Artist Classification by Spectrogram Learning

AKA Improving upon Shazam

Julien Guinot
Applied Machine learning, Aug. 2021
a1831082

"Identify songs playing around you in seconds" - Shazam

→ **Audio segmenting** → **Frequency fingerprint** → **Hard matching**

Shazam's algorithm

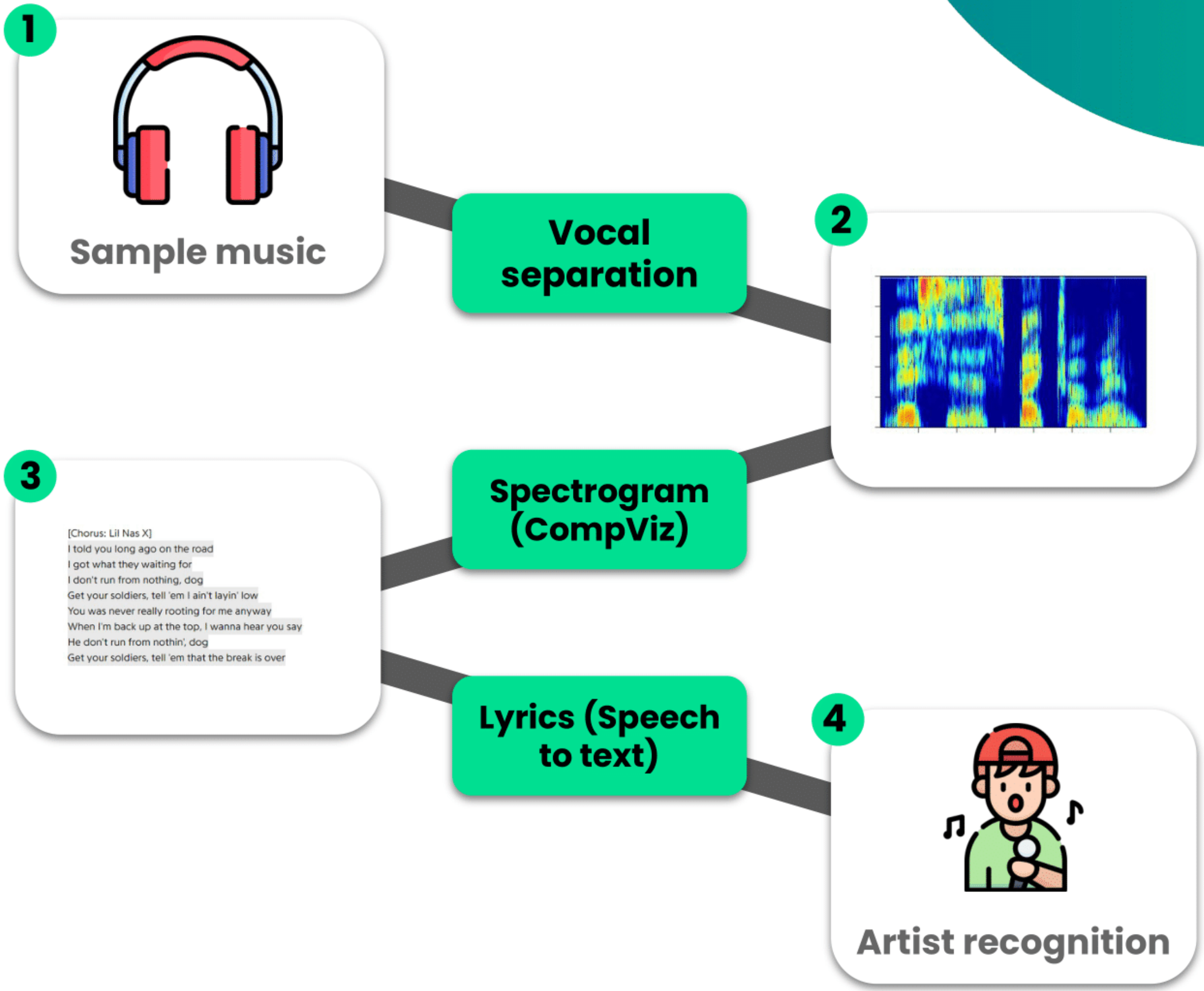
A real world problem

In practice :

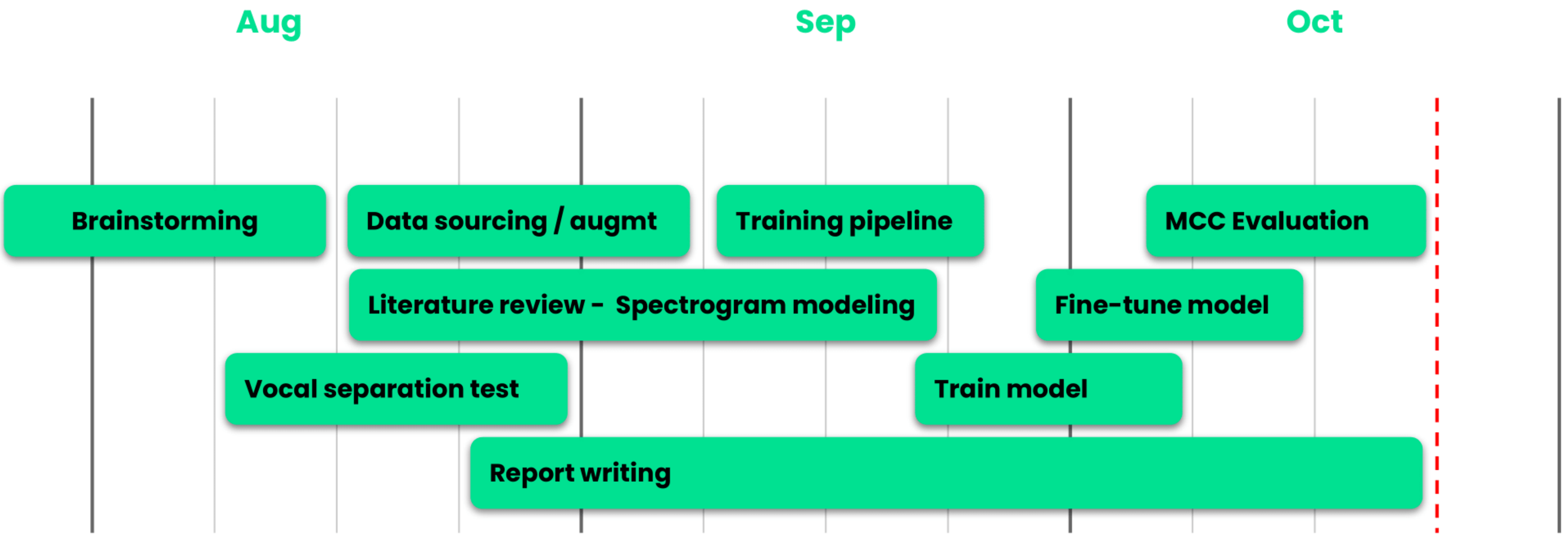
- Does not work on live performances
- Sped up, slowed down, filters

How to mimic an expert friend one would ask?

