MLDL, A.Y. 2020/21

Neural Music Genre Classification

Politecnico di Torino

OUR TEAM

Hafez Ghaemi
s289963

Francesco Capobianco
s281307

Alessia Leclercq
s291871

OVERVIEW

1 Introduction

2 Methodology

3 Results

4 Discussion Conclusion

INTRODUCTION

Goal: music genre classification using deep learning architectures

TRADITIONALLY

- Feature vectors
- Traditional ML algorithms

NOWADAYS

- Spectrograms
- Deep Learning

SPECTROGRAM

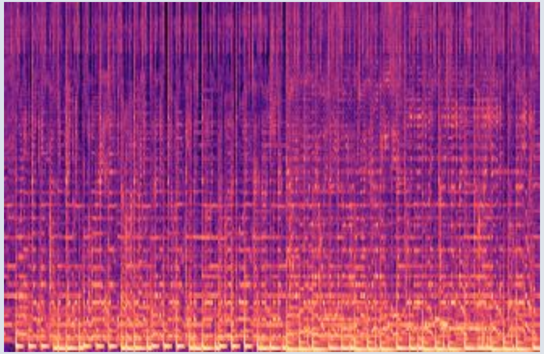
A spectrogram is a visual representation of the spectrum of frequencies of a signal as it varies over time.

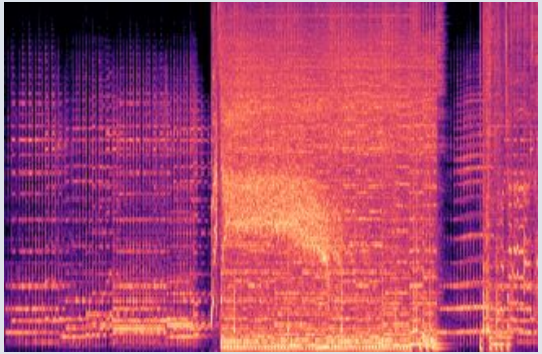
- invertible
- both temporal and frequency contents

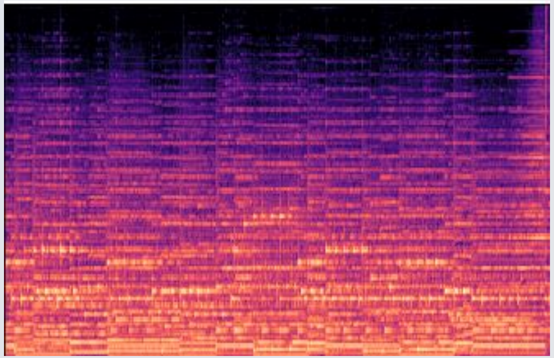
$$STFT\{x(t)\}(\tau, w) = X(\tau, w) = \int_{-\infty}^{+\infty} x(t)w(t-\tau)e^{-iwt}dt$$

$$m = 2595\log_{10}(1 + \frac{f}{700})$$

$$d = 10\log_{10}(\frac{m}{r})$$







CRNN

CRNN exploits:

- Convolutional Neural Network to perform feature extraction
- Recurrent Neural Network to keep the temporal overview over the features

As a consequence, both temporal and frequency related contents are managed simultaneously

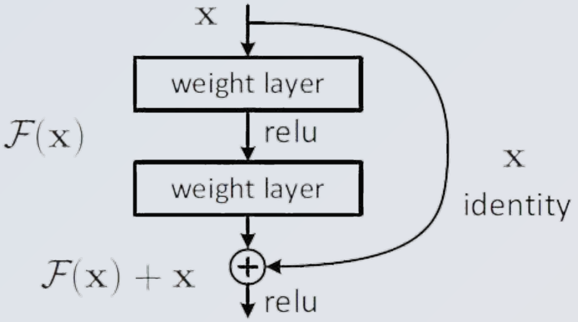
THE MUSIC GENRE CLASSIFICATION TASK IS TURNED INTO A COMPUTER VISION TASK

TRANSFER LEARNING

- Transfer learning helps in transferring the knowledge acquired on a specific domain to another and related problem.
- In this case we exploited the knowledge of the backbone ResNet-18 architecture trained on ImageNet to perform feature extraction on the spectrograms
- We will keep the recurrent layers at the bottom of the ResNet to keep the temporal overview on the extracted features

