Filière : Génie Logiciel

Rapport PFA

Sujet à thematique IA : Classification des genres de musique

Réalisé par :
Bahaoui Aymen-Hatim
Dadda Ziyad

Encadrante :
Pr. Sanaa El Fkihi

12 juin 2020

Table des matières

Introduction **1**

1 Contexte général du projet **2**

1.1 Introduction 2

1.2 Problématique et motivation 2

1.3 Etude de l'existant 3

1.3.1 Music IP Mixer 3

1.3.2 Clementine 3

1.3.3 MusicBee 4

1.4 Revue de la littérature : 4

1.5 Conclusion 5

2 Théorie du traitement du signal Audio **6**

2.1 Definition 6

2.1.1 Forme Ondulatoire 6

2.1.2 Propriétés : fréquence , amplitude : 7

2.1.3 Conversion Analogique Numérique : 8

2.2 Transformée de Fourier : 8

2.3 Short time Fourier transform STFT : 9

2.4 L'Échelle de fréquences de Mel MFCC : 10

2.5 Extractions des caractéristiques du son : 10

2.5.1 Zero Crossing Rate 11

2.5.2 Centroïde spectral 11

2.5.3 Spectral Rolloff 11

2.5.4 Les fréquences chroma 12
2.6 Conclusion 13

3 Apprentissage automatique 14
3.1 Introduction au Apprentissage automatique 14
3.2 Types Machine Learning 14
3.3 Algorithmes d'apprentissage automatiques 15
3.3.1 Support Vector Machine SVM 15
3.3.2 K Nearest Neighbors KNN 16
3.4 L'apprentissage profond 17
3.5 Neurone artificiel 17
3.6 Algorithmes de Deep Learning 18
3.6.1 Modèle 1 : MultiLayer Perceptron 18
3.6.2 Modèle 2 : Réseau de neurones à convolution CNN 19
3.7 Le Machine Learning trditionnel contre Le Deep Learning 20
3.8 Conclusion 20

4 Réalisation 21
4.1 Preparation du Dataset 21
4.1.1 Description 21
4.1.2 Prétraitement du dataset pour le deep Learning 22
4.2 Outils de travail 24
4.2.1 Python 24
4.2.2 TensorFlow 24
4.2.3 Google Colab 25
4.3 Résultats des Modèles 25
4.3.1 Modèle 1 : KNN 25
4.3.2 Modèle 2 : SVM 27
4.3.3 Modèle 3 : MultiLayer Perceptron 28
4.3.4 Modèle 2 : Réseau de neurones à convolution CNN 30
4.4 Déploiement du modèle CNN 31

4.4.1 Flask 31
4.5 Architecture 31
4.5.1 Étapes du pipeline 32

Table des figures

1.1 musicip logo 3
1.2 Clementine logo 3
1.3 musicBee logo 4

2.1 forme ondulatoire 6
2.2 fréquence 7
2.3 Amplitude 7
2.4 CAN 8
2.5 Transformée de Fourier 8
2.6 Développement en série de Fourier 9
2.7 Spectogramme STFT 9
2.8 Spectogramme MFCC 10
2.9 Le centroïde spectral d'un morceaux qui se situe à la fin 11
2.10 les fréquences chroma 12
2.11 Étapes de traitement du signal 13
2.12 Overview 13

3.1 Division de l'espace par SVM 15
3.2 Algorithme knn 16
3.3 réseau de neurones 17
3.4 Neurone artificiel 18
3.5 Modèle CNN 19

4.1 exemple des spectogramme de Mel pour les classes musicales 22

4.2 Librosa Logo 22
4.3 python logo 24
4.4 TensorFlow logo 24
4.5 google colab logo 25
4.6 Matrice de confusion KNN 26
4.7 Matrice de confusion KNN 26
4.8 Matrice de confusion SVM 27
4.9 résultats du modèle du MLP 28
4.10 résultats après résolution du overfitting 29
4.11 Matrice de confusion MLP 29
4.12 Résultats de CNN 30
4.13 Matrice de confusion CNN 30
4.14 Flask logo 31
4.15 Architecture client serveur 31
4.16 Étapes du pipeline 32
4.17 Interface d'accueil 33
4.18 interface de prédiction 34

Introduction Générale de IA

L'intelligence artificielle IA consiste à mettre en œuvre un certain nombre de techniques visant à permettre aux machines d'imiter une forme d'intelligence réelle. L'IA se retrouve implémentée dans un nombre grandissant de domaines d'application.

La notion voit le jour dans les années 1950 grâce au mathématicien Alan Turing. Dans son livre Computing Machinery and Intelligence, ce dernier soulève la question d'apporter aux machines une forme d'intelligence. Il décrit alors un test aujourd'hui connu sous le nom « **Test de Turing** » dans lequel un sujet interagit à l'aveugle avec un autre humain, puis avec une machine programmée pour formuler des réponses sensées. Si le sujet n'est pas capable de faire la différence, alors la machine a réussi le test et, selon l'auteur, peut véritablement être considérée comme « intelligente ».