A challenging technical background



The (provisional) game plan



Artist Classification by Spectrogram Learning

Data and Project Goals

Julien Guinot
Applied Machine learning, Aug. 2021
a1831082

Data Needs Analysis and Sourcing

Objective: Classify artists by vocal characteristics

Data needs : Recordings of artist vocals

Large Dataset

Period Diversity

Vocalist Diversity

Genre Diversity

Objective: Classify artists by vocal characteristics

1

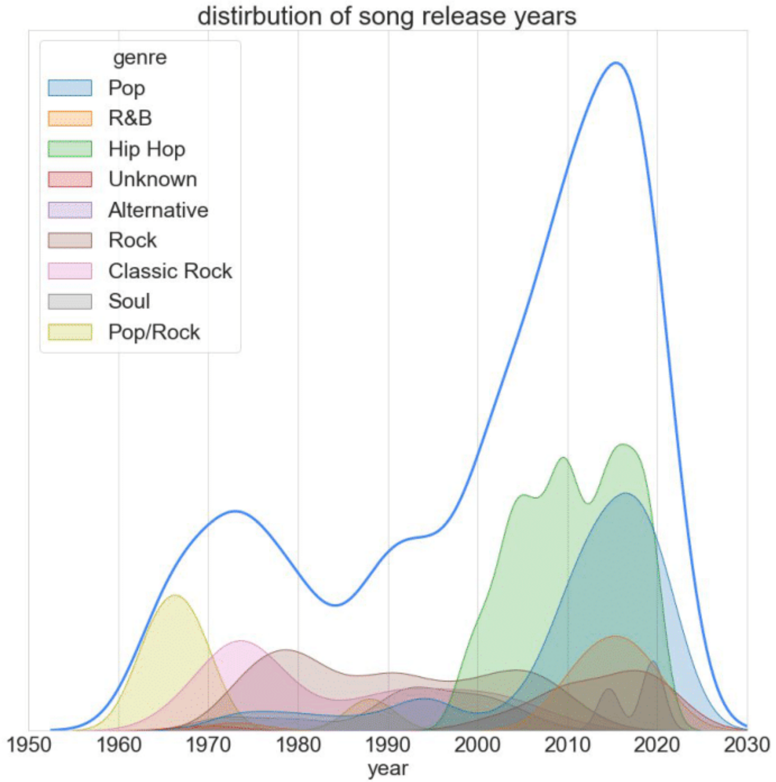
23 Artists

Divided into...

