

Détection du genre musical à l'aide des techniques de Machine Learning

Présenté par:

- BENNABI Malek
- LUCCHINI Gabriel
- PLUVEN Nicolas

Plan

- Introduction
- Données Traitées
- Approches Proposées
- Résultats
- Conclusion

Objectif:

Ce projet consiste en la classification de chanson selon son genre musical parmi les 20 genres disponibles.

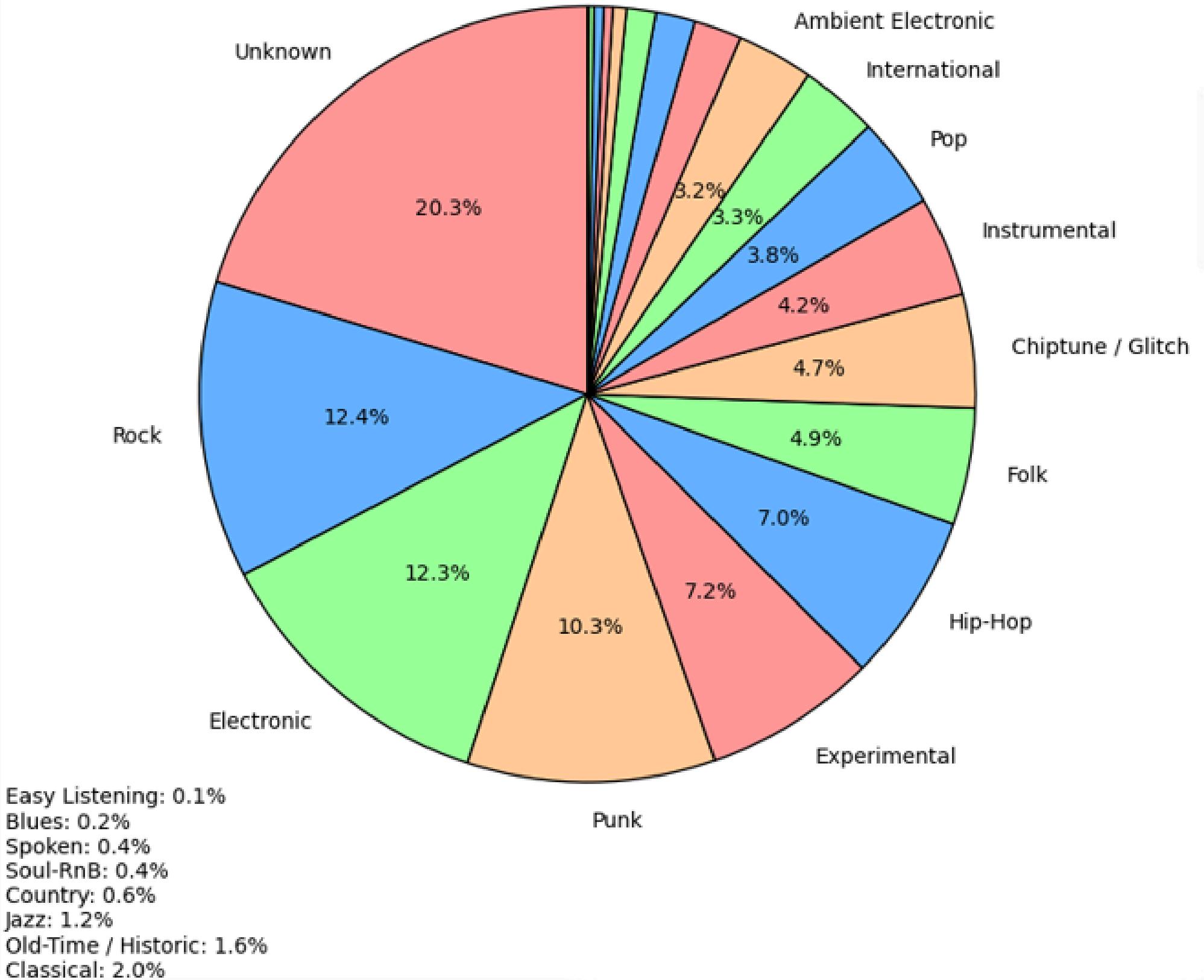
Jeux de Données

lewtun/music genres - Hugging face:

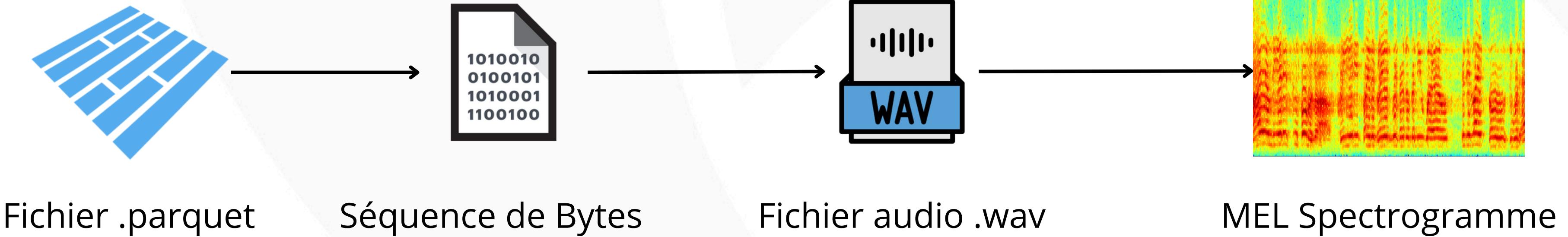
- Format .parquet
- < Bytes, song_id, genre_id, genre >
- 20 Genres Musicaux
- 25000 enregistrements (~9.8 Gb)
- ~30s par audio

