DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Final Project

DeepMusicGenreClassifier

21 de agosto de 2025 — Signal, Audio and Speech Processing

Integrante	LU	Correo electrónico
Aaron Bernal Huanca	815/22	aaronbernal28@gmail.com

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - (Pabellón I/Planta Baja)
Intendente Güiraldes 2610 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (++54 +11) 4576-3300
http://www.exactas.uba.ar

Resumen

This report addresses music genre classification using the GTZAN dataset (10 genres, 100 songs of 30 seconds each). The objective is to analyze and classify music using various neural network architectures. Methods are implemented that include sequential processing with RNNs, spectrogram analysis (Mel, MFCC, CQT) with CNNs (1D), and the use of pre-trained neural networks for relevant feature extraction (transfer learning) such as Wav2vec2 and EnCodecMAE. Cross-entropy and accuracy are employed as evaluation criteria to determine which of these architectures offers better performance in the classification task.
Dataset: https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification/data
Repository: https://github.com/aaronbernal28/DeepMusicGenreClassifier

1. Introduction

We have a dataset of 100 songs, each with a duration of 30 seconds, evenly distributed across 10 different musical genres. From these songs, different representations (embeddings or spectrograms) are extracted, which we will henceforth call features for simplicity.

The data was stratified and divided into three subsets: 10 % for final evaluation, and the remaining 90 % was separated into 80 % for training and 20 % for validation. This division was performed independently of the features.

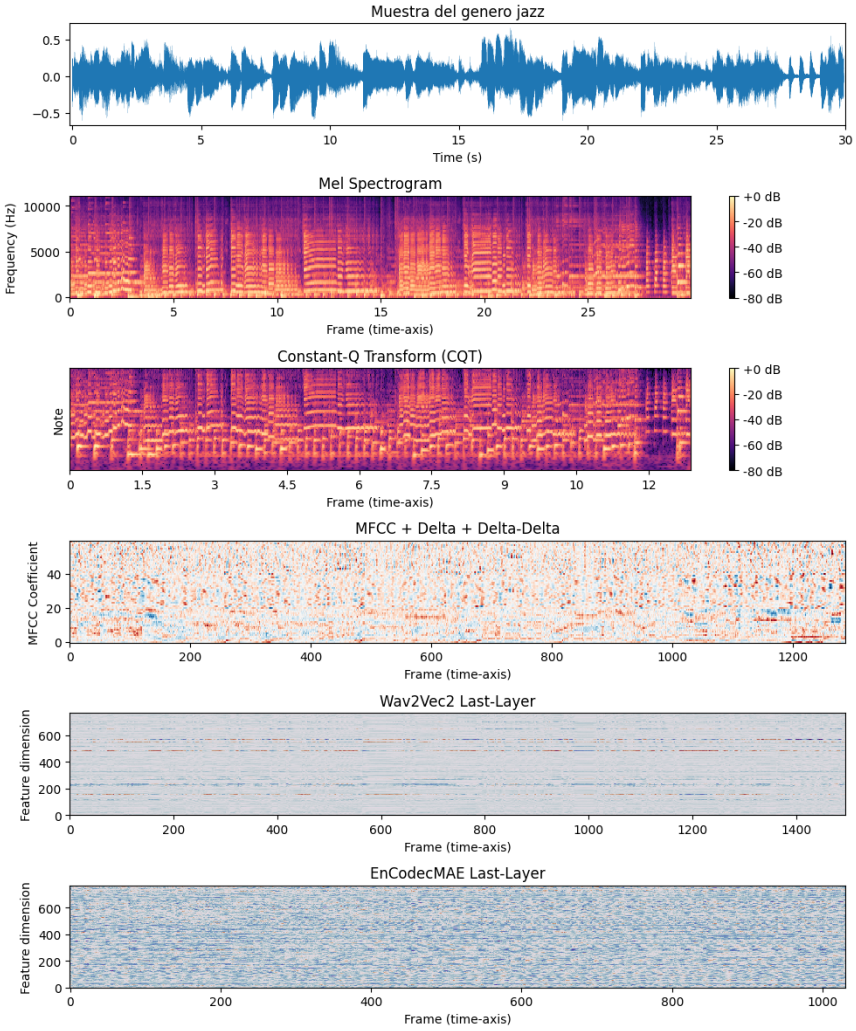


Figura 1: Different representations of a jazz genre sample

The CQT representation is specifically designed to capture musical structures such as notes, chords and their harmonics ([2], [3]). It is oriented towards musical audio analysis; see Figure 1 for a sample of the jazz genre.