De Google à Microsoft en passant par Apple, IBM ou Facebook, toutes les grandes entreprises dans le monde de l'informatique planchent aujourd'hui sur les problématiques de l'intelligence artificielle en tentant de l'appliquer à quelques domaines précis. Chacun a ainsi mis en place des réseaux de neurones artificiels constitués de serveurs et permettant de traiter de lourds calculs au sein de gigantesques bases de données.

La vision artificielle, par exemple, permet à la machine de déterminer précisément le contenu d'une image pour ensuite la classer automatiquement selon l'objet, la couleur ou le visage repéré.

Les algorithmes sont en mesure d'optimiser leurs calculs au fur et à mesure qu'ils effectuent des traitements. C'est ainsi que les filtres antispam deviennent de plus en plus efficaces au fur et à mesure que l'utilisateur identifie un message indésirable ou au contraire traite les faux-positifs.

Sans oublier la classification automatique des genres de musique qui donne aux plate-formes de streaming un très grand succès grâce à leurs systèmes de recommandation qui attire plus d'utilisateur.

Au fur et à mesure de l'évolution de ces travaux, l'intelligence artificielle passe du simple chabot générique à un système de gestion de fonds automatique en finance, une aide au diagnostic en médecine, une évaluation des risques dans le domaine des prêts bancaires ou des assurances ou encore un allié décisionnel sur le terrain militaire.

Chapitre 1

Contexte général du projet

Nous allons à travers ce chapitre discuter la problématique et la motivation ainsi qu'une étude de l'existant .

1.1 Introduction

Les informations audio constituent une source importante de la perception de l'Homme, de ce fait , la simulation de cette perception par les machines peut avoir plusieurs utilisations pour son bien-être . Et parmi ces utilisation on peut citer :

— La reconnaissance vocale
— La classification des sons environnementaux
— La classification de genres musicaux cette dernière sera le but de notre projet

1.2 Problématique et motivation

Actuellement, la classification des genres est effectuée manuellement par les humains en appliquant leur compréhension personnelle de la musique. Cette tâche n'a pas encore été automatisée par les approches algorithmiques conventionnelles car les distinctions entre les genres musicaux sont relativement subjectives et mal définies. Cependant, l'ambiguïté de la classification des genres rend l'intelligence artificielle bien adaptée à cette tâche. Étant donné suffisamment de données audio, dont de grandes quantités peuvent être facilement récoltées à partir de musique librement disponible en ligne, l'apprentissage automatique peut observer et faire des prédictions en utilisant ces modèles mal définis. Le but de ce projet est de construire un classificateur de genres musicaux en utilisant une approche d'apprentissage profond qui peut correctement prédire le genre avec un niveau de confiance .

1.3 Etude de l'existant

1.3.1 Music IP Mixer

FIGURE 1.1 – musicip logo

L'application se base sur différents critères pour générer des listes de lecture dont les morceaux sont du même genre (comme l'artiste, le genre musicale...). L'application classera les morceaux par genre, artiste et album, on peut créer son propre mix ou bien générer un mix aléatoire pour en fin de compte créer une ou plusieurs listes de lecture. L'application se base sur les descriptions et les mots clé fournies avec les morceaux et ne procède pas à l'analyse du signal audio .

1.3.2 Clementine

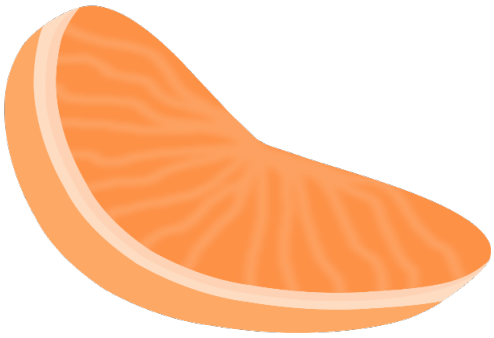


FIGURE 1.2 – Clementine logo

Clementine est un gestionnaire et un lecteur de musique open source entièrement gratuit. ce logiciel propose énormément d'options. Il fonctionne sur le principe d'une liste de lecture, qui s'enrichit au fur et à mesure que l'utilisateur sélectionne des pistes, des artistes et des albums. Clementine pense aux utilisateurs qui préfèrent naviguer dans l'arborescence de leurs dossiers pour sélectionner les fichiers manuellement, en intégrant un explorateur directe- ment dans la fenêtre du logiciel. Le gestionnaire de pochettes analyse les dossiers pour repérer les illustrations manquantes et les télécharge automatiquement. Elle est basé sur l'expérience utilisateur et les morceaux les plus écouté pour la classification

1.3.3 MusicBee

FIGURE 1.3 – musicBee logo

MusicBee est un lecteur audio gratuit et indépendant sorti en 2008 mis à jour régulièrement. À l'initialisation, il faut lui préciser le répertoire de stockage de la musique, qu'il scanne rapidement. Sa fonction de tag automatique et son gestionnaire de pochettes permettent de réorganiser efficacement une bibliothèque musicale désordonnée. Elle est rapide est légère et gratuite aussi elle se base sur les mots clés des tags des morceaux et les statistiques d'écoute d'un morceau donné .

