 Results Confusion Matrix For MERGE Audio Complete

		Predicted			
		Q1	Q2	Q3	Q4
Actual	Q1	61.3%	10.4%	6.6%	21.7%
	Q2	9.8%	82.4%	5.9%	2.0%
	Q3	1.4%	4.3%	78.3%	15.9%
	Q4	7.3%	0.0%	18.2%	74.5%

As shown in Table 2, the R^2 metric for valence outperformed the one for arousal, although having a larger RMSE. This indicates that the relative valence throughout songs is reasonably captured, despite the larger RMSE error.

Although the attained results show room for improvement, they are a good starting point for the user. Given the subjective nature of each user’s emotional perception, we believe that the personalization feature included in the MERGE app is a valuable mechanism for handling subjectivity in MER.

In terms of the results for separate quadrants (Table 3), we can see that some Q1 songs are confused with Q4 songs (21.77% Q1 songs are incorrectly classified as Q4). Moreover, there is also some confusion between Q3 and Q4 (15.9% of Q3 songs are predicted as Q4 and 18.2% of Q4 songs are classified as Q3). This is a known difficulty in MER, as discussed in Panda et al. (2020) that needs further research.

5 CONCLUSION AND FUTURE WORK

We presented the prototype for the MERGE application. Currently, the initial version has implemented music playback features, the ability to add and filter songs to a shared database, list and plane views, the latter based on Russell’s Circumplex model, and user management functionalities. Moreover, a bimodal audio-lyrics model is incorporated into the backend of the prototype to allow for AV value prediction of user-uploaded songs. Path-based playlist generation has also been implemented, enabling users to craft a playlist that follows a specific emotional trajectory they have selected.

Still, many more functionalities are planned for the application in future iterations. The highlighted

functionalities include user-generated tags for a more customized filtering experience that would be available to other users; automatic lyrics for the full song scraped from an available API, e.g., Genius; and Music Emotion Variation Detection (MEVD) prediction support, including visualization with the same color code used in the plot. A standalone desktop application is also planned without the cross-user features. In addition to implementing these upcoming features, We plan to conduct in-depth user experience studies to gain a more comprehensive understanding of the system's efficacy and user satisfaction.

Validation experiments on two recently proposed datasets are provided alongside a thorough system description, relaying insights into the obtained results. These are still below the categorical approach presented in Louro et al. (2024b) due to the already discussed semi-automatic AV mapping approach in Section 3.4. Despite this, the predictions are a good starting point to be further adjusted to the user's perception.

Regarding the actual model, neither feature learning portion may be ideal for the problem at hand since they were originally developed for a categorical problem. Developing more suitable architectures should thus be considered future work. Furthermore, the data representations, especially the word embeddings, may also be further improved, considering that the pre-trained model used is limited to a context window of 512 tokens.

To conclude, we believe the proposed app might be useful for music listeners. Although there is room for improvement (as the attained classification results show), the personalization mechanism is a useful feature for handling prediction errors and subjectivity. Finally, the personalization feature and the multi-user environment have the potential to acquire quality user annotations, leading to a future larger and more robust MER dataset.

ACKNOWLEDGEMENTS

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 (PID-DAC) 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.

We thank all reviewers for their valuable suggestions, which help to improve the article.

REFERENCES

- Cardoso, L., Panda, R., and Paiva, R. P. (2011). Moodetector: A prototype software tool for mood-based playlist generation. In *Simpósio de Informática - INForum 2011*, Coimbra, Portugal.
- Delbouys, R., Hennequin, R., Piccoli, F., Royo-Letelier, J., and Moussallam, M. (2018). Music Mood Detection Based On Audio And Lyrics With Deep Neural Net. In *Proceedings of the 19th International Society for Music Information Retrieval Conference*, pages 370–375, Paris, France.
- Hu, X., Downie, J. S., Laurier, C., Bay, M., and Ehmann, A. F. (2008). The 2007 Mirex Audio Mood Classification Task: Lessons Learned. In *Proceedings of the 9th International Society for Music Information Retrieval Conference*, pages 462–467, Drexel University, Philadelphia, Pennsylvania, USA.
- Louro, P. L., Redinho, H., Malheiro, R., Paiva, R. P., and Panda, R. (2024a). A Comparison Study of Deep Learning Methodologies for Music Emotion Recognition. *Sensors*, 24(7):2201.
- Louro, P. L., Redinho, H., Santos, R., Malheiro, R., Panda, R., and Paiva, R. P. (2024b). MERGE – A Bimodal Dataset for Static Music Emotion Recognition.
- McFee, B., Raffel, C., Liang, D., Ellis, D., McVicar, M., Battenberg, E., and Nieto, O. (2015). Librosa: Audio and Music Signal Analysis in Python. In *Python in Science Conference*, pages 18–24, Austin, Texas.
- O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., et al. (2019). Keras Tuner. <https://github.com/keras-team/keras-tuner>.
- Panda, R., Malheiro, R., and Paiva, R. P. (2020). Novel Audio Features for Music Emotion Recognition. *IEEE Transactions on Affective Computing*, 11(4):614–626.
- Pyrovolakis, K., Tzouveli, P., and Stamou, G. (2022). Multi-Modal Song Mood Detection with Deep Learning. *Sensors*, 22(3):1065.
- Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178.
- Warriner, A. B., Kuperman, V., and Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods*, 45(4):1191–1207.