Pros of residual blocks:

- Deeper model and more features to be learned
- The skip connection helps in mitigating the vanishing gradient
- Avoids the deterioration of performance



K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

METHODOLOGY (OVERVIEW)

- Dataset Description
- Preprocessing
- Approaches
- Hyperparameters
- Evaluation

METHODOLOGY (DATASET)

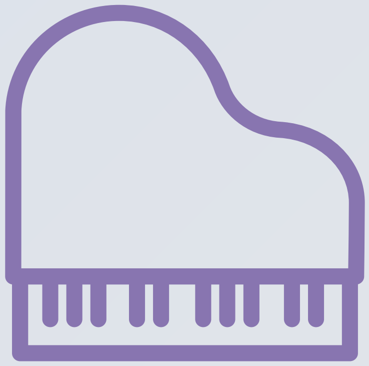
- First introduced by Tzanetakis *et. al.* [1]
- Can be accessed on Kaggle
- One hundred 30-sec tracks for each 10 genre, a total of 1000 tracks.
- Two CSV files along spectrograms

Blues, Classical, Country, Disco, Hiphop, Jazz, Metal, Pop, Reggae, Rock

[1] Tzanetakis, George, and Perry Cook. "Musical genre classification of audio signals." IEEE Transactions on speech and audio processing 10.5 (2002): 293-302.

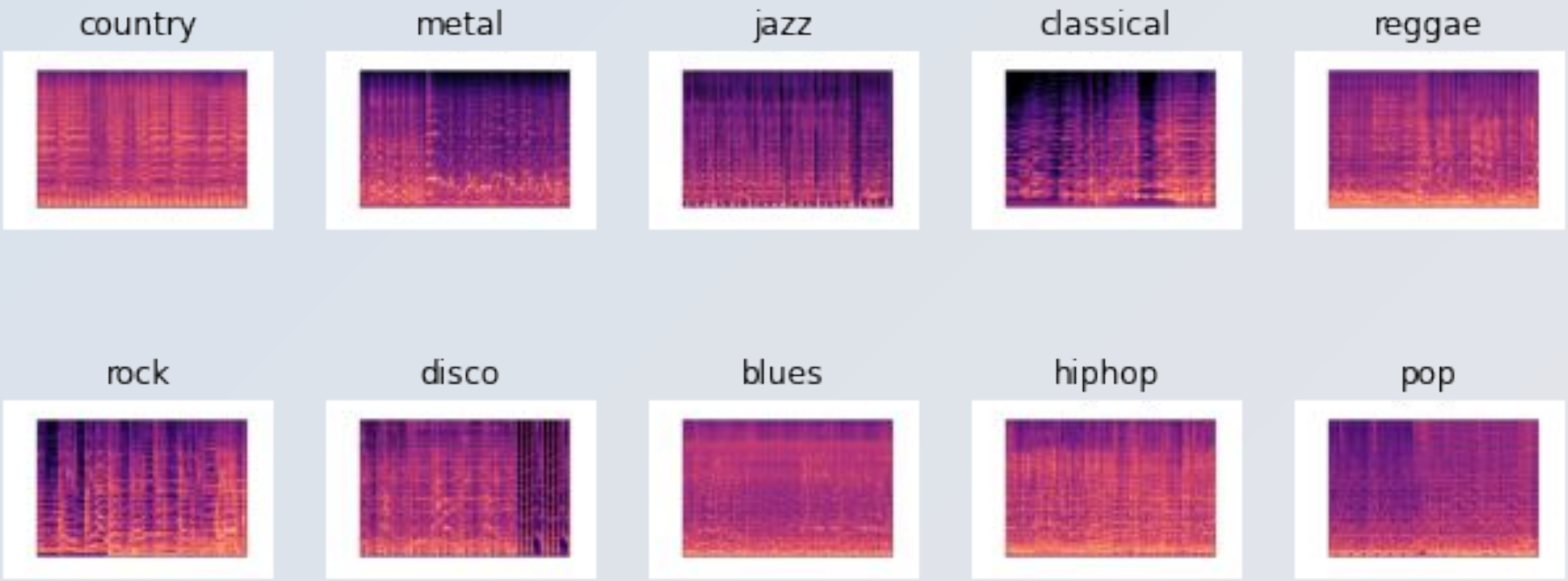
METHODOLOGY (PREPROCESSING)