Distribution

Figure: Distribution des genres musicaux dans le dataset

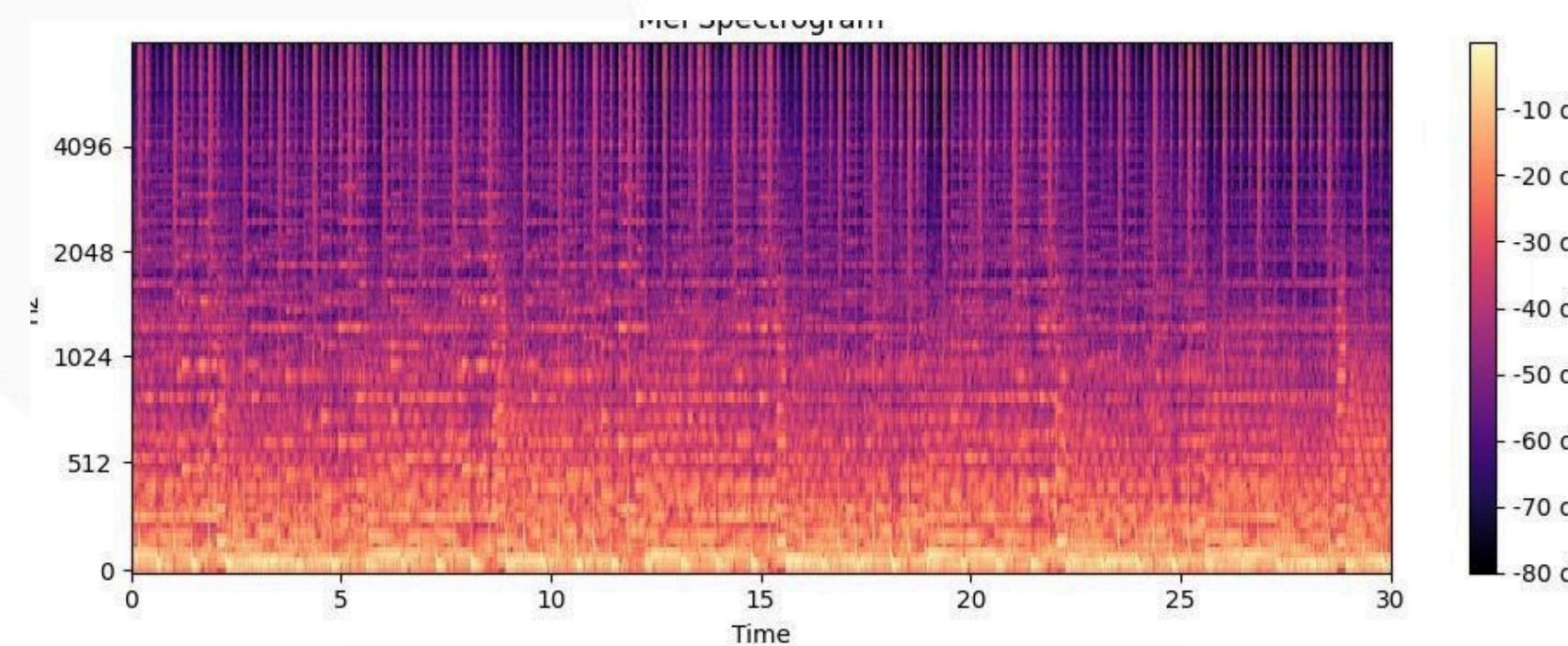


Pre-Traitement



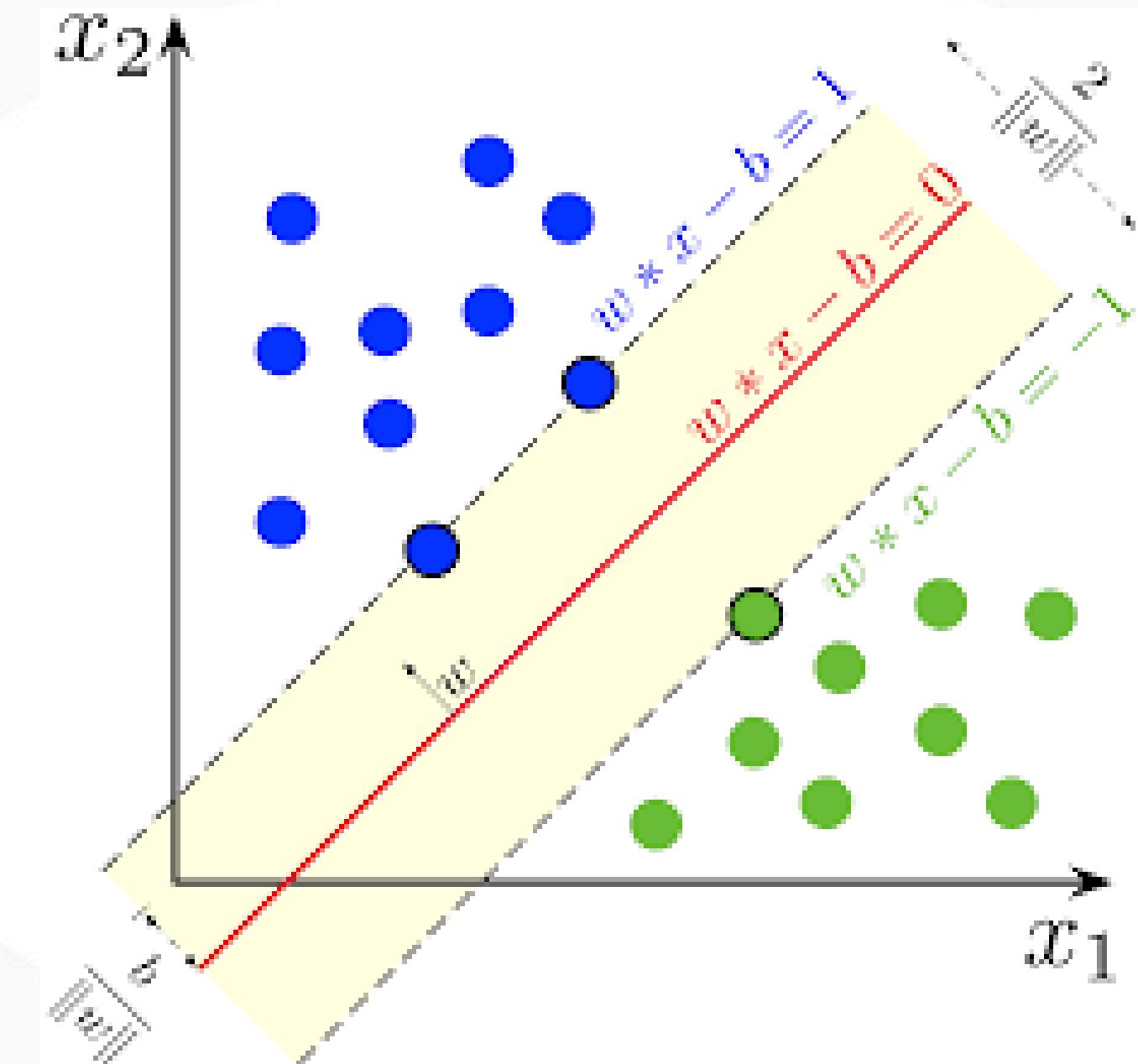
MEL-Spectrogramme

C'est une représentation visuelle de l'énergie spectrale d'un signal audio dans le domaine des fréquences, ajustée selon l'échelle de Mel, qui correspond davantage à la manière dont l'oreille humaine perçoit les sons.



Approche SVM

1. Prétraitement des données
2. Division des ensembles d'entraînement et de test : 70/30 + crossValidation
3. Ajustement des hyperparamètres (niveau de régularisation, échelle pour RBF, noyaux)
4. Utilisation du meilleur modèle pour prédire l'ensemble de test