1.4 Revue de la littérature :

Depuis le courant siècle beaucoup de recherches scientifiques ont eu lieu et qui ont tous pour objectifs majeurs qui est l'exploration des différents modèles et méthodes pour la classification des genres musicaux et les comparer en termes d'efficacité ainsi qu'en termes de précision.

L'article le plus célèbre est celui de Tzanetakis et Cook (2002) [1] qui ont abordé ce problème avec des approches de Machine Learning supervisé telles que le modèle des mélanges de Gaussiennes et les classificateurs KNN (K nearest neighbor) . Ils ont comparé les différents modèle en termes d'efficacité et ainsi ils ont extrait trois ensembles de propriété utiles pour cette tâche catégorisées comme structure timbrale, contenu rythmique et contenu du pitch .

Dans l'article de Scaringella et Zoia,2005 [2] ils ont exploré des modèles largement utilisés pour la reconnaissance vocale et ils les ont utilisés pour la classification musicale qui sont les modèles de Markov (Hidden Markov Models : HMMs).

Les auteurs discutaient l'apport des caractéristiques psycho-acoustiques pour la reconnaissance du genre musical, en particulier de l'importance du STFT et les coefficients cepstraux à fréquence de Mel (MFCC), qui a été parmi les caractéristiques Les plus importantes dans le premier article de Tzanetakis et Cook [1].

Avec le succès récent des réseaux de neurones profonds, un certain nombre d'études

appliquent ces techniques à la parole et à d'autres formes de données audio , pour alimenter les réseaux de neurones ; le problème posé était la représentation des fichiers audio en termes de fréquences et de temps , chose qui a été premièrement abordée par Van Den Oordet al. (2016) [3] , mais une représentation alternative commune a vu le jour qui revient à transformer le fichier en son spectogramme et en constituer un réseau de neurones à convolution Wyse 2017 [4] Et finalement dans l'article de l'auteur a fait une comparaison entre le deep learning utilisant CNNs et les méthodes traditionnelles du machines learning Hareesh Bahuleyan 2018 [5]

1.5 Conclusion

Après ce tour d'horizon sur les différentes applications qui existent dans le marché pour la classification des genres musicaux pour utilisation personnelle , ainsi que les recherches scientifiques qui étaient menées dans ce domaines , on a découvert un besoin d'une application qui doit implémenter les méthodes du Machine Leaning décrites et bien étudiées et comparées dans les articles scientifiques , qui sont utilisées dans les applications de streaming pour des utilisations internes de ces applications, et qui sont malheureusement peu utilisées pour les applications éxistants dans le marché qui propose une utilisation personnelle de la classification automatique des morceaux de musique . Donc il s'est avéré nécessaire la création d'une application qui à la fois utilisable pour le besoin de l'utilisateur d'un coté et qui implémente les meilleurs méthodes connues de machines Learning pour des résultats acceptables et satisfaisantes .

Chapitre 2

Théorie du traitement du signal Audio

Nous allons à travers ce chapitre etudier le son et extraire les caractéristiques qui nous seront utiles pour l'analyse

2.1 Definition

Le son est une manifestation des vibrations physiques microscopiques des molécules qui nous entourent , ces vibrations cause une perturbation dans la pression de l'air et ce qui produit des ondes de pression

2.1.1 Forme Ondulatoire

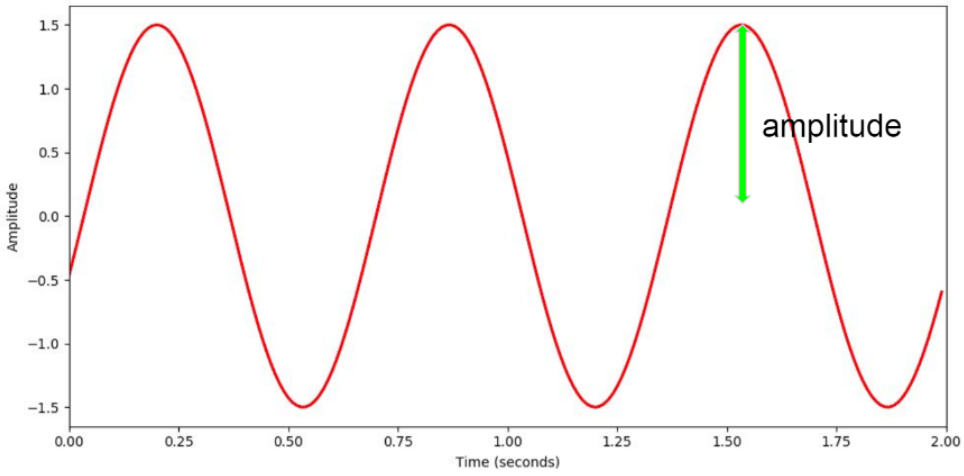


FIGURE 2.1 – forme ondulatoire

Le son est comme toute autre ondes dans sa forme la plus simple a une amplitude , une période et une fréquences et peut être représenté sous la forme d'une fonction sinusoïdale .