261	Albums
3097	Songs
197	Hours of music

Data Needs Analysis and Sourcing

Objective: Classify artists by vocal characteristics

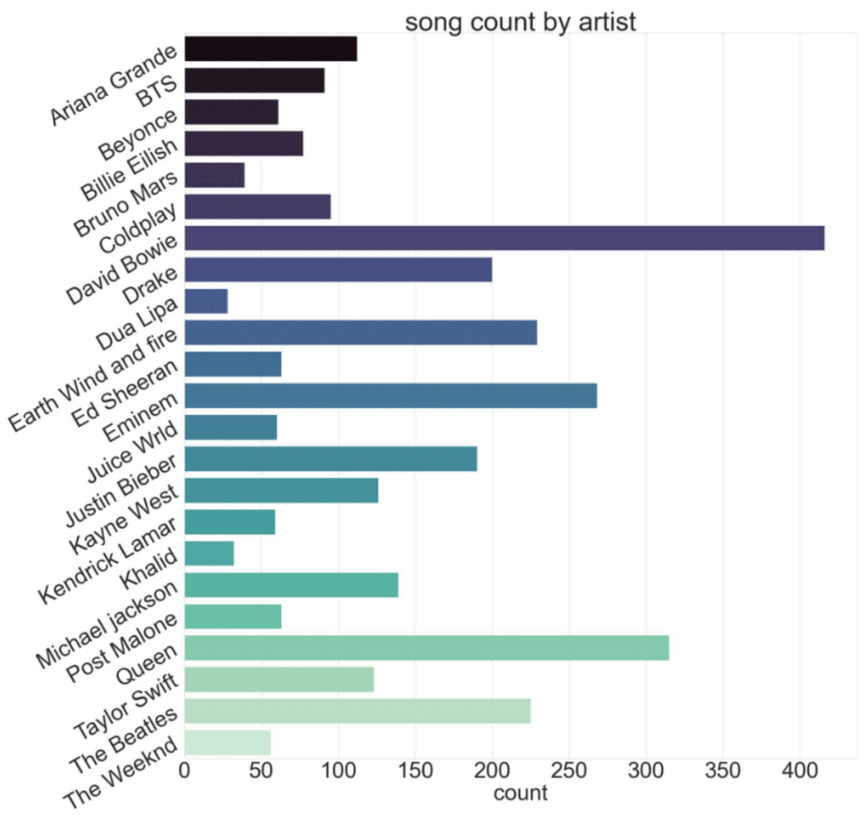


7 decades

Of music

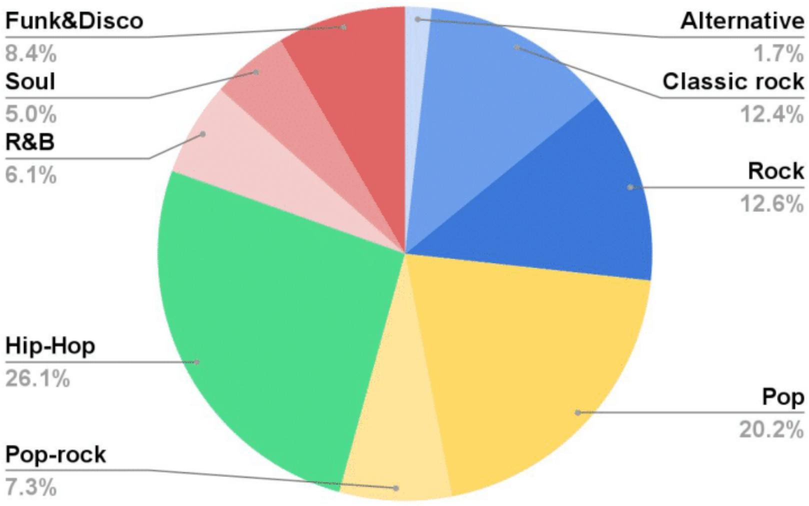
Data Needs Analysis and Sourcing

Objective: Classify artists by vocal characteristics



Data Needs Analysis and Sourcing

Objective: Classify artists by vocal characteristics



Healthy ¼ Ratio for big vocalist genres in music

Rock, Rap, R&B, Pop

4

Data Needs Analysis and Sourcing

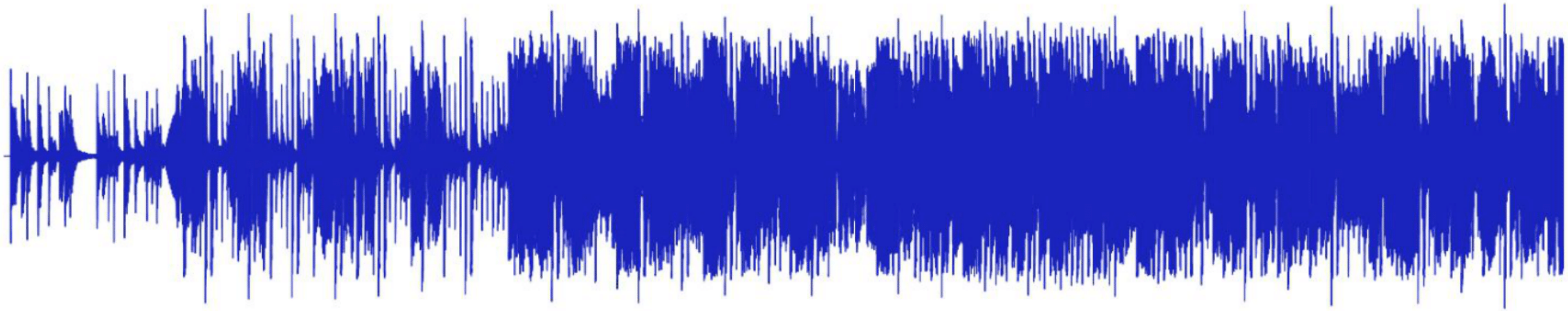
Data Cleaning

1. Remove songs that are less than 30s
2. Remove songs with multiple vocalists
3. Remove instrumental only songs

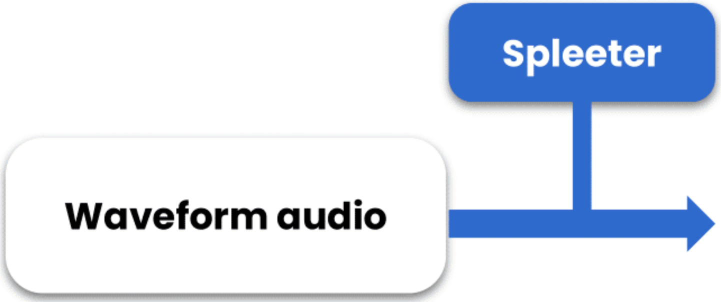
Clean waveform dataset

PreProcessing
Separation

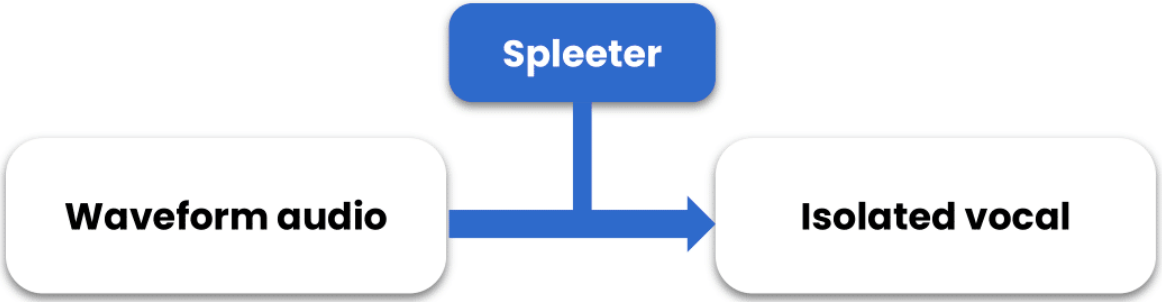
Waveform audio



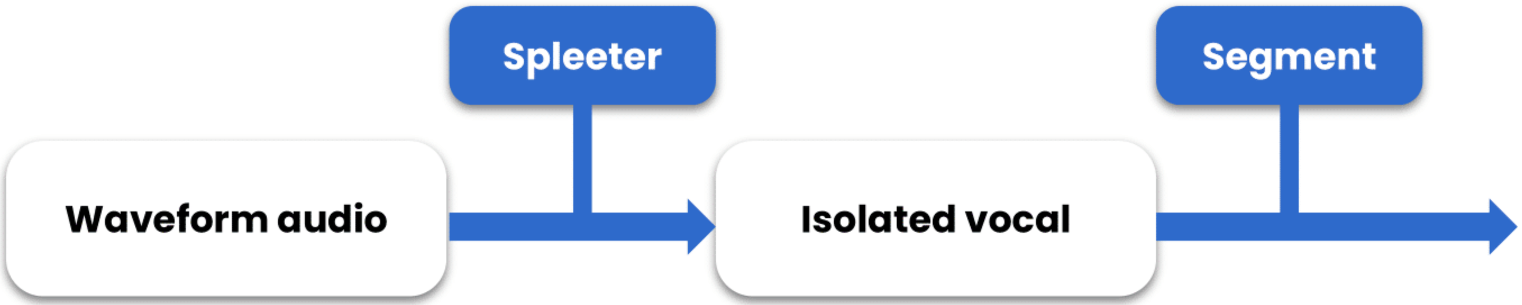
PreProcessing
Separation

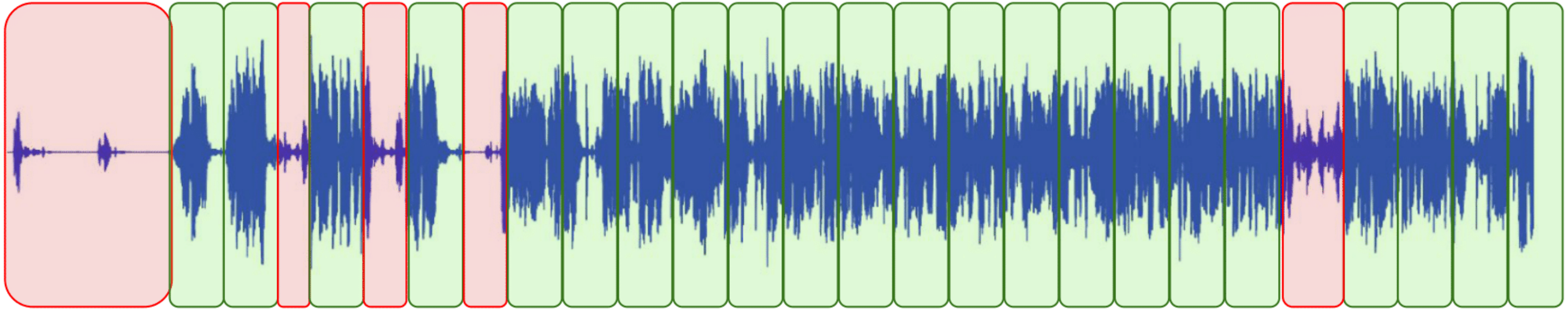


PreProcessing
Segmentation

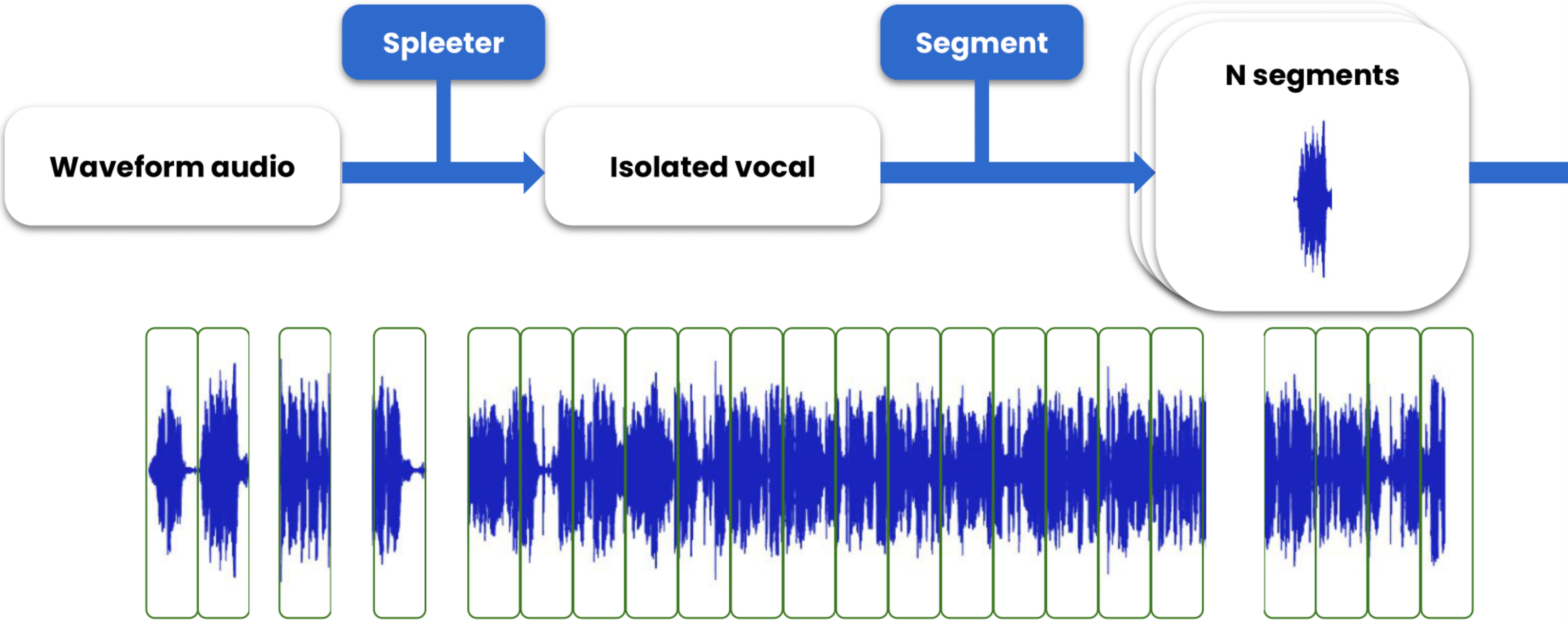