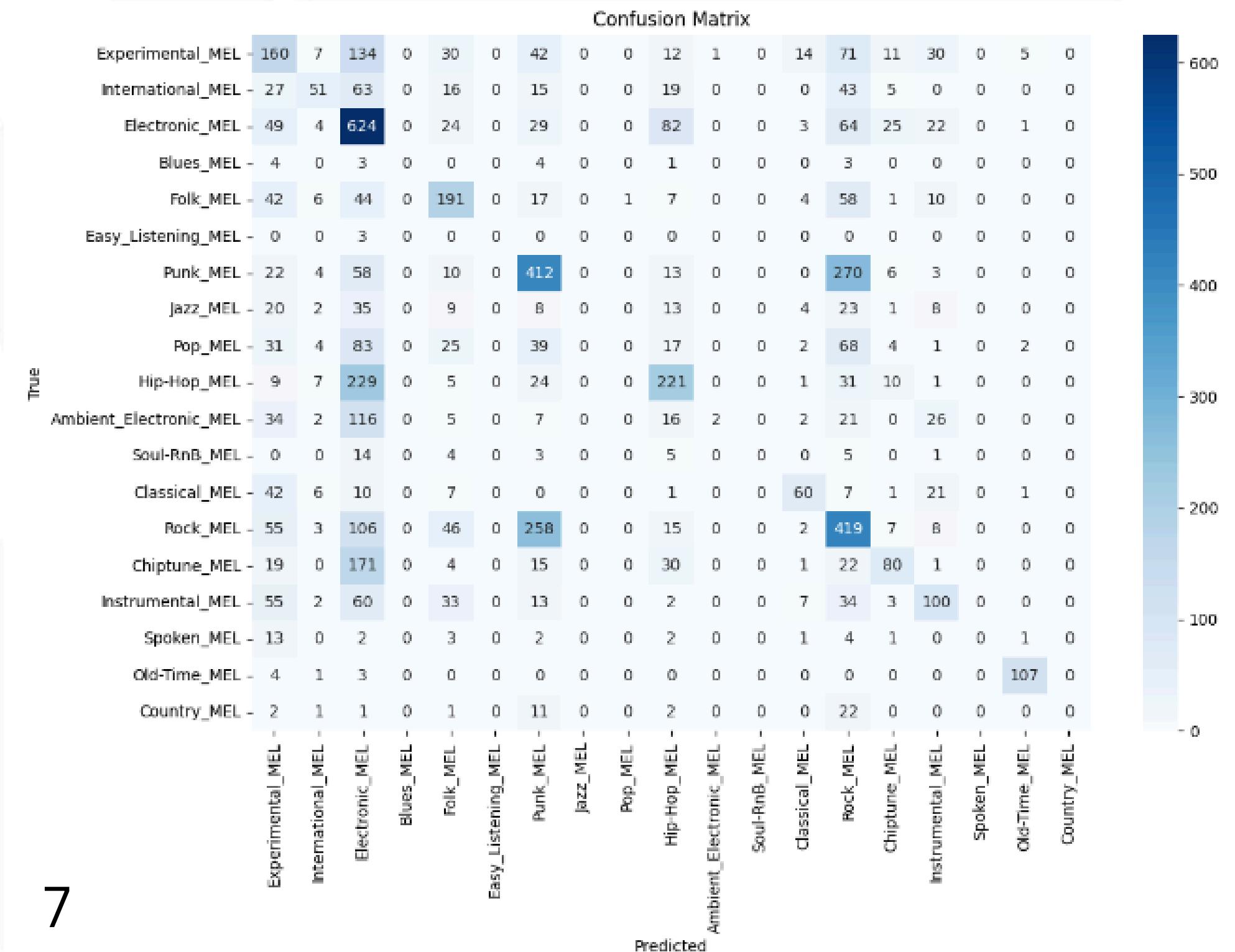


SVM - Résultats

Accuracy de 41%

Limites :

- Peu adapté aux grands ensemble de données comme le notre
- Meilleur pour les problèmes de classification binaire.



Approche CNN : From Scratch

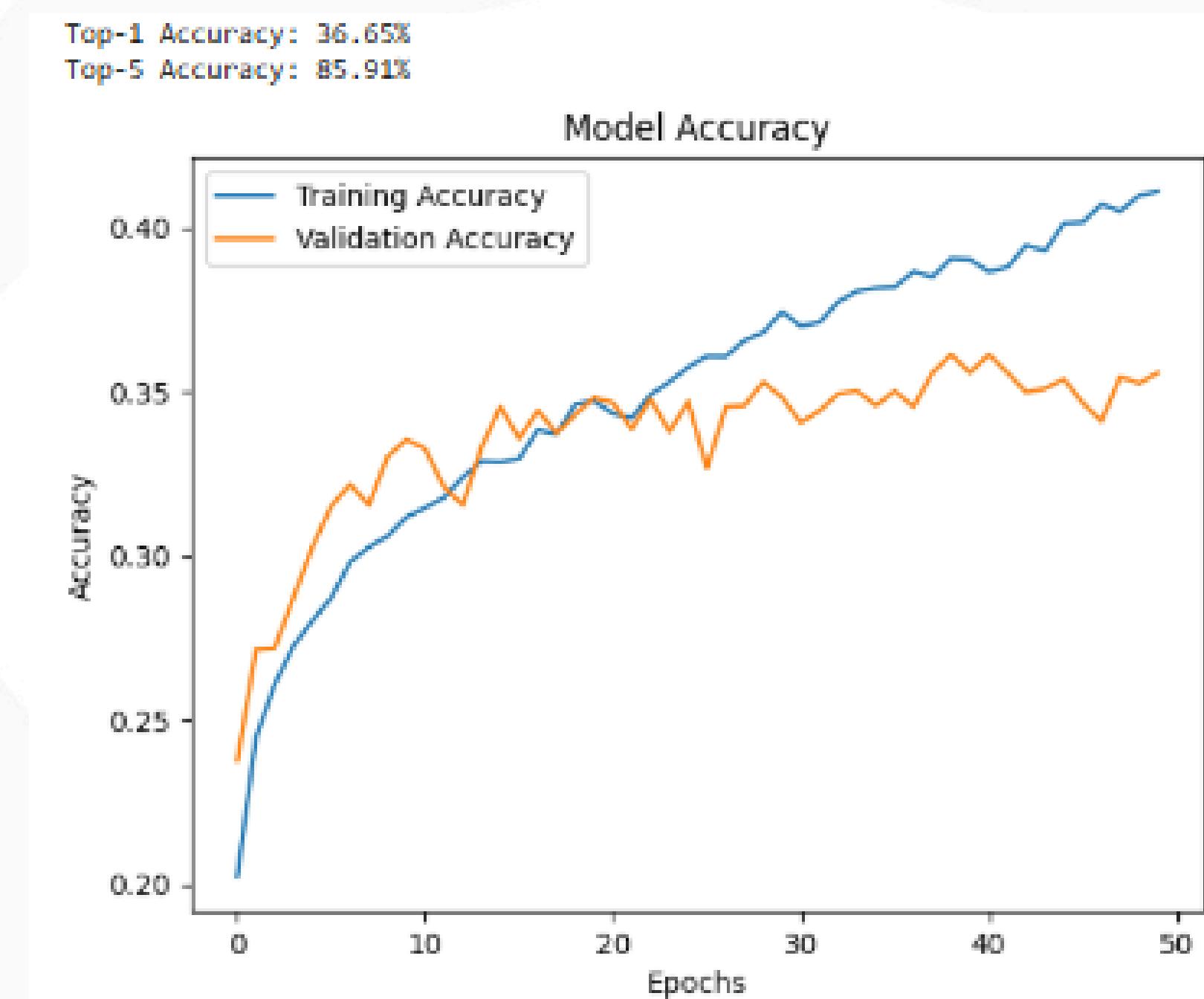
- Couche d'entrée : Images RGB de taille (128x128)
- Couches convolutives :
 - Couche1: Conv2D 32 filtres:
 - Couche2: Conv2D 64 filtres
 - Couche3: Conv2D 128 filtreschacune des couches est suivie d'une couche MaxPooling2D et Dropout pour la régularisation.
- Couche d'aplatissement : Matrice 2D --> vecteur.
- Couches denses :
 - Couche1: 128 neurones et activation ReLU, par Dropout
 - Couche 2: 20 neurones, utilisant l'activation softmax pour la classification.