Original audio files → Split into chunks → Generate mel spectrograms → Four balanced datasets with 1000, 3000, 10000, and 30000 labeled images



Sampling Rate	22050 Hz
Number of Mel Bins	192
Highest Frequency	8000 Hz
Hop Length	256

METHODOLOGY (PREPROCESSING)

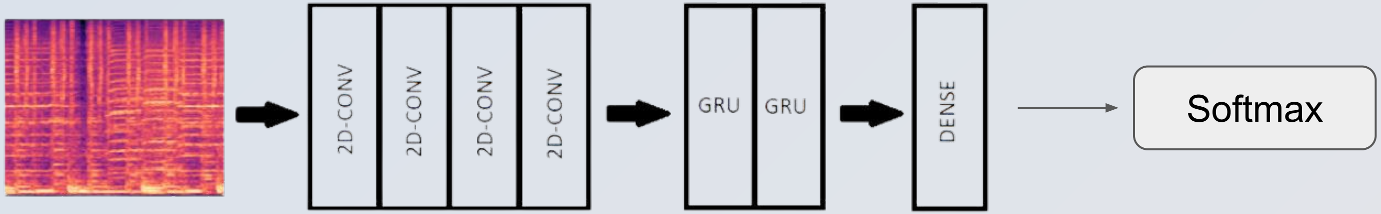


METHODOLOGY (APPROACHES)

- ML baselines (SVM, KNN, RF, LR)
- Base CRNN
- Large CRNN
- ResNet-18 CNN backbone with transfer learning

METHODOLOGY (THE BASE CRNN)

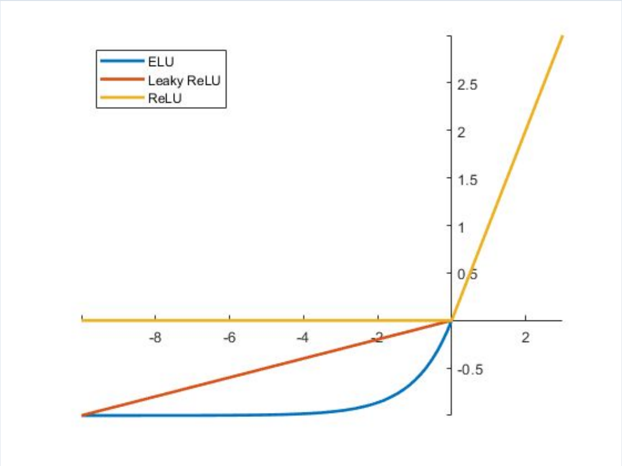
- Inspired by Nasrullah and Zhao [1]
- Originally for music artist classification
- Reimplemented in PyTorch



[1] Nasrullah, Zain, and Yue Zhao. "Music artist classification with convolutional recurrent neural networks." 2019 International Joint Conference on Neural Networks (IJCNN). IEEE, 2019.

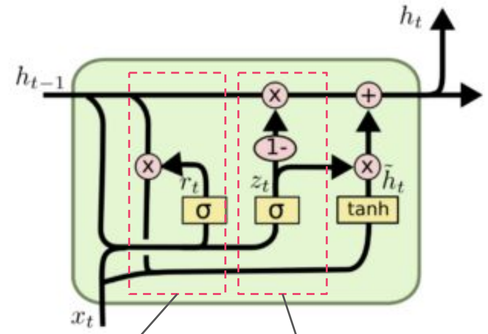
METHOD (THE LARGE CRNN)