Additional information that may be of interest:

feature	dimension
Mel	80
MFCC	60
CQT	84
Wav2vec2	768
EnCodecMAE	768

Tabla 1: Dimension of each feature per frame or time. 10 ms hop length and 25 ms window where applicable, the rest of parameters by default

It should be clarified that the feature extraction of EnCodecMAE was performed with the first 15 seconds (not the original 30s) due to memory constraints.

Finally, although the architecture remained identical for all representations, differences in size and memory consumption forced adjustments to some hyperparameters. For example, if the CQT matrix $(84, 1290)$ allowed a batch size of 64, the Wav2Vec2 embedding $(1496, 768)$ would require batch size 28, which implied respective learning rates. Similarly, the number of epochs was set dynamically, stopping training when the validation loss stabilized or began to deviate significantly from that of training.

2. Development

2.1. Random Forest

This model, for each feature, takes the mean with respect to the sequence or time, that is, we go from $(T, feature\_dim)$ to $(, feature\_dim)$. Subsequently, a classic Random Forest with 100 ensembles is trained.

Feature	**Accuracy**
Mel	0.27
MFCC	0.14
CQT	0.32
Wav2vec2	0.66
EnCodecMAE	0.79

Tabla 2: Performance of each feature taking the mean

EnCodecMAE gives good results comparable with the following models and serves as our reference (Table 2). Considering that we have fixed-dimension vectors $feature\_dim_{EnCodecMAE} = 768$ and good performance, we project to two dimensions with the help of UMAP to understand how the classes are distributed.

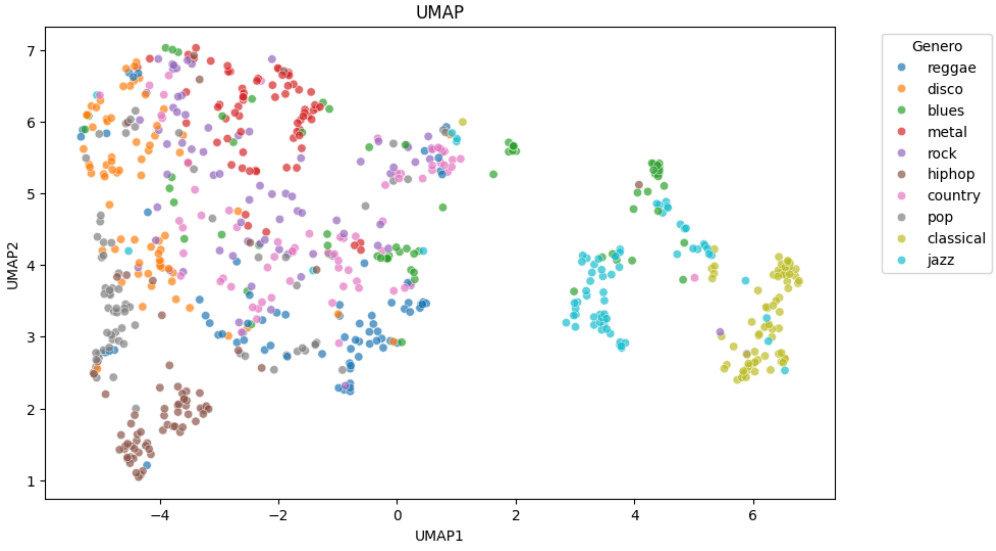


Figura 2: UMAP projection to 2D

There is a clear non-perfect separation for some classes as can be observed in 2, for example between *hiphop*, *jazz* and *classical*. Additionally, *classical* is more distant from the rest. This corresponds to the purpose of the representations, in particular, that they are discriminative and capture semantic relationships.

2.2. Multilayer perceptron (MLP)

We present a naive approach where we extract the first 2049 frequencies from the Fourier series of each sample (Table 3).

Layer Type	Input Size	Output Size
Input Layer	2049	–
Hidden Block 1	2049	1024
... 2	1024	512
... 3	512	256
... 4	256	128
... 5	128	64
Output Layer	64	10

Tabla 3: Network architecture with Gelu activation, dropout 0.5 and batchnorm 1D. Number of parameters: $2,801,098$

As can be seen in Figure 3, the model overfits quickly and the test data loss stagnates. After 100 epochs it reaches an accuracy of 0.49.