PreProcessing
Segmentation

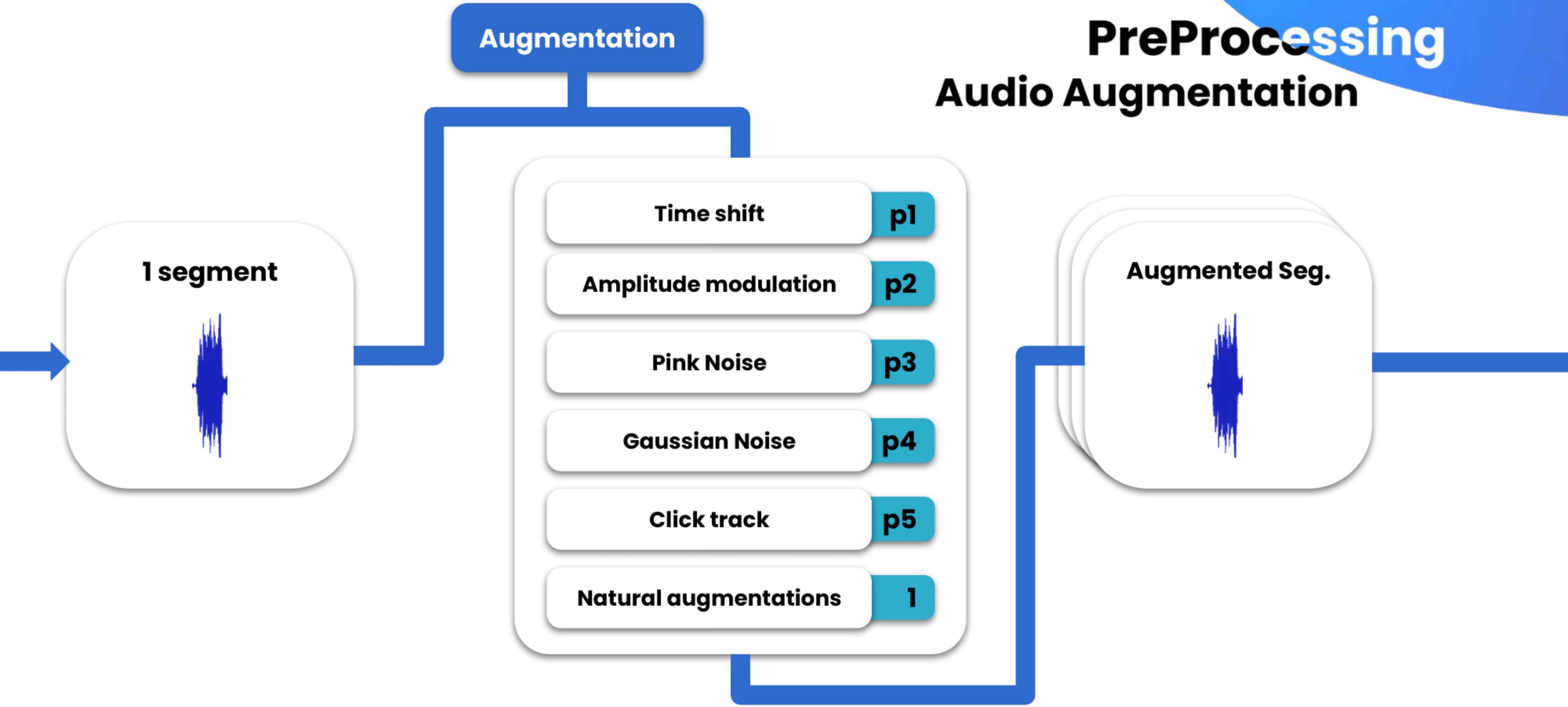




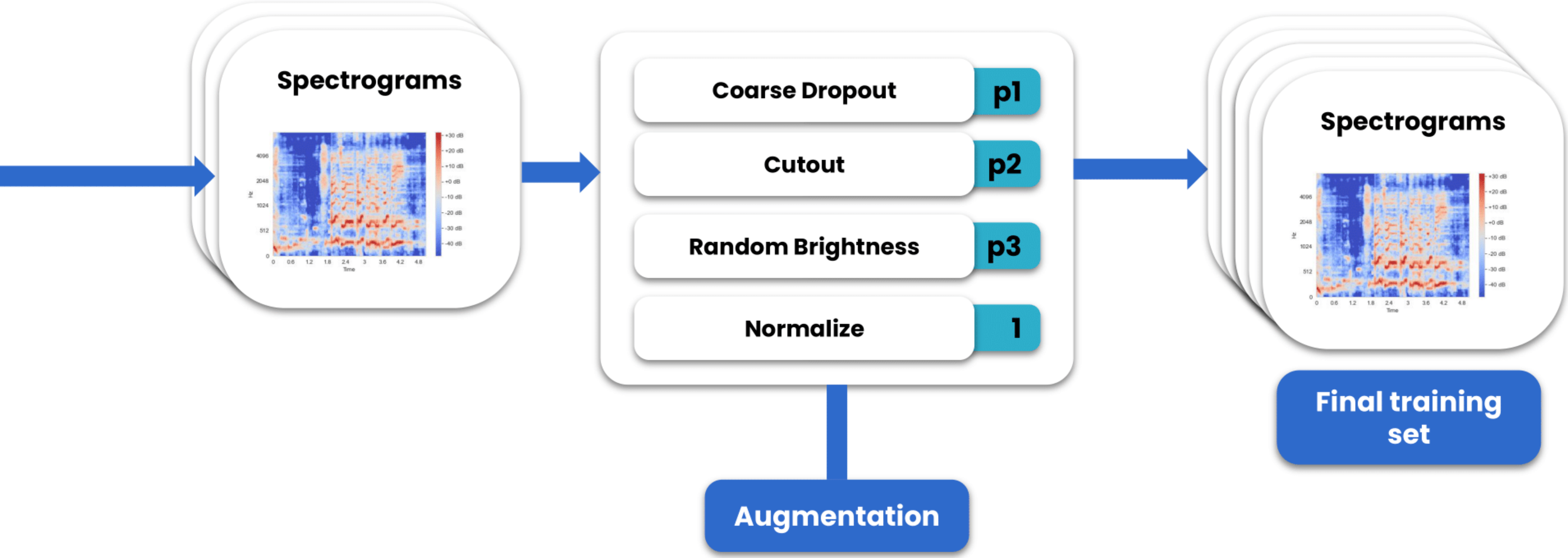
PreProcessing
Segmentation



PreProcessing
Audio Augmentation



PreProcessing
Image augmentation



Artist Classification by Spectrogram Learning

Methodology

Julien Guinot
Applied Machine learning, Sep. 2021
a1831082

Refresher

Objective and dataset

Objective

Recognize artist by a snippet of music or performance

Original data

23 Artists

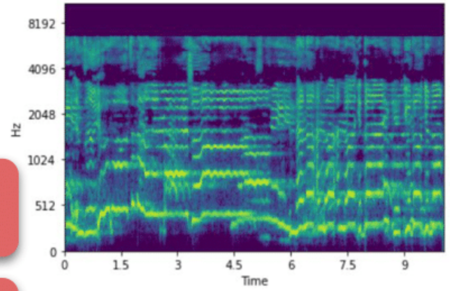
Divided into...

261	Albums
3097	Songs
197	Hours

Splitting

Segmenting

Augmenting



60 000 spectrograms of class-balanced 10s Audio clips

Training data

Method

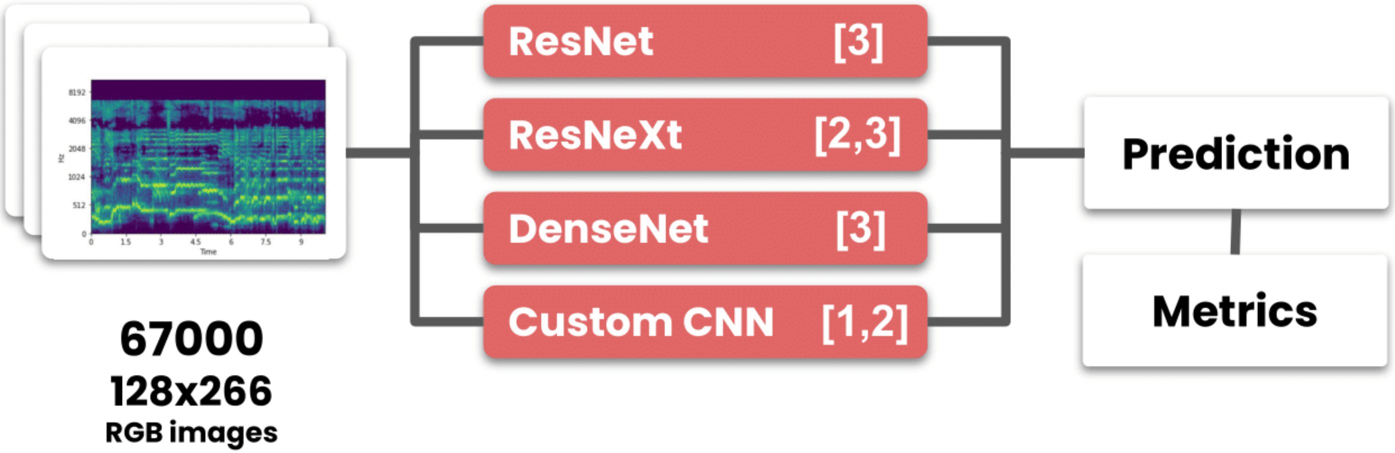
Computer Vision classification

[1] *Nasrullah, Z. and Zhao, Y., 2019, July. Music artist classification with convolutional recurrent neural networks. In 2019 IJCNN (pp. 1-8). IEEE.*

[2] Lasseck, M., 2018. Audio-based Bird Species Identification with Deep Convolutional Neural Networks. *CLEF (Working Notes)*, *2125*.