Hyperparameter	The Base CRNN	The Large CRNN
Filters	[64, 128, 128, 128]	[64, 128, 256, 512, 512]
Kernel	3×3	3×3
Activation	ELU	ELU
Batch Normalization	Channel	Channel
Pooling	[(2,2), (4,2), (4,2), (4,2)]	[(2,2), (2,2), (2,2), (4,1), (4,1)]
Dropout	0.1	0.1



$$R(z) = \begin{cases} z; & z > 0 \\ \alpha.(e^z - 1); & z \leq 0 \end{cases}$$

METHODOLOGY (GRU and Dense Layers)



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

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

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

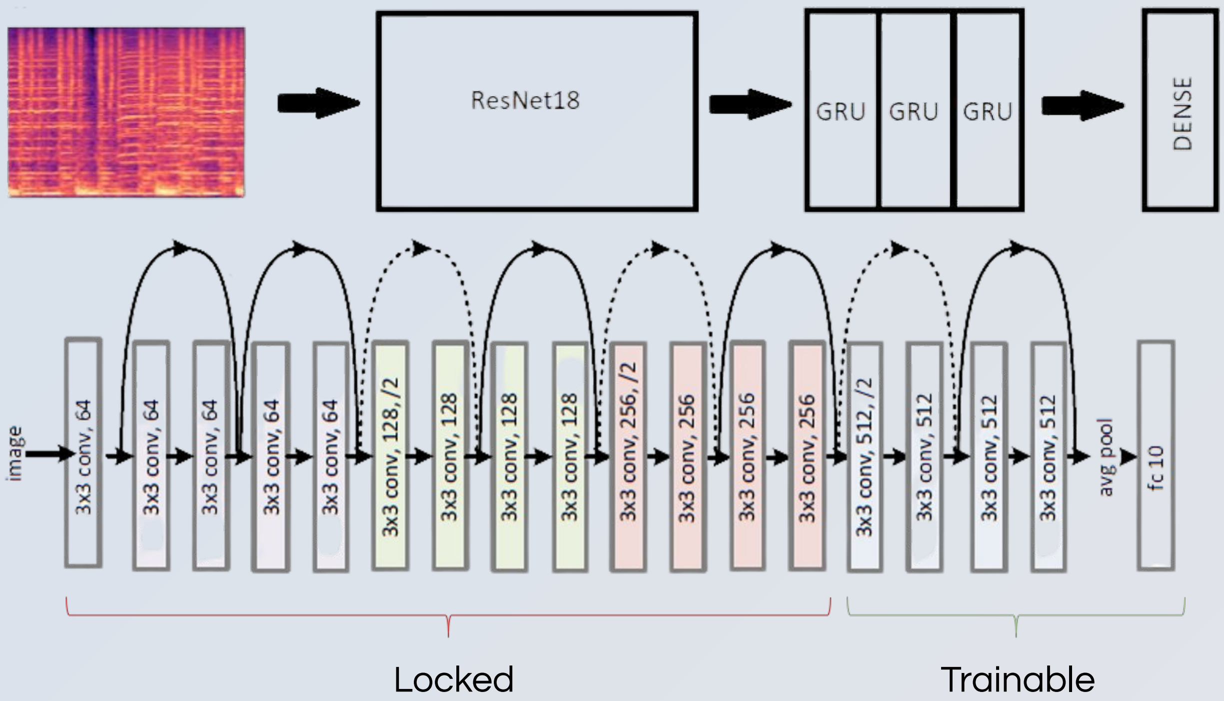
Reset Gate

Update Gate

GRU Units per Layer	32
GRU Dropout	0.3
Dense Layer Neurons	20
Dense Layer Activation	Softmax

$$s(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{No.\,classes} e^{x_j}}$$

METHODOLOGY (RESNET-18 BACKBONE)



Skip Connections

Transfer Learning

[1] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

METHODOLOGY (Evaluation)