CNN from scratch – Results

Accuracy de 36.8%

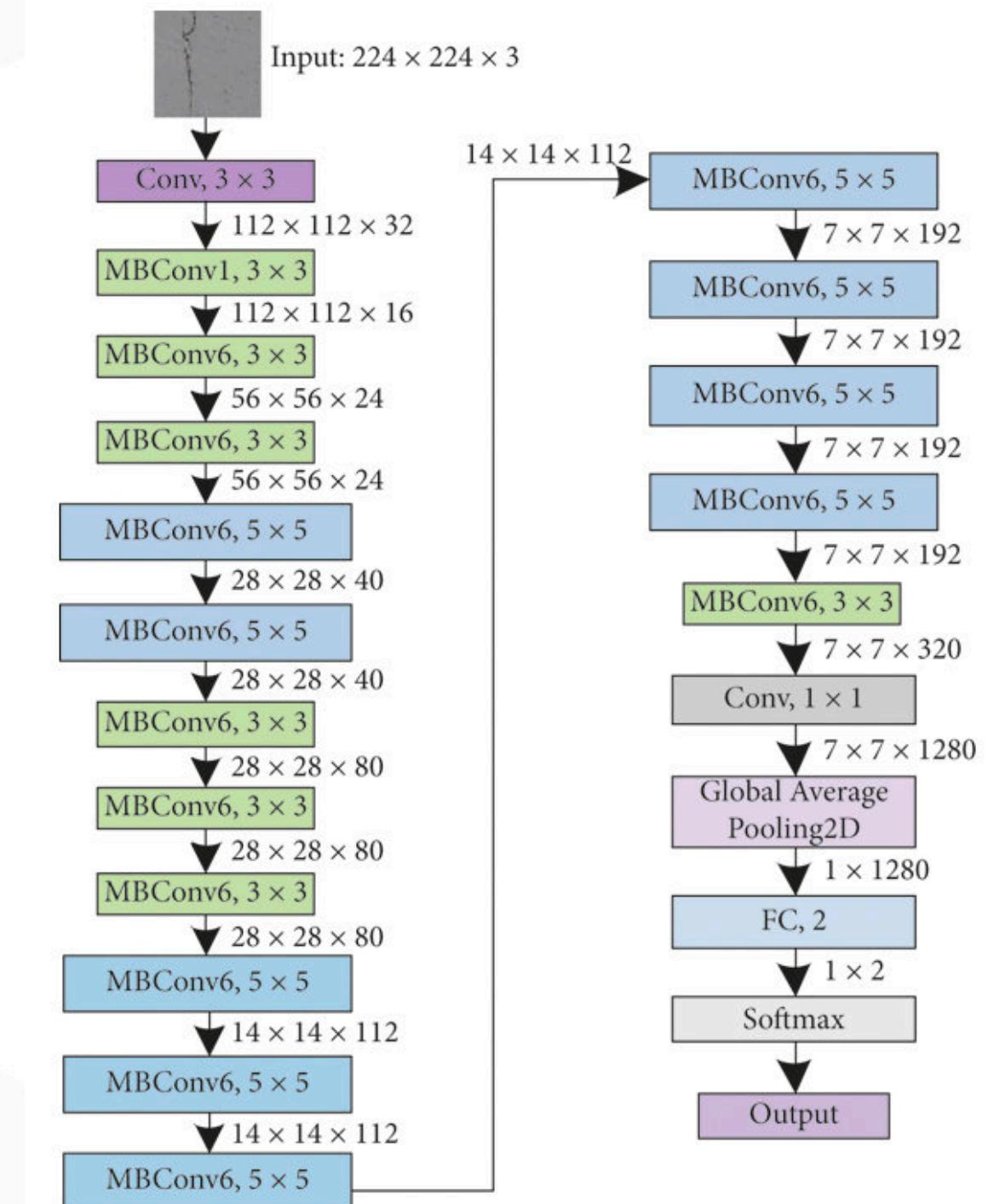
Limites :

- Peu adapté aux grands ensemble de données.
- Lenteur d'entraînement
- risque de sur-apprentissage.