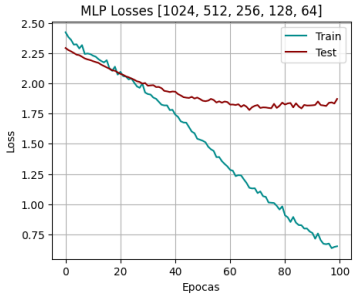


Figura 3: Learning curve of the network

2.3. Recurrent networks

This section describes the two architectures that obtained the best performance. In all cases, the Mel, CQT and MFCC representations are considered as sequences of feature vectors with dimensions specified in Table 1.

The first model uses LSTM and the second a variation of the same: GRU. Both have as output a hidden state of dimension $(2*hidden\_size, T)$. Then, a simple attention mechanism is applied to finally obtain an output of the dimension of the number of classes, in our case, 10. See Figure 4.

2.3.1. LSTM

Below is a detailed schematic of the LSTM-based architecture:

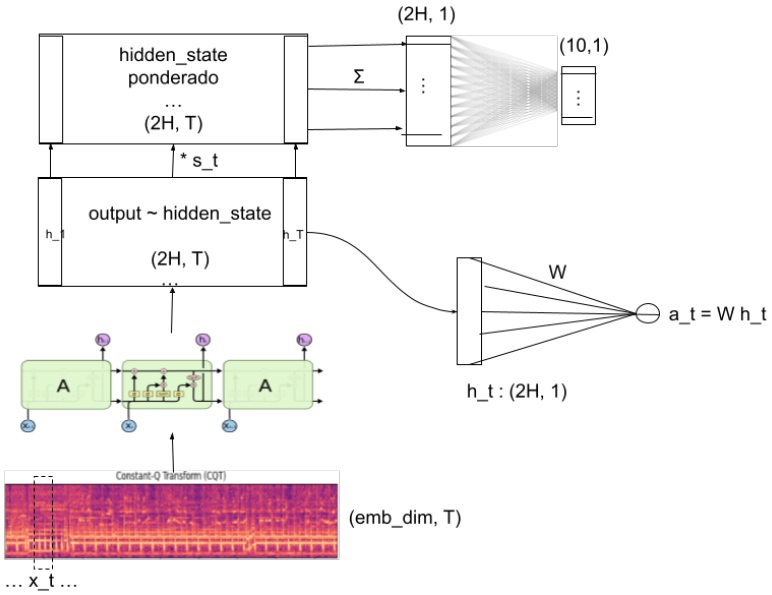


Figura 4: Model scheme with bidirectional LSTM where $(s_1, \ldots, s_T) = softmax(a_1, \ldots, a_T)$ and $H = 16$

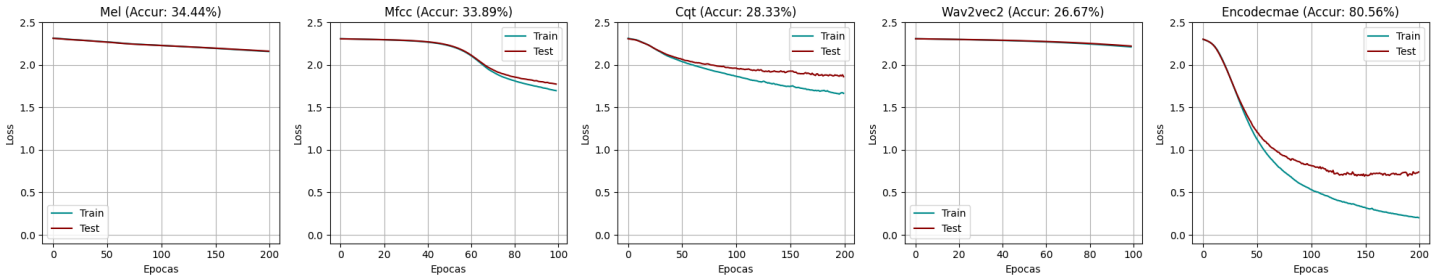
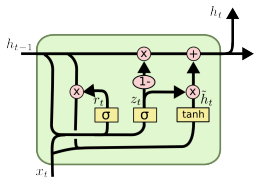


Figura 5: LSTM learning curves

From the loss and accuracy curves obtained for each type of representation 5, we can state that EnCodecMAE achieves good performance and stabilizes. The rest of the features have poor performance with very slow learning.

2.3.2. GRU

The architecture is the same as in scheme 4, however, the LSTM recurrent unit is replaced by GRU. Additionally, the *hidden_size* becomes 32.