- Traditional ML methods: 80/20 train/test split of the two tabular feature sets
- Deep-learning methods: 80/10/10 train/val/test split of the four image datasets (30/10/3/1 second splits)

$$Loss = -\sum_{i=1}^{No.\,classes} y_i . \log \hat{y}_i$$

Early stopping with patience of 10 epochs

Categorical crossentropy loss after softmax and ADAM optimization

RESULTS

Traditional Machine-Learning approaches results on GTZAN dataset

Model	Train F1 score 30 sec. (%)	Test F1 score 30 sec. (%)	Train F1 score 3 sec. (%)	Test F1 score 3 sec. (%)
SVM (default)	88.89	69.63	92.23	85.98
SVM (C = 10)	99.87	78.03	99.65	**91.61**
KNN (k = 1)	100	66.67	99.89	**91.47**
KNN (k = 5)	78.75	69.41	93.44	89.67
Random Forest	100	68.04	100	87.65
Logistic Reg.	100	67	100	72.89

RESULTS

Deep-Learning approaches results on GTZAN dataset

30 seconds chunks

Model	Train F1 score (%)	Validation F1 score (%)	Test F1 score (%)
Base	99.38	57	69
Extended	54.43	55.37	55.08
Transfer	99.88	71	**88**

10 seconds chunks

Model	Train F1 score (%)	Validation F1 score (%)	Test F1 score (%)
Base	99.71	77.52	76.72
Extended	99.8	78.83	80.33
Transfer	99.92	89.25	**89.18**