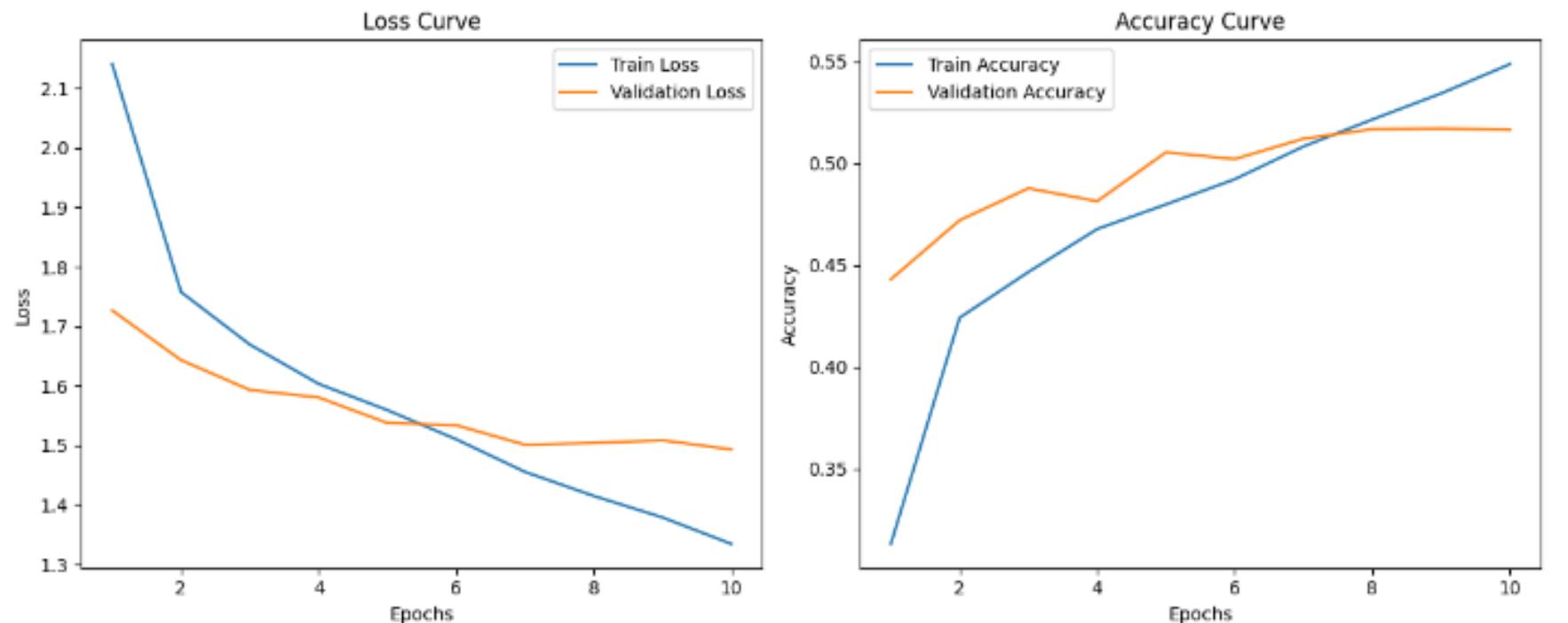


Approche CNN : EfficientNet

1. Chargement des données
2. Prétraitement des données : data augmentation + normalisation
3. Division en ensemble de train, validation et test : 70/15/15
4. Chargement du model préentraîné EfficientNet
5. Entrainement -> Ajustement des hyperparamètres
6. Evaluation finale sur l'ensemble de test



EfficientNet - Résultats :



- Accuracy de 51%
- Entraîné sur 10 epochs

Paramètres :

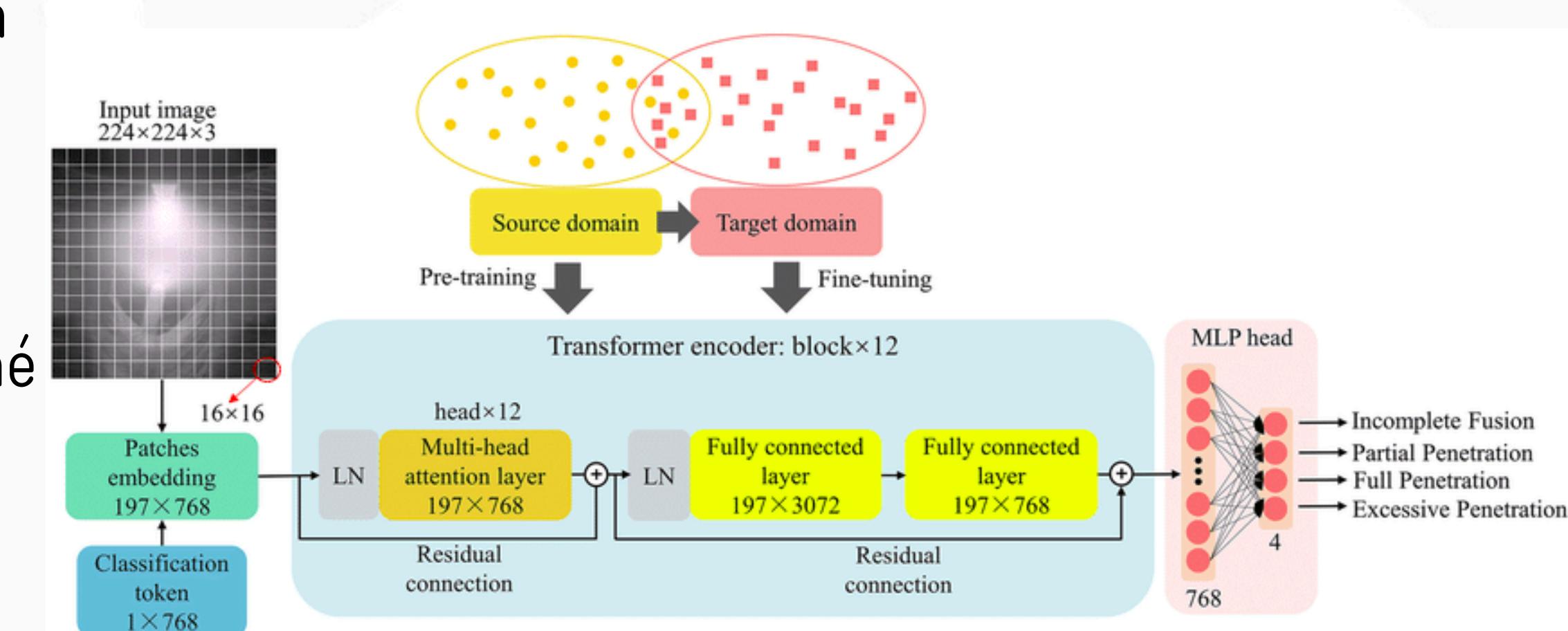
- taille des images : 128*128
- taille du batch : 16
- optimiseur : Adam ($\text{lr}=1\text{e}-4$)