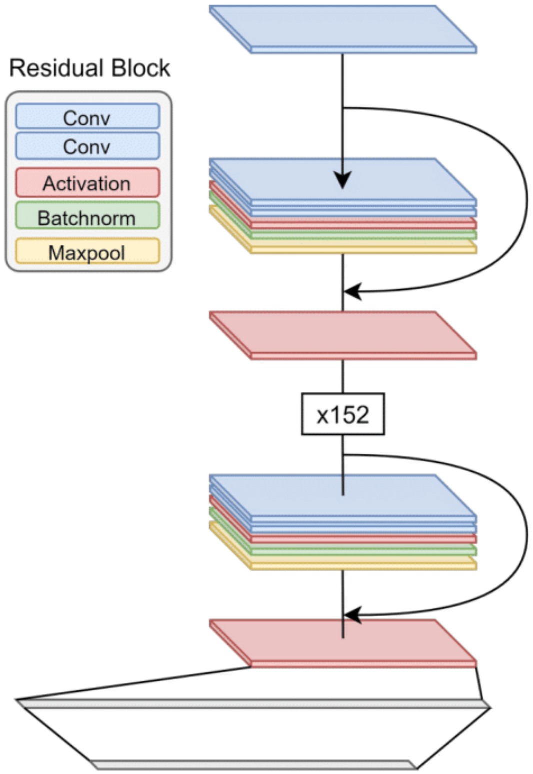
[3] Palanisamy, K., Singhania, D. and Yao, A., 2020. Rethinking cnn models for audio classification. *arXiv preprint arXiv:2007.11154*.

23 - class computer vision classification problem



Method

ResNet Architecture



Pros:

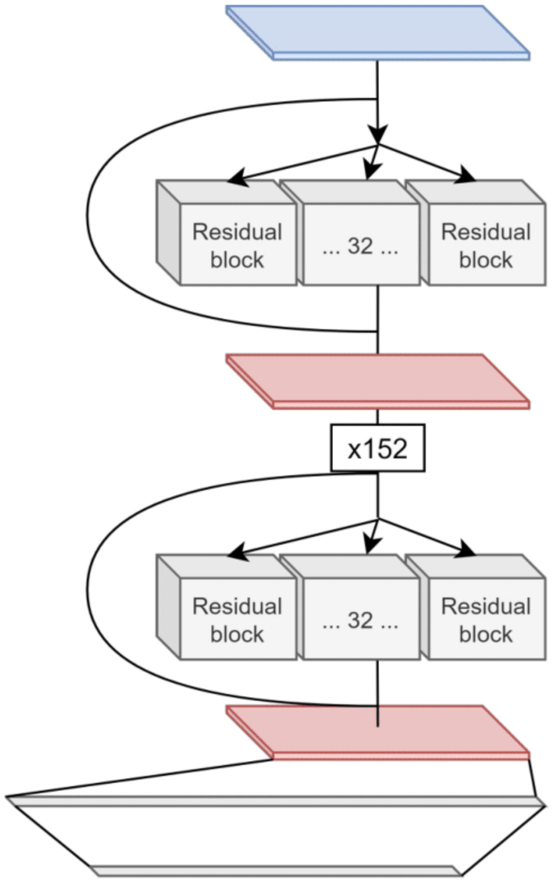
- Residual blocks aid in not forgetting features
- Very deep networks capture subtleties
- Modulable depth
- Proven to work in [2]

Cons:

- Lots of weights to train
- Potential of overfitting due to complex architecture

Method

ResNeXt Architecture



Pros:

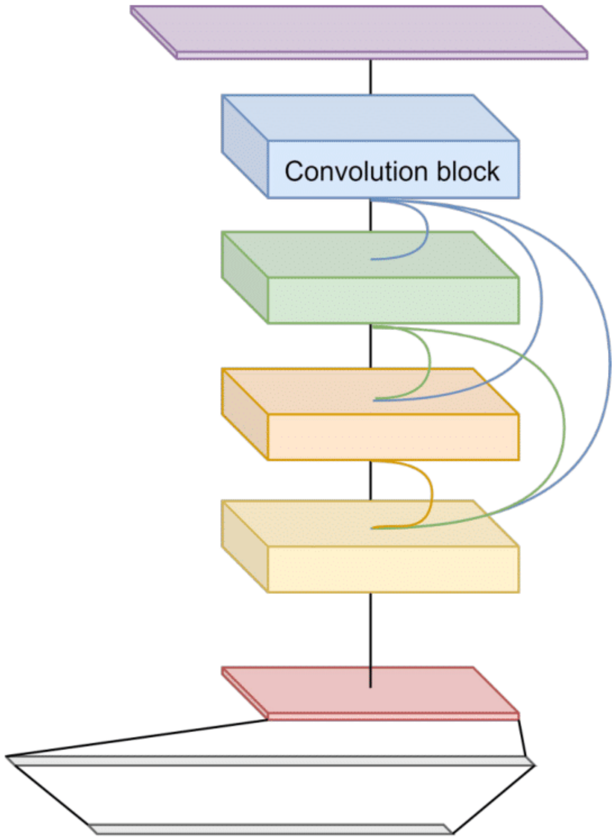
- Cardinality aids in augmenting accuracy
- Very deep networks capture subtleties
- Modulable depth
- Proven to work in [2,3]

Cons:

- More weights to train than ResNet
- Potential of overfitting due to complex architecture

Method

DenseNet Architecture



Pros:

- Encourages feature propagation
- Alleviates gradient vanishing
- Less weights
- Proven to work in [2]

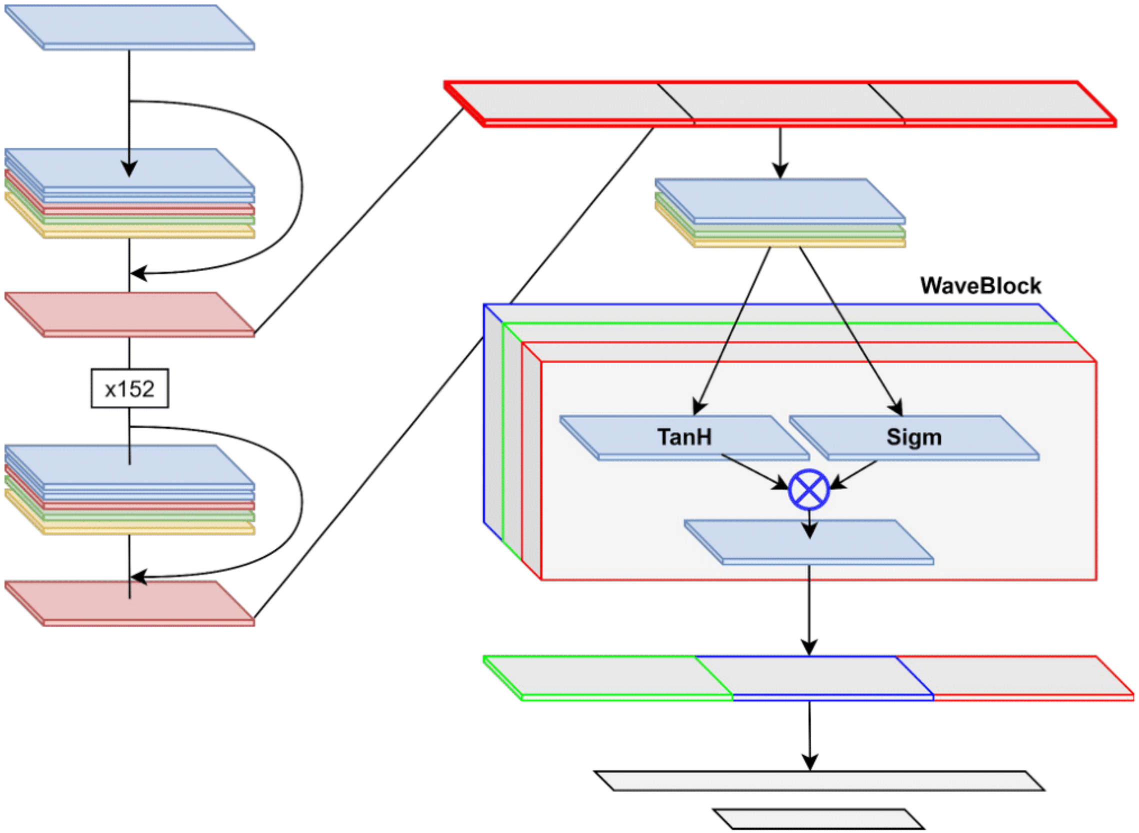
Cons:

- Computing efficiency diminishes
- Prone to overfitting

Method

[4] *https://www.kaggle.com/aikhmelnytskyy/resnet-wavenet-my-best-single-model-ensemble*

Experimental WaveNet



Training

Hyperparameter tuning

Mini-batch gradient descent with momentum and weight decay

Learning rate
Range test

Batch size
Memory fit

Momentum
Grid search

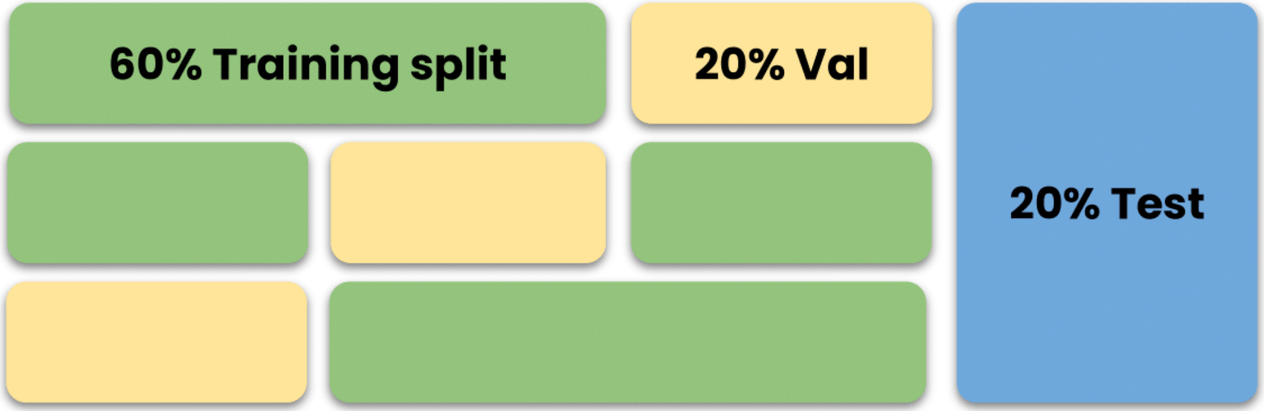
Weight decay
Grid search

Training

TTV Split & Validation

Train-Test-Validation split

K-Fold validation (K=3)



Metric

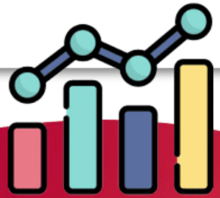
Choice and estimation

Average Accuracy

Acc = Correct/Total

- [1] On songs — 84%
- [2] On Bird species — 69%
- [3] On Genre classif. — 91%
- [4] On birds — 88%

70% Goal



Macro and micro Averaging F1, Precision, Recall

Measuring global metrics considering class distribution or not

One Vs all accuracy

Measuring class-specific metrics

Confusion matrix

Great for understanding Misclassifications

Artist Classification by Spectrogram Learning

Software and results

Julien Guinot
Applied Machine learning, Sep. 2021
a1831082

Refresher

Dataset and Model

Objective

Recognize artist by a snippet of music or performance

Original data

23 Artists

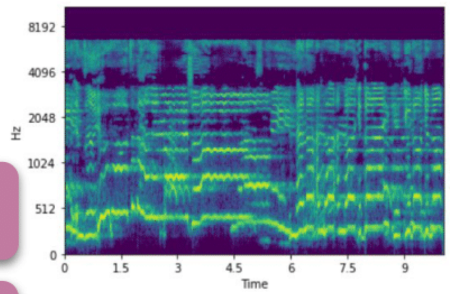
Divided into...