RESULTS

Deep-Learning approaches results on GTZAN dataset

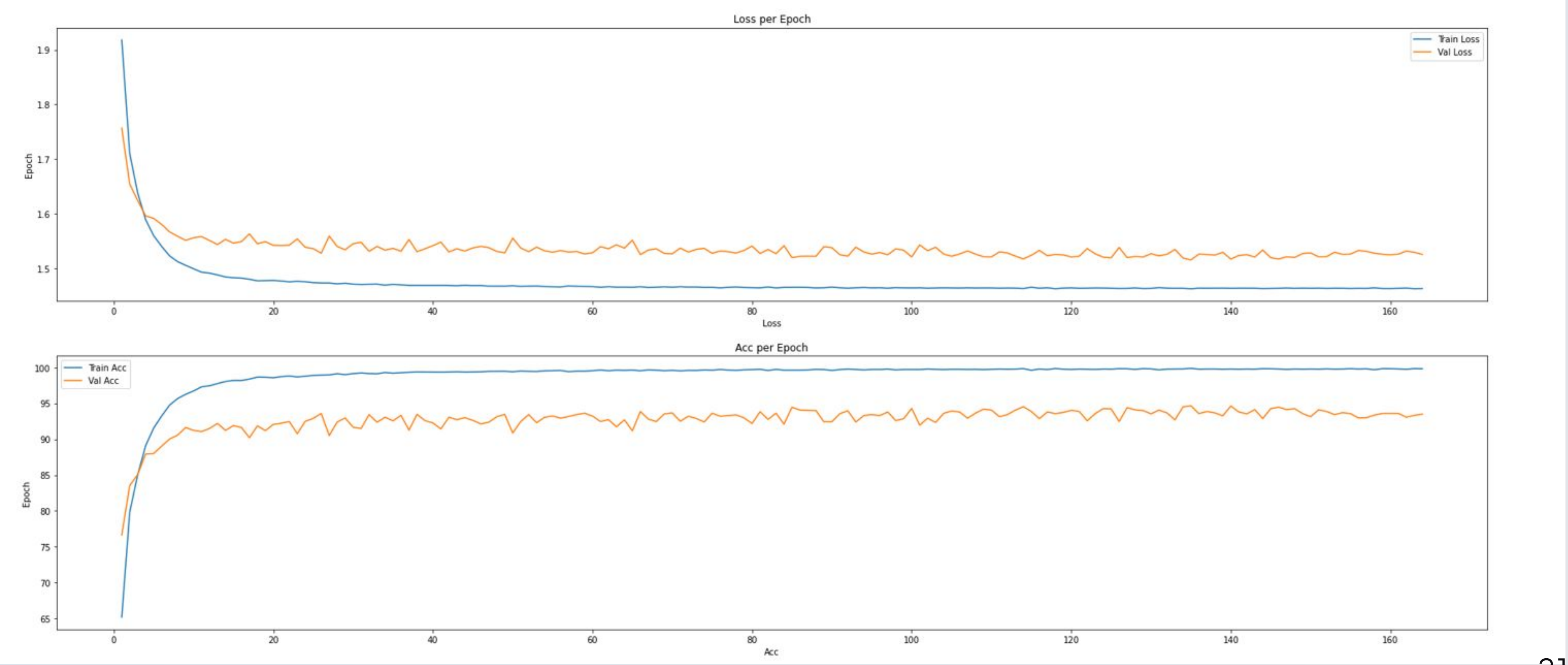
3 seconds chunks

Model	Train F1 score (%)	Validation F1 score (%)	Test F1 score (%)
Base	99.26	89.25	89.18
Extended	99.84	90.7	90.6
Transfer	99.92	90.93	**91.7**

1 second chunks

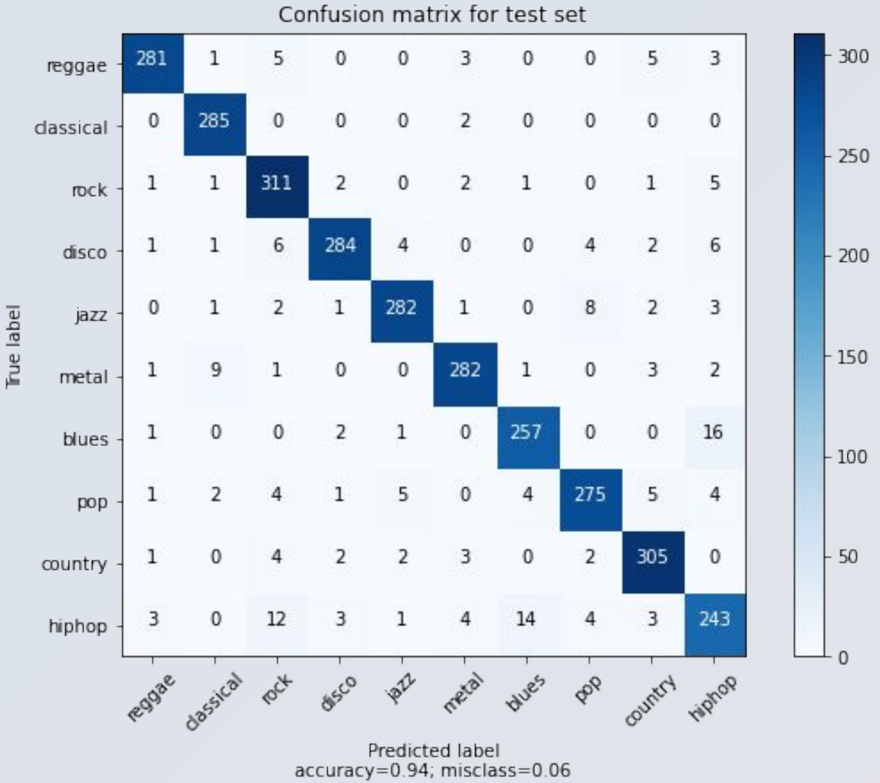
Model	Train F1 score (%)	Validation F1 score (%)	Test F1 score (%)
Base	99.34	90.93	**91.7**
Extended	99.76	90.17	89.74
Transfer	99.92	93.5	**93.5**

RESULTS



Loss per Epoch
Train Loss
Val Loss
Epoch
1.9
1.8
1.7
1.6
1.5
0
20
40
60
80
100
120
140
160
Loss
Acc per Epoch
Train Acc
Val Acc
Epoch
100
95
90
85
80
75
70
65
0
20
40
60
80
100
120
140
160
Acc

RESULTS



Confusion matrix for test set
True label
Predicted label
accuracy=0.94; misclass=0.06
reggae
classical
rock
disco
jazz
metal
blues
pop
country
hiphop
281 1 5 0 0 3 0 0 5 3
0 285 0 0 0 2 0 0 0 0
1 1 311 2 0 2 1 0 1 5
1 1 6 284 4 0 0 4 2 6
0 1 2 1 282 1 0 8 2 3
1 9 1 0 0 282 1 0 3 2
1 0 0 2 1 0 257 0 0 16
1 2 4 1 5 0 4 275 5 4
1 0 4 2 2 3 0 2 305 0
3 0 12 3 1 4 14 4 3 243
300
250
200
150
100
50
0