Confusion Matrix

TRUE	Ambient_Electronic_MEL	Blues_MEL	Chiptune_MEL	Classical_MEL	Country_MEL	Easy_Listening_MEL	Electronic_MEL	Experimental_MEL	Folk_MEL	Hip-Hop_MEL	Instrumental_MEL	International_MEL	Jazz_MEL	Old-Time_MEL	Pop_MEL	Punk_MEL	Rock_MEL	Soul-RnB_MEL	Spoken_MEL
Ambient_Electronic_MEL	26	0	5	2	0	0	47	10	1	7	11	1	0	1	0	2	2	0	0
Blues_MEL	0	0	1	0	0	0	0	0	0	0	1	0	3	0	0	1	1	0	0
Chiptune_MEL	6	0	102	0	0	0	46	7	0	8	3	1	0	0	0	2	3	0	0
Classical_MEL	0	0	1	69	0	0	0	0	2	0	0	8	1	0	0	0	0	0	1
Country_MEL	0	0	0	0	3	0	0	0	0	7	0	0	5	0	0	3	5	3	0
Easy_Listening_MEL	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
Electronic_MEL	17	0	35	2	0	0	236	23	5	71	11	6	5	0	5	8	29	0	0
Experimental_MEL	10	0	7	13	0	0	38	130	7	9	17	6	6	2	4	5	24	0	2
Folk_MEL	0	0	0	2	1	0	4	6	139	1	11	5	3	0	9	0	18	0	0
Hip-Hop_MEL	3	0	4	0	0	0	32	6	0	210	3	4	1	1	0	4	8	0	2
Instrumental_MEL	10	0	1	19	0	0	25	19	2	1	60	3	2	0	0	1	11	0	0
International_MEL	1	0	2	1	0	0	18	1	9	10	2	65	3	1	2	2	5	0	4
Jazz_MEL	1	0	0	6	0	0	4	2	2	3	5	10	11	1	2	2	4	0	0
Old-Time_MEL	0	0	0	1	0	0	0	0	0	0	1	0	51	0	0	0	0	0	0
Pop_MEL	4	0	4	4	0	0	20	8	20	9	8	4	0	1	13	5	47	0	0
Punk_MEL	0	0	3	1	0	0	12	10	4	6	2	10	3	0	4	188	129	1	0
Rock_MEL	8	0	6	2	1	0	31	25	20	6	11	10	2	2	14	89	212	0	1
Soul-RnB_MEL	0	0	1	0	0	0	6	0	0	2	0	3	1	0	1	2	1	0	0
Spoken_MEL	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	2

Approche Transformer : ViTb16

1. Chargement des données
2. Prétraitement des données : data augmentation + normalisation
3. Division en ensemble de train, validation et test : 70/15/15
4. Chargement du modèle préentraîné
ViTb16
5. Entrainement → Ajustement des hyperparamètres
6. Evaluation finale sur l'ensemble de test