2.1.2 Propriétés : fréquence , amplitude :

La fréquence peut se définir simplement par le nombre de périodes par seconde , sa manifestations pour l'être humain est le caractère de l'aiguité ou le pitch. L'amplitude est un caractère qui décrit , le volume du son , un son qui a une grande amplitude et plus détecté par les instruments d'observation du son .

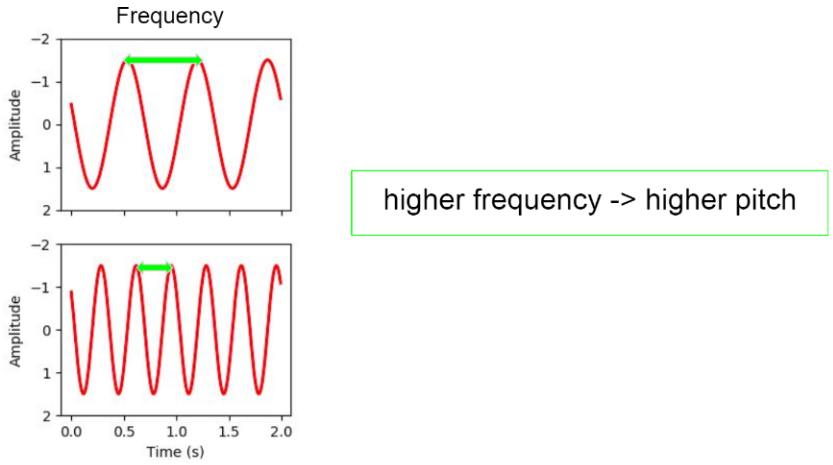


FIGURE 2.2 – fréquence

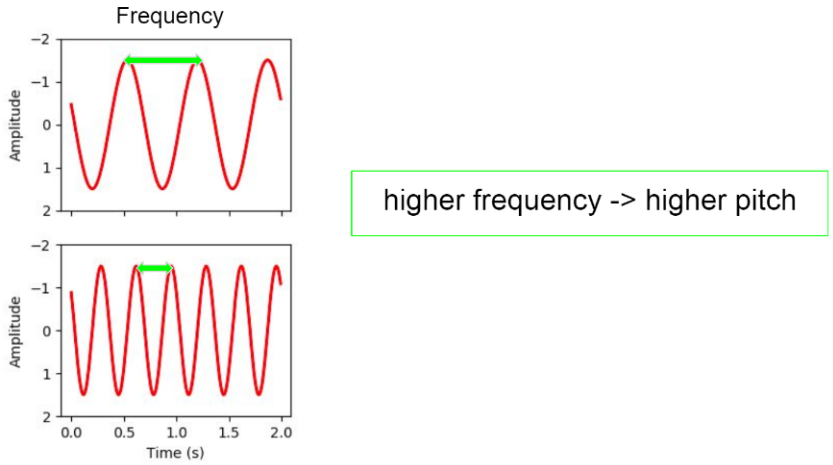


FIGURE 2.3 – Amplitude

2.1.3 Conversion Analogique Numérique :

Comme tous les signaux analogiques le son doit être transformé en un signal numérique pour qu'il soit traité par la machine , cette numérisation se passe par un échantillonage uniforme du signal par une fréquence d'échantillonnage «Sample rate » ainsi qu'une quantification d'amplitude par un nombre limité de bits.

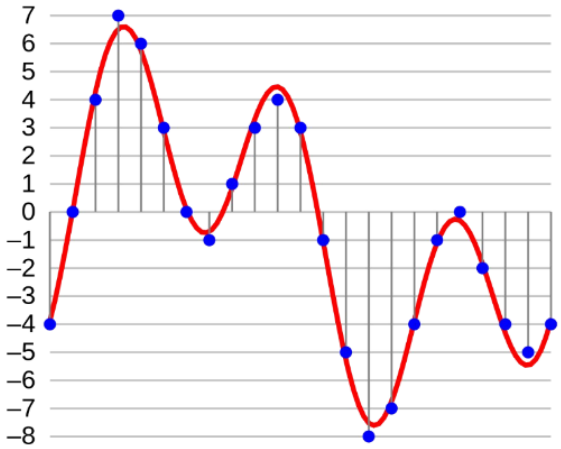


FIGURE 2.4 – CAN

2.2 Transformée de Fourier :

Un signal musical est une forme très complexe au niveau ondulatoire , car il présente une superposition de plusieurs signaux sinusoïdaux simple comme l'affirme le théorème de Fourier , l'utilisation de la transformée de Fourier simplifie beaucoup le traitement du n'importe quel signal car il utilise une transformation du domaine du temps qui est compliqué vers le domaine des fréquences qui est plus simple.