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$
$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Figura 6: GRU scheme

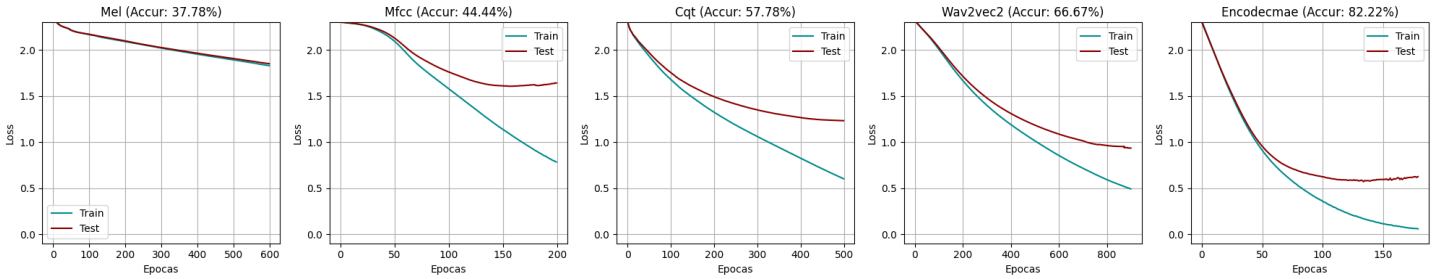


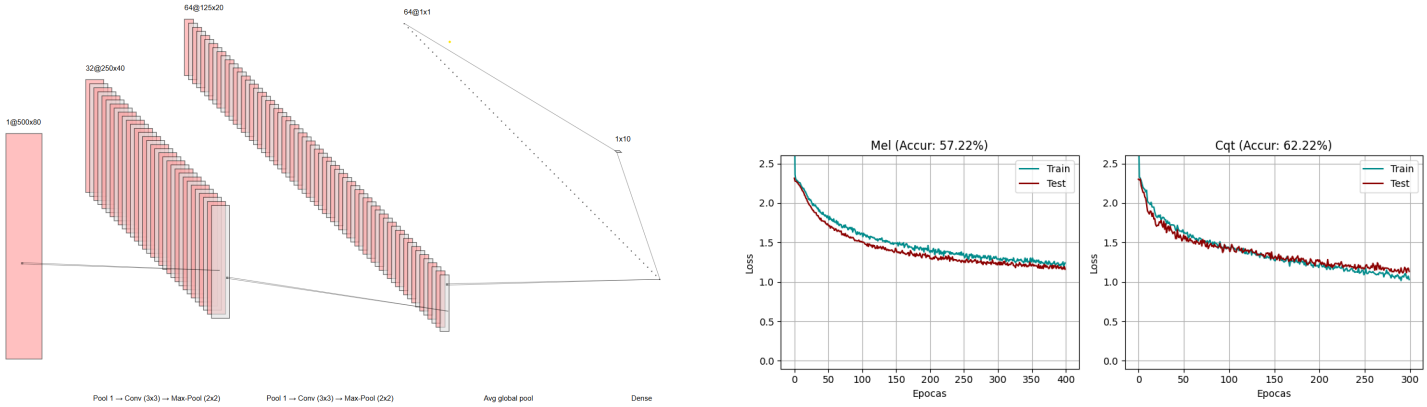
Figura 7: GRU learning curves

As can be observed in Figure 7, performance improves consistently as we move from traditional spectrograms towards self-supervised embeddings. With Mel we obtain 37 % accuracy, MFCC scales to 44 % and CQT reaches 58 %. When incorporating Wav2Vec2 representations, precision rises to 67 %, while EnCodecMAE excels with more than 82 %. This gap between spectrograms and representations may be due to their ability to capture high-level musical features. It should be clarified that regularization methods such as dropout were experimented with but did not produce additional improvements.

2.3.3. 2D Convolutional networks

MEL and CQT are the only representations that show various patterns worth exploring with 2D convolutional networks. For this, Figure 8a, we propose a simple architecture with:

- Convolutions with 3×3 kernel, followed by 2×2 max-pooling.
- Filter progression from 32 to 64 channels.
- Global average pooling to reduce spatial dimension to 64.
- Final fully connected layer from $64 \rightarrow 10$ (number of classes).



(a) Architecture scheme (illustrative dimensions)

(b) Model learning curves

Figura 8: (a) CNN2D architecture with filters from 32 to 64 and global pooling. (b) Evolution of loss in training and validation.

The results of 8b are reasonable, however, there is a loss gap $\mathcal{L} = 1$ that cannot be overcome or requires many epochs.