261	Albums
3097	Songs
197	Hours

Splitting

Segmenting

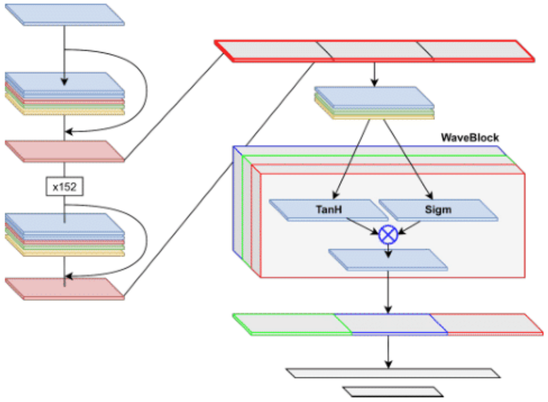
Augmenting



60 000 spectrograms of class-balanced 10s Audio clips

Training data

ResNet + WaveNet



70% Accuracy objective

Software

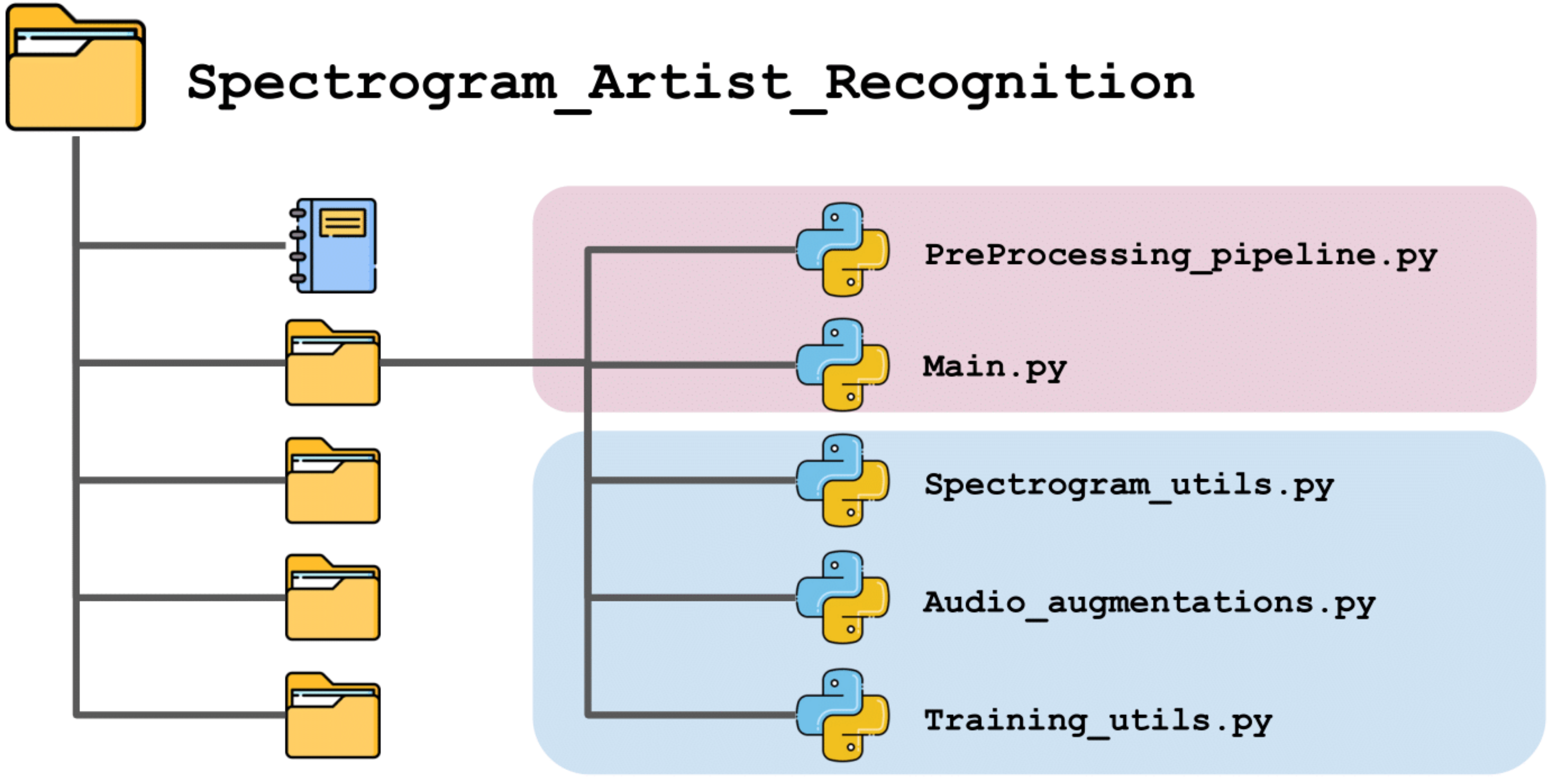
Project structure

Spectrogram_Artist_Recognition

- **Main Jupyter Notebook** (Training and Evaluation)
- **Src** (Source code for preprocessing and training)
- **Config** (Config files for preprocessing)
- **Data** (Isolated, split and unsplit vocals)
- **Notebooks** (Notebooks for dataset exploration)

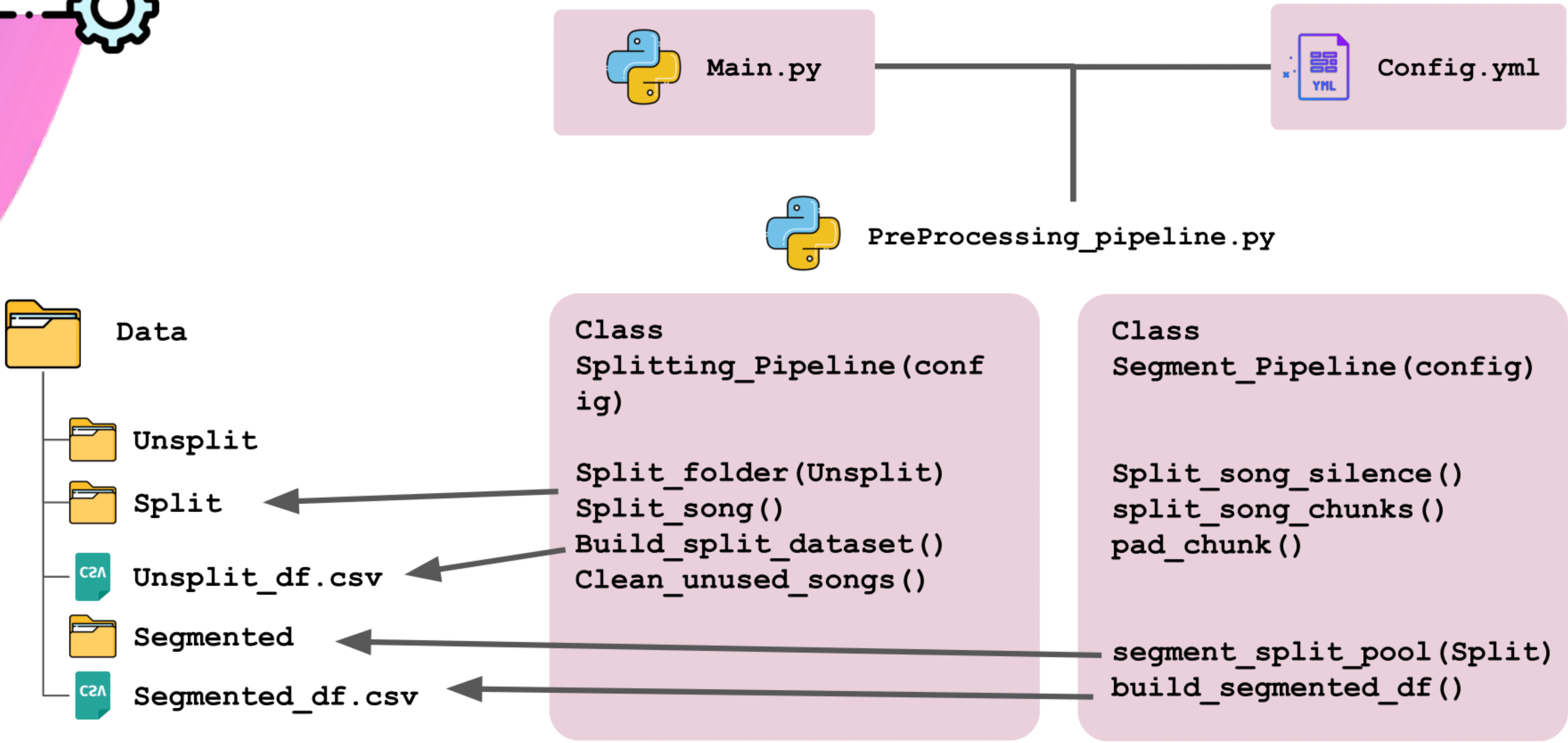
Software

Project structure



Software

Code structure - Preprocess



Software

Code structure - Train

.ipynb

Spectrogram_utils.py

Audio_augmentations.py

Training_utils.py

Data

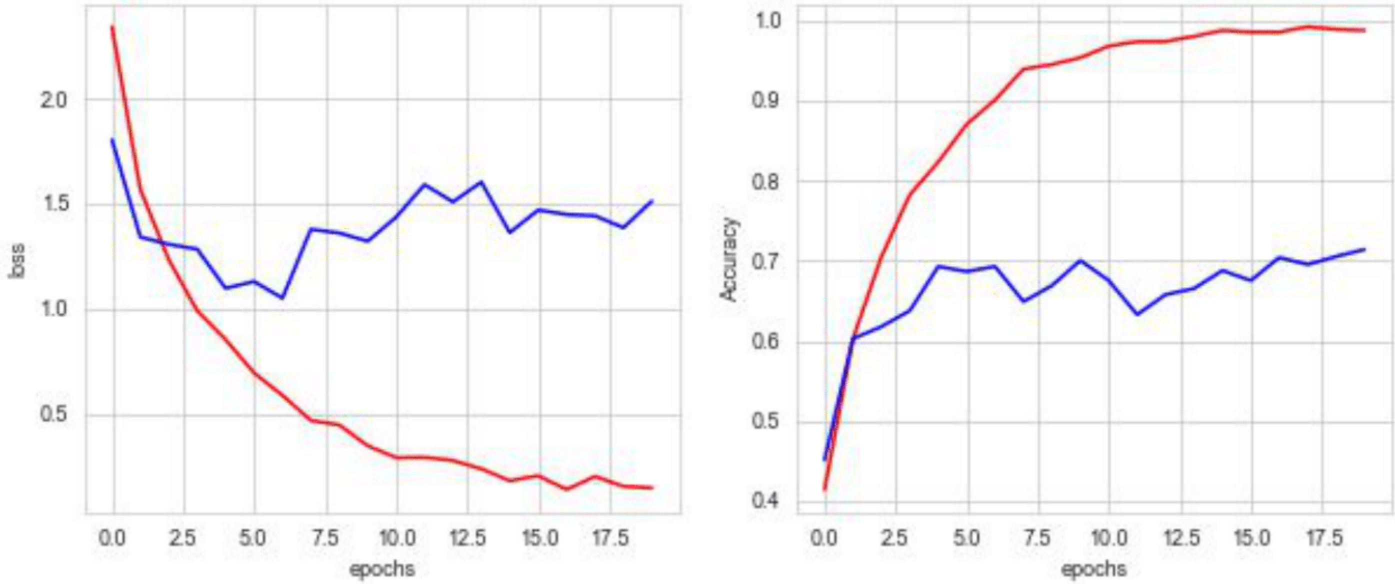
- Segmented
- Segmented_df.csv

```
Class RecoNetDataset

__getitem__()


Class RecoNetTestDataset

__getitem__()
No transformations
```

```
Class RecoNetTrainer

get_model()
configure_optimizers()
generate_dataloaders()
train_step()
train()
evaluate()
load_model_checkpoint()
plot_losses()
score()
```