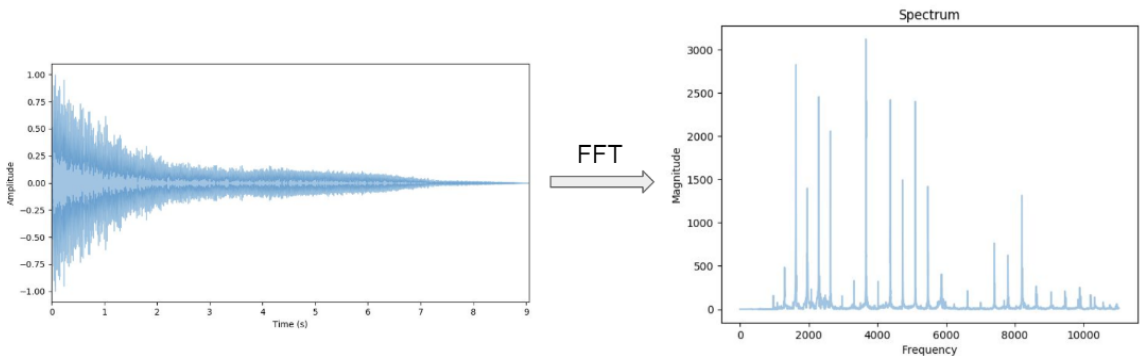
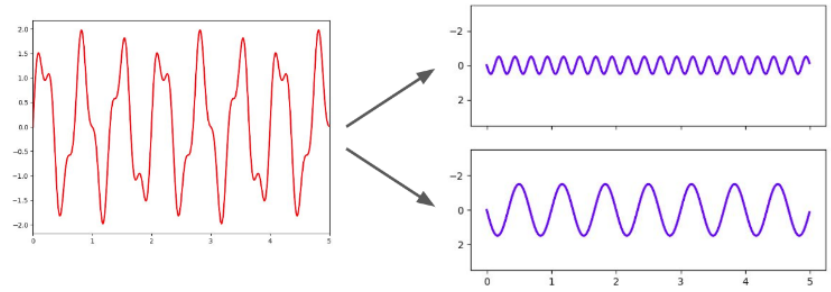


FIGURE 2.5 – Transformée de Fourier



$$s = A_1 \sin(2\pi f_1 t + \varphi_1) + A_2 \sin(2\pi f_2 t + \varphi_2)$$

FIGURE 2.6 – Développement en série de Fourier

Limites de la transformée de Fourier ! ! :
La transformée de Fourier nous donnes une images générale et statique du son qui varient rapidement dans le temps ce qui ne donne pas des résultats d'analyses exactes, d'où la nécessité d'une autre transformation plus raffinée .

2.3 Short time Fourier transform STFT :

Cette méthode consiste à diviser le signal en plusieurs échantillons et d'appliquer la transformée de Fourier pour chaque échantillon , ce qui protège le caractère du temps ainsi que il donne un spéctogramme presque utilisable comme input pour les algorithmes d'apprentissage :

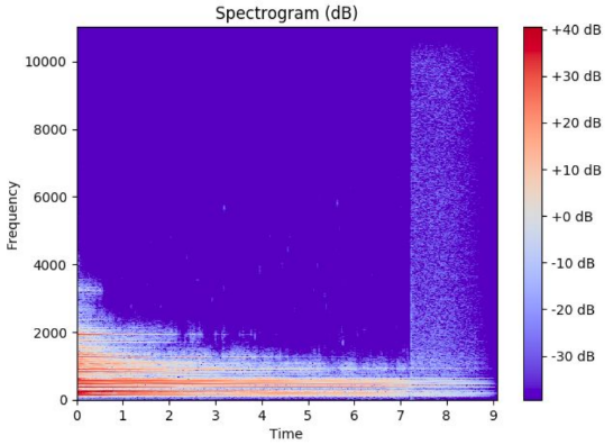


FIGURE 2.7 – Spectogramme STFT

Limites du STFT :
Cette transformée donne un input utilisable pour la machine mais ne simule pas la perception humaine du son qui fait la différence entre des marges de fréquences mais aussi regroupe plusieurs fréquences dans une même sensation et perceptions d'où la nécessité d'une représentation qui prend en compte l'aspect sensation humaines

2.4 L'Échelle de fréquences de Mel MFCC :

C'est une échelle logarithmique qui fusionne des bandes de fréquences dans une seule valeur et qui est le plus proche du système auditoire humain

$$m = 2595log_{10}(1 + \frac{f}{700}) = 1127ln(1 + \frac{f}{700})$$

Cette formule est la plus utilisée et elle est même utilisée dans les recherches citée plus haut . Après la prise en compte de cet aspect on peut finalement avoir un spectogramme qui est le plus adapté pour être un input pour notre analyse et qui distingue les pistes en prenant en considération la perception humaine