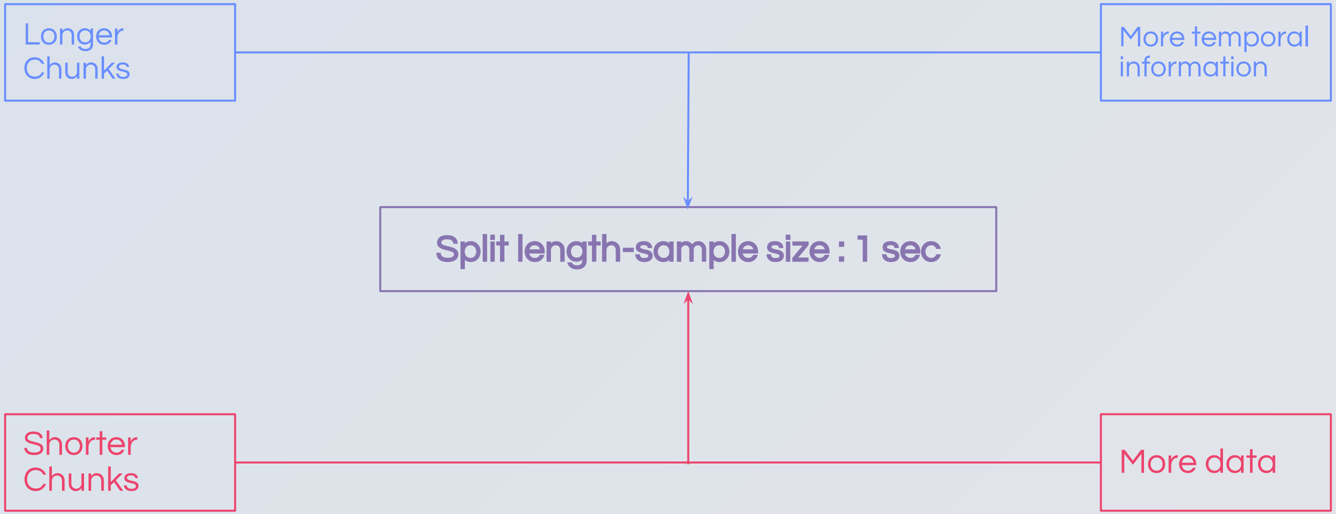
DISCUSSION

Convergence analysis

- Difference of accuracies between train and validation set (model variance) not always mean **overfitting**!
- **Early stopping** helps us to avoid overfitting!

DISCUSSION

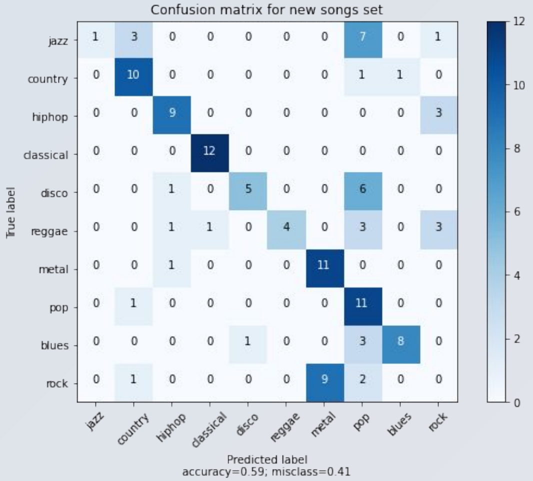
Possible trade-off between the number of samples and the length of the splits



DISCUSSION

Results on external song

- Ten songs, one for each genre, not included in GTZAN
- ResNet-18 based model
- Why much lower performance? **Lack of data and poor variety of songs in GTZAN!**



DISCUSSION

Further works and limitations

- GTZAN: not the richest dataset!
- Model complexity w.r.t. the computational power
- Transfer Learning: a way to solve this problem?

CONCLUSION

What have we learn from this experience?

- Power of mel spectrograms
- Combination of CNN and RNN
- Our extension : ResNet-18 as backbone

THANKS FOR YOUR ATTENTION