2.3.4. 1D Convolutional networks

The architecture is detailed below:

- Conv1d from feature_len → 128 (kernel=3, padding=1), LeakyReLU and MaxPool1d (k=3, s=3).
- Conv1d from 128→256 (kernel=3, padding=1), LeakyReLU and MaxPool1d (k=3, s=3).
- Conv1d from 256→512 (kernel=3, padding=1), LeakyReLU and MaxPool1d (k=3, s=3).
- AdaptiveAvgPool1d to size 1, reshape to $(B, 512)$ and Dropout($p = 0{,}5$).
- Linear from 512→128 + LeakyReLU, followed by Linear from 128→10.

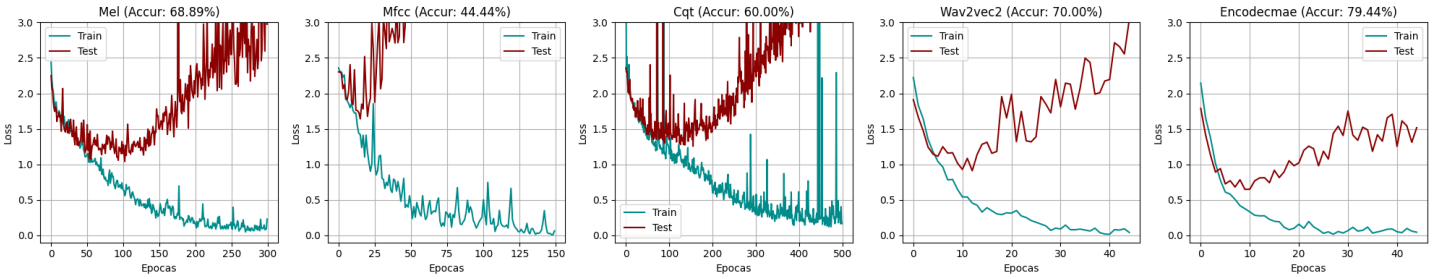


Figura 9: 1D CNN learning curves

In general, performance comparable to or better than previous ones is obtained. However, MEL, MFCC and CQT show great instability while all overfit rapidly.

2.4. Encoder-Transformer

The idea behind this approach is to provide context to representations that are not embeddings and slightly adjust those that are. For this, we use $N = 4$ Transformer encoder layers (see Figure 10) and $dim\_feedforward = 16$.

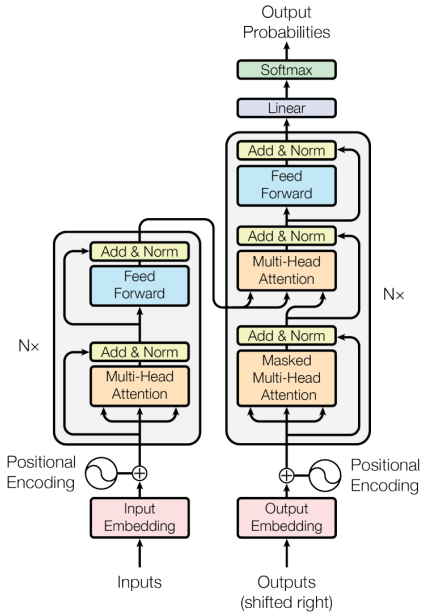


Figura 10: Transformer scheme

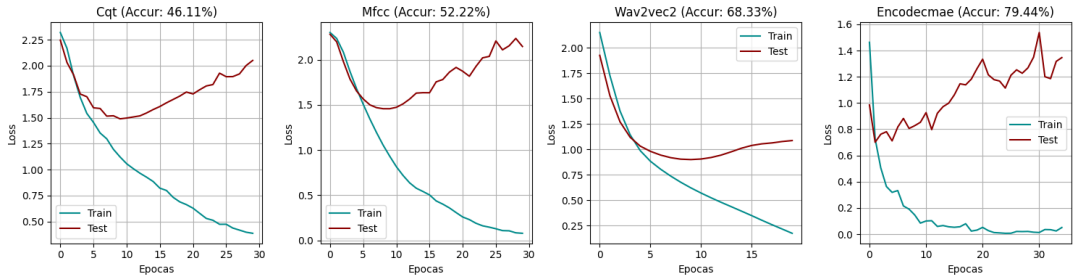


Figura 11: Transformer learning curves

In the curves of Figure 11 it is observed that the Transformer manages to improve stability compared to CNN1D in CQT and MFCC, with final precisions of 46 % and 52 % respectively, moderating overfitting. With Wav2Vec2 it reaches 68 % and with EnCodecMAE it maintains a solid 79 %. Note that MEL was not included due to hardware limitations.