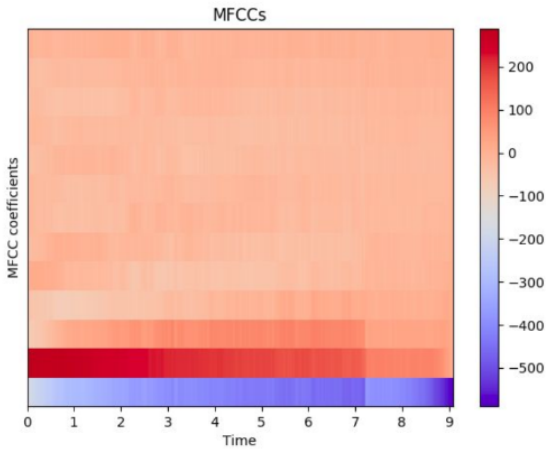


FIGURE 2.8 – Spectogramme MFCC

2.5 Extractions des caractéristiques du son :

En plus des MFCCS, chaque signal audio comprend de nombreuses caractéristiques. Cependant, nous devons extraire les caractéristiques pertinentes au problème que nous

essayons de résoudre. Le processus d'extraction des caractéristiques pour les utiliser à des fins d'analyse est appelé extraction des caractéristiques. Étudions en détail quelques-unes des fonctionnalités.

2.5.1 Zero Crossing Rate

Le taux de passage par zéro est le taux de changements de signe le long d'un signal, c'est-à-dire le taux auquel le signal passe du positif au négatif ou inversement. Cette fonctionnalité a été largement utilisée à la fois dans la reconnaissance vocale et la récupération d'informations musicales. Il a généralement des valeurs plus élevées pour les sons hautement percutants comme ceux du métal et du rock.

2.5.2 Centroïde spectral

Il indique où se situe le «centre de masse» d'un son et est calculé comme la moyenne pondérée des fréquences présentes dans le son. Considérons deux chansons, l'une d'un genre blues et l'autre appartenant au métal. Maintenant, par rapport à la chanson de genre blues qui est la même sur toute sa longueur, la chanson métal a plus de fréquences vers la fin. Ainsi, le centroïde spectral pour la chanson de blues se situera quelque part près du milieu de son spectre tandis que celui d'une chanson de métal serait vers sa fin.

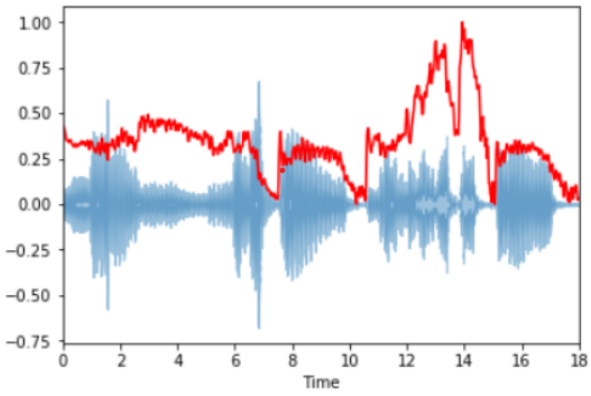


FIGURE 2.9 – Le centroïde spectral d'un morceaux qui se situe à la fin

2.5.3 Spectral Rolloff

C'est une mesure de la forme du signal. Il représente la fréquence en dessous de laquelle un pourcentage spécifié de l'énergie spectrale totale, par ex. 85

2.5.4 Les fréquences chroma

Les caractéristiques de chrominance sont une représentation intéressante et puissante pour la musique audio dans laquelle tout le spectre est projeté sur 12 cases représentant les 12 demi-tons distincts (ou chrominance) de l'octave musicale.

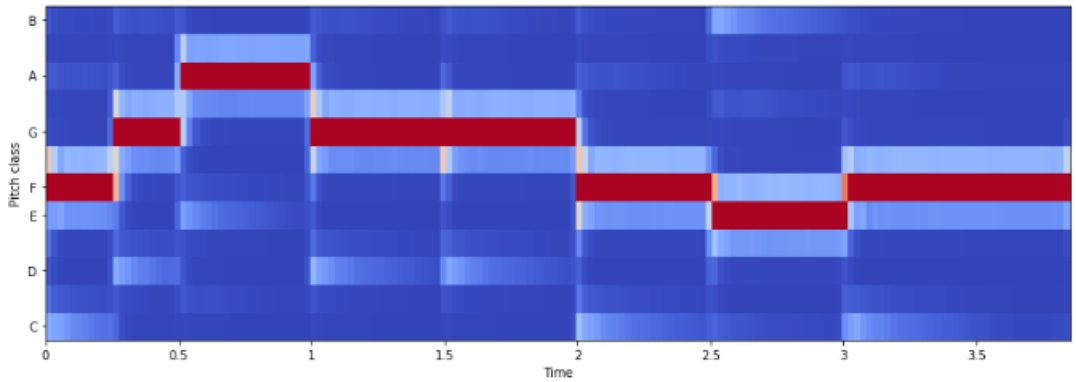


FIGURE 2.10 – les fréquences chroma

2.6 Conclusion

Dans cette section on a présenté une vue théorique du son ainsi que le pré-traitement nécessaire pour le rendre utilisable pour notre projet , Le pré-traitement nécessaire se résume comme suit :