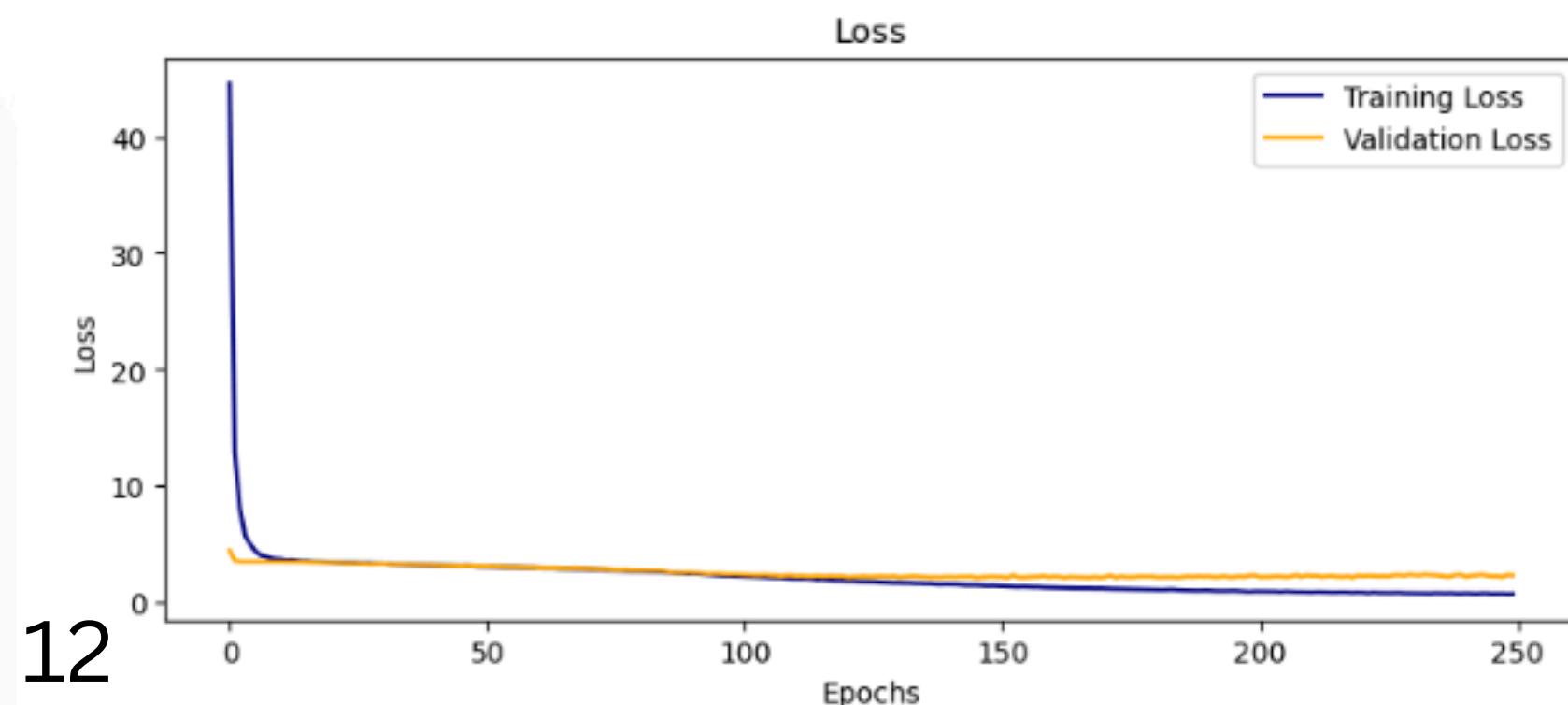
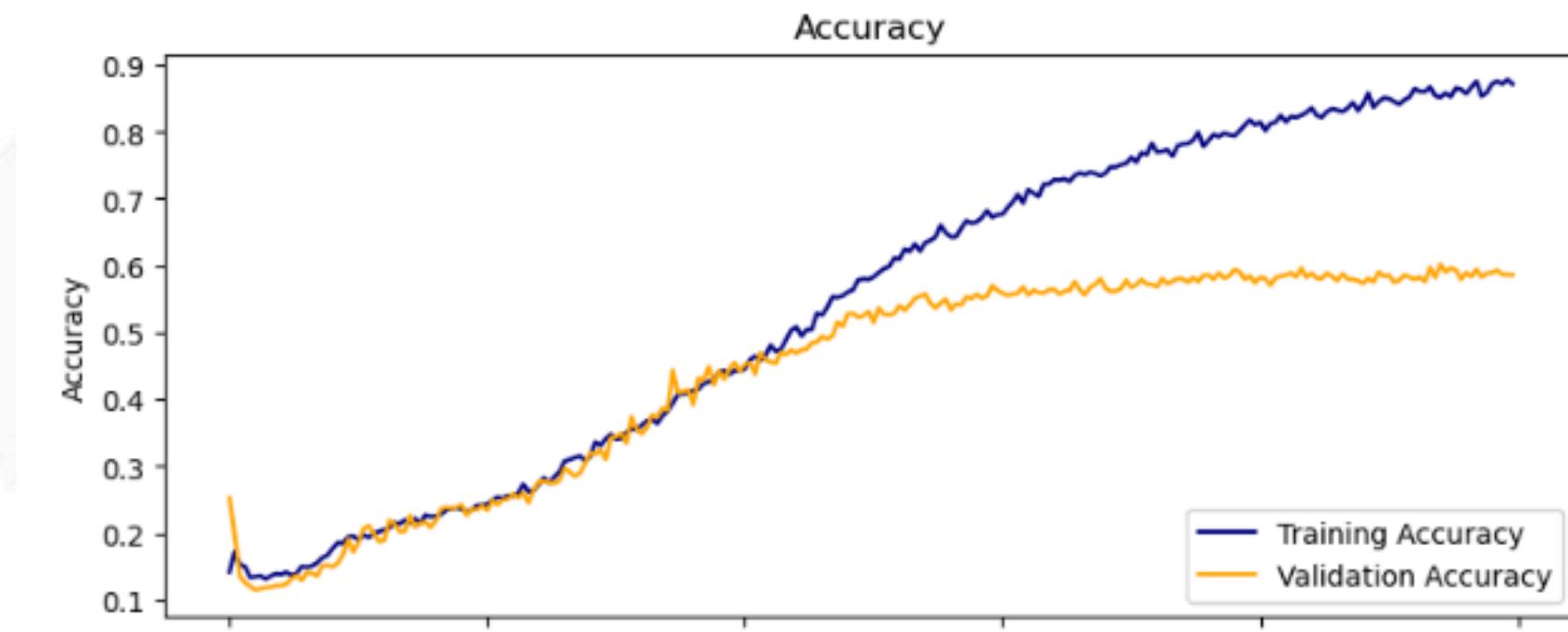


ViTb16 - Résultats :

- Accuracy de 62%
- Entraîné sur 250 epochs

Paramètres :

- taille image : 256*256
- taille batch : 64



Comparaison des performances

Modèle	Accuracy	Temps d'entraînement	Epoch
SVM	41%	3min	-
CNN from scratch	36.8%	33min	50
EfficientNet	51%	1h12	10
ViT_b_16	62%	4h23	110

Défis Rencontrés

- Limites matérielles : utilisation de kaggle (Mémoire limitée, GPU limitée, ...)
- Corruption de certains fichiers .wav
- Dataset non équilibré : beaucoup de rock et hip/hop, peu de jazz et blues
- Une musique peut correspondre à plusieurs genres -> évaluer le top-3/top-5

Conclusions + Perspectives

Conclusions :

- Le ViT obtient les meilleures performances pour la classification des genres musicaux
- Le SVM est performant et demande peu de ressource

Perspectives :

- Homogénéiser le dataset
- Combinaison des données MEL et MFCC
- Combiner différents Datasets pour une meilleur généralisation
- Tester d'autres architectures comme ResNet ou Transformer audio
- Utiliser des données non supervisées

Merci pour votre attention.