Observation: $dim\_feedforward = 16$ creates a bottleneck, experiments were conducted with $dim\_feedforward = 2 \times emb\_dim$ as recommended in the literature, however the results were not superior.

3. Evaluation

The best model is obtained using the GRU recurrent unit 2.3.2. Below it is tested with evaluation data.

Genre	**Precision**	**Recall**	**F1-score**	**Support**
blues	0.90	0.90	0.90	10
classical	1.00	1.00	1.00	10
country	0.73	0.80	0.76	10
disco	0.90	0.90	0.90	10
hiphop	0.75	0.90	0.82	10
jazz	1.00	1.00	1.00	10
metal	0.91	1.00	0.95	10
pop	0.88	0.70	0.78	10
reggae	0.82	0.90	0.86	10
rock	0.57	0.40	0.47	10
macro avg	0.85	0.85	0.84	100
accuracy	0.85	-	-	100

Tabla 4: Best model performance

The best version (Table 4), presents an overall performance of 85 % accuracy, with an average F1-score of 0.84.

The classical and jazz classes achieve perfect scores (1,00, 1,00, 1,00), indicating that these categories are perfectly discriminated, which clearly corresponds to what was observed in the UMAP 2 from the introduction. On the other hand, genres such as metal, blues and disco also show very high metrics ($F1 \geq 0{,}90$), which is new because in the UMAP these classes were very mixed.

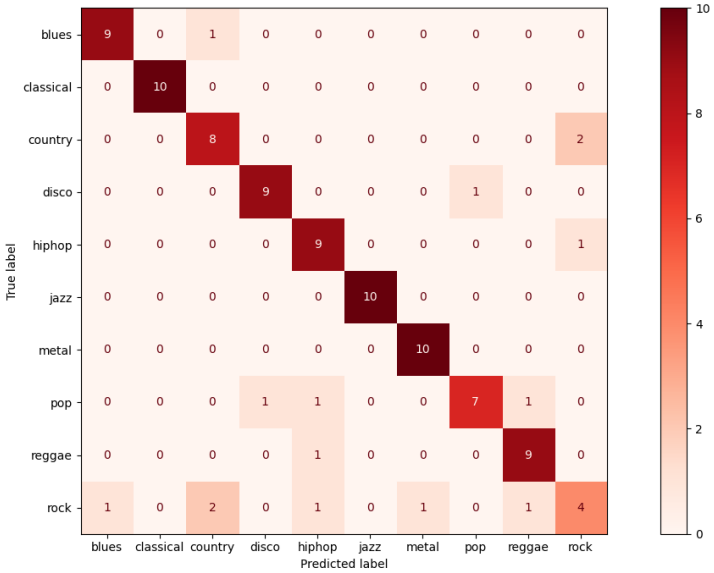


Figura 12: Confusion matrix of the best model

On the other hand, rock is the most problematic, with an F1-score of only 0.47 and a recall of 0.40, suggesting frequent confusion with other styles. More specifically, from the confusion matrix 12, only 4 of the 10 rock samples are correctly labeled; the rest is distributed among blues (1), country (2), disco (1) and pop (1), which explains its low recall (0.40) and F1-score (0.47).

4. Conclusion

The results show that pre-trained embeddings, especially those from EnCodecMAE, provide a richness of characteristics that translates into notably superior performance compared to traditional representations.

The best performance was obtained with the bidirectional GRU on EnCodecMAE embeddings, reaching 85 % accuracy and a macro F1-score of 0.84, with perfect classification of genres such as *classical* and *jazz*. However, genres such as *rock* present significant errors, suggesting the need for new specific strategies (e.g. increasing the amount of data, class weighting or more complex architectures) to improve discrimination in more complex categories. Of course, this would imply an upgrade in current hardware.

Referencias

[1] Leonardo Pepino, Pablo Riera, Luciana Ferrer. *EnCodecMAE: Leveraging neural codecs for universal audio representation learning* 2023

[2] Schoerkhuber, Christian, and Anssi Klapuri. *Constant-Q transform toolbox for music processing.* 7th Sound and Music Computing Conference, Barcelona, Spain. 2010.

[3] Lena Sophie Brüder. *Music classification using Constant-Q based Features* Master's Thesis, Ruhr-University Bochum, in cooperation with Research In Motion Deutschland GmbH. 2013