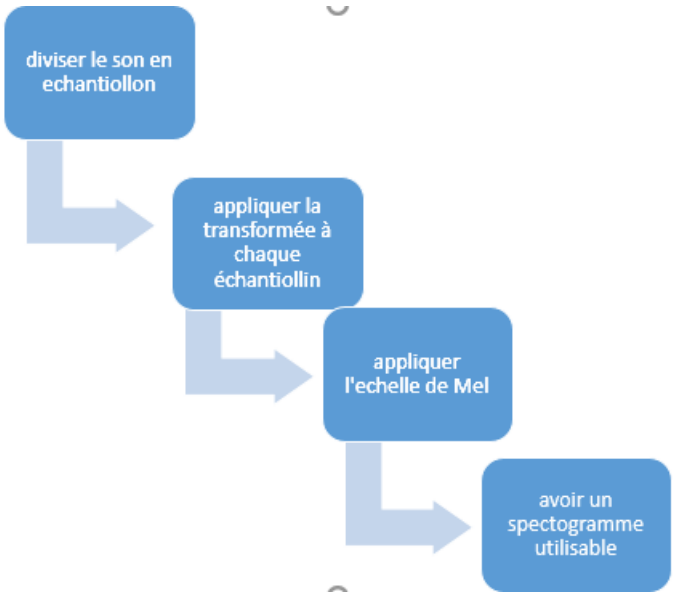


FIGURE 2.11 – Étapes de traitement du signal

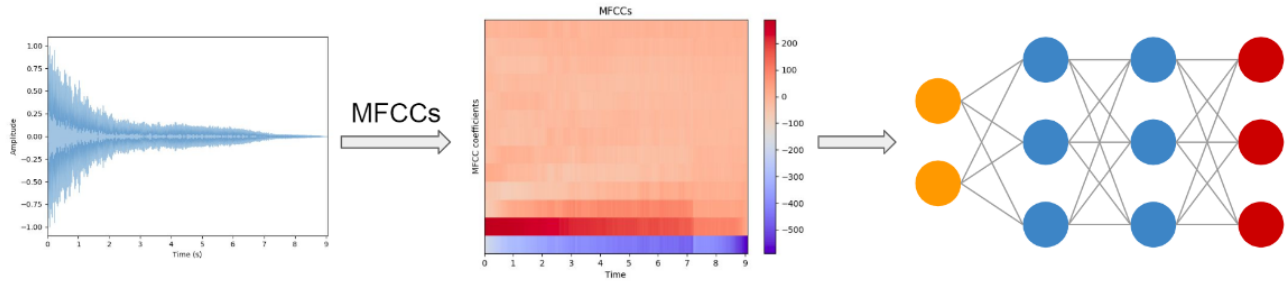


FIGURE 2.12 – Overview

Chapitre 3

Apprentissage automatique

Nous allons à travers ce chapitre explorer le monde de l'apprentissage automatique en présentant sa théorie

3.1 Introduction au Apprentissage automatique

L'apprentissage automatique d'après La définition La plus connue et répandue est la capabilité de la machine (l'ordinateur) à apprendre sans être explicitement programmée avec des règles bien définies en contraire des systèmes experts qui sont créés avec des règles très bien déterminées par le programmeur , il utilise des algorithmes qui se reposent soit sur des modèles statistiques comme la régression linéaire , la régression logistique , KNN , SVM ... , soit sur les réseaux de neurones qui constitue l'apprentissage profond ou le deep Learning .

3.2 Types Machine Learning

Le Machine Learning dans sa définition est un très large domaine mais il peut globalement être réparti en trois types fondamentaux :

- L'apprentissage supervisé : c'est le types le plus utilisés dans la vie réel , il est prescrit généralement pour les problèmes de classification et régression , et nécessite un dataset étiqueté
- L'apprentissage non supervisé il est utilisé là où les données ne sont pas étiquetées. Il s'agit donc de découvrir les structures sous-jacentes à ces données non étiquetées.
- L'apprentissage par renforcement : consiste, pour un agent autonome à apprendre les actions à prendre, à partir d'expériences, de façon à optimiser une récompense quantitative au cours du temps , il est le types d'apprentissage utilisé pour les agents

des jeux comme les échecs , Go ...

3.3 Algorithmes d'apprentissage automatiques

On présente deux algorithmes très puissants dans les problèmes de classification

3.3.1 Support Vector Machine SVM

Les machines à vecteurs de support (SVM) sont des modèles d'apprentissage supervisé avec des algorithmes d'apprentissage associés destinées à résoudre des problèmes de discrimination et de régression.

Un modèle SVM est une représentation des données sous forme de points dans l'espace, divisés de manière à ce que les données des catégories distinctes soient divisés par un espace aussi clair que possible. De nouvelles données sont ensuite ajoutés dans ce même espace et devraient appartenir à une catégorie en fonction du côté de l'écart sur lequel elles se trouvent.