Results

Training



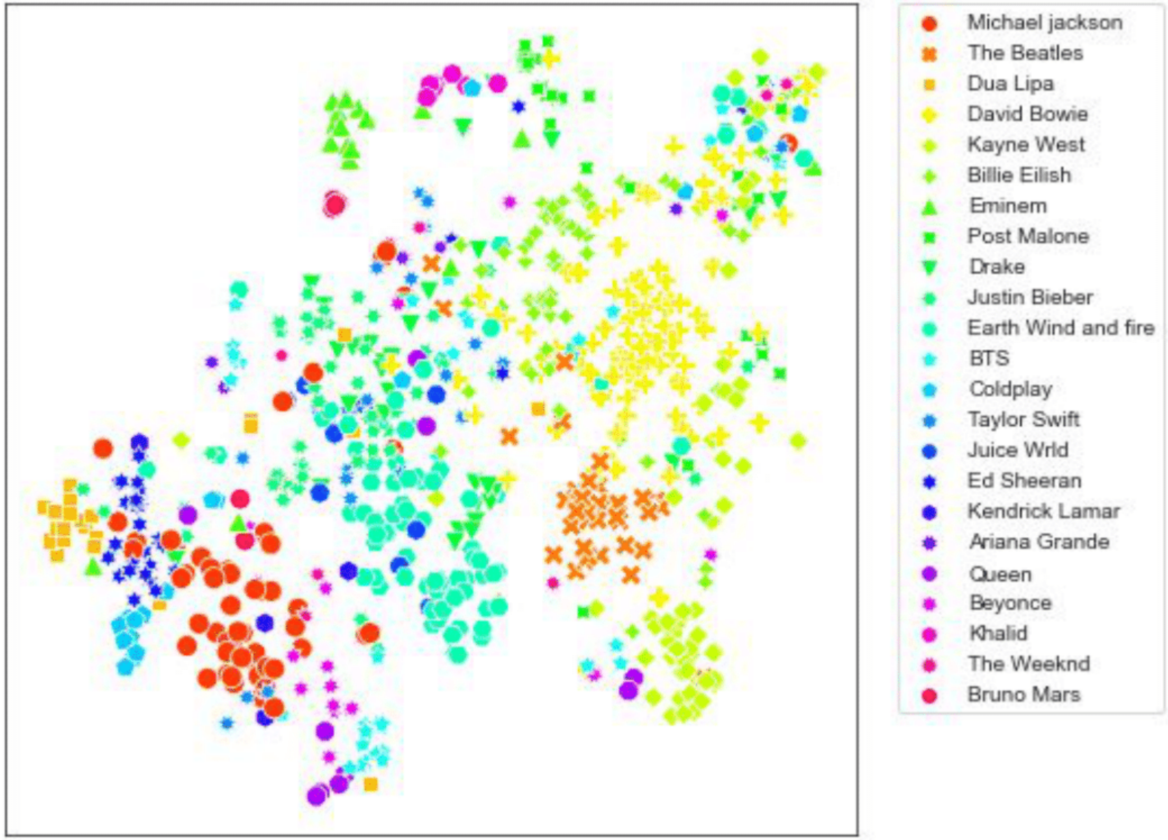
Training curves - Loss and Accuracy

- **Limited number of epochs**
- **Clear overfitting**
- **70% validation accuracy**

Is the model learning?

Results

Training

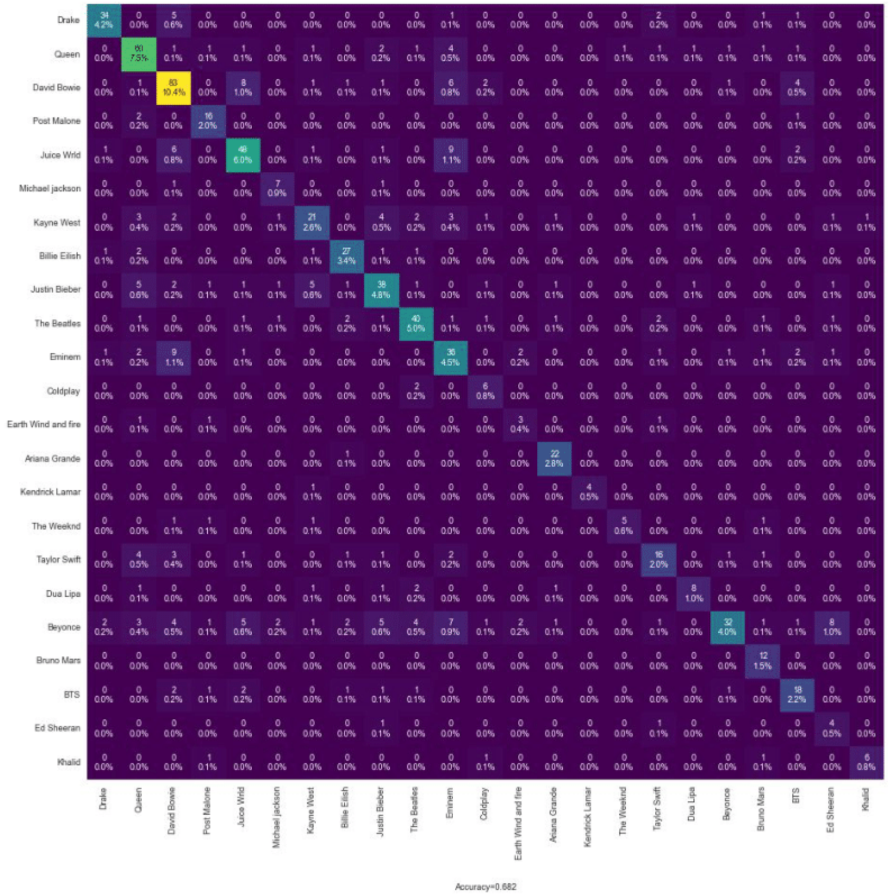


t-SNE on last connected layer

- **Clear separation for some artists, not so much for others**
- **Artists with distinct voices are well separated**
- **The model is learning relevant features, but not deep enough**

Results

Multi-class performance on test set



type	accuracy	precision	recall	f1
Macro	0.682	0.679255	0.698233	0.671996
Weighted	0.682	0.704398	0.682500	0.679423

68% Accuracy

We fall just short of our 70% objective, but technical difficulties and overfitting explain that quite well. The model can perform better

Results

Single-class performance on test set

class	accuracy	precision	recall	f1
Drake	0.98	0.87	0.77	0.82
Queen	0.95	0.71	0.78	0.74
Post Malone	0.92	0.70	0.77	0.73
David Bowie	0.99	0.70	0.84	0.76
Juice Wrld	0.95	0.71	0.71	0.71
Michael jackson	0.99	0.58	0.78	0.67
Kayne West	0.96	0.62	0.51	0.56
Billie Eilish	0.98	0.75	0.82	0.78
Justin Bieber	0.95	0.66	0.64	0.65
The Beatles	0.97	0.74	0.75	0.75
Eminem	0.93	0.52	0.63	0.57
Coldplay	0.99	0.46	0.75	0.57
Earth Wind and fire	0.99	0.43	0.50	0.46
Ariana Grande	0.99	0.81	0.96	0.88
Kendrick Lamar	1.00	1.00	0.80	0.89
The Weeknd	0.99	0.83	0.56	0.67
Taylor Swift	0.97	0.64	0.53	0.58
Dua Lipa	0.99	0.73	0.57	0.64
Beyonce	0.93	0.86	0.39	0.53
Bruno Mars	0.99	0.60	1.00	0.75
BTS	0.97	0.60	0.67	0.63
Ed Sheeran	0.98	0.25	0.67	0.36
Khalid	0.99	0.86	0.67	0.75

The model performs quite well on One-vs-all classifications depending on artists

Our metric of interest, recall, can be pushed up to 100% for artists with distinct vocals (Bruno Mars, Ariana Grande)

Beyonce, who has had a section of her discography in the spice girls, shows low recall, which is unfortunate but coherent.

Overall, the Model has learned relevant features but could use more data and could be simplified to reduce overfitting.