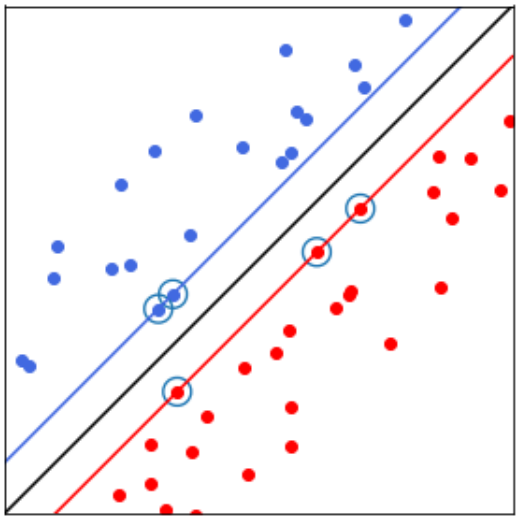


FIGURE 3.1 – Division de l'espace par SVM

SVM dans python

Dans Scikit-learn, les SVM sont implémentées dans le module "sklearn.svm". Scikit-Learn contient la bibliothèque svm, qui contient des classes intégrées pour différents algorithmes SVM.

3.3.2 K Nearest Neighbors KNN

La méthode des K plus proches voisins (KNN) a pour but de classifier des points cibles (classe méconnue) en fonction de leurs distances par rapport à des points constituant un échantillon d'apprentissage (c'est-à-dire dont la classe est connue a priori).

KNN est une approche de classification supervisée intuitive, souvent utilisée dans le cadre du machine learning. Il s'agit d'une généralisation de la méthode du voisin le plus proche (NN). NN est un cas particulier de KNN, où k = 1.

L'approche de classification KNN se base sur l'hypothèse que chaque cas de l'échantillon d'apprentissage est un vecteur aléatoire. Chaque point est décrit comme x =< a1(x), a2(x), a3(x),.., an(x) > où ar(x) correspond à la valeur I du rème attribut. ar(x) peut être soit une variable quantitative soit une variable qualitative.

Afin de déterminer la classe d'un point cible, chaque chacun des k points les plus proches de xq procèdent à un vote. La classe de xq correspond à la classe majoritaire.

Pour les données quantitatives on utilise les distances disponibles(métriques) : Euclidienne, Minkowski, Manhatan, Tchebychev, Canberra

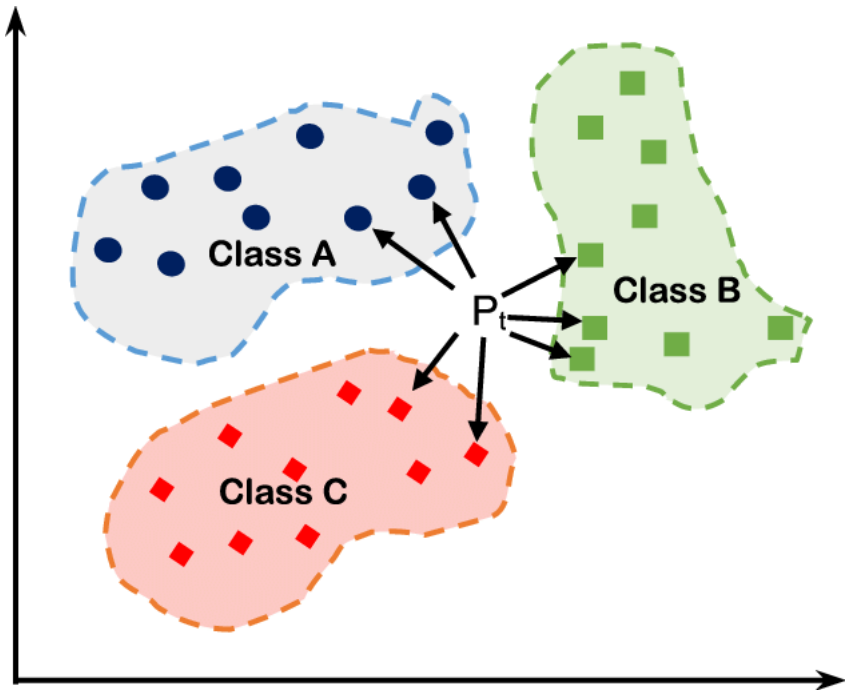


FIGURE 3.2 – Algorithme knn

knn dans python

Dans Scikit-learn, les knn sont implémentées dans le module "sklearn.neighbors". Scikit-Learn contient la bibliothèque KNeighborsClassifier, qui contient des classes intégrées pour différents algorithmes knn.

3.4 L'apprentissage profond

Le deep learning est un sous domaine du machine learning qui se base sur les réseaux de neurones artificiels , .ce dernier est un ensemble d'algorithmes qui utilisent des neurones artificiels qui simulent les neurones biologiques , ces neurones sont réparties en multiples couches en commençant par la couche des input et terminant par la couche des output en passant par des couches internes qui met les inputs initiales en des transformations à l'aides des calculs mathématiques . Il est notamment utilisé pour des problèmes complexes et avec des données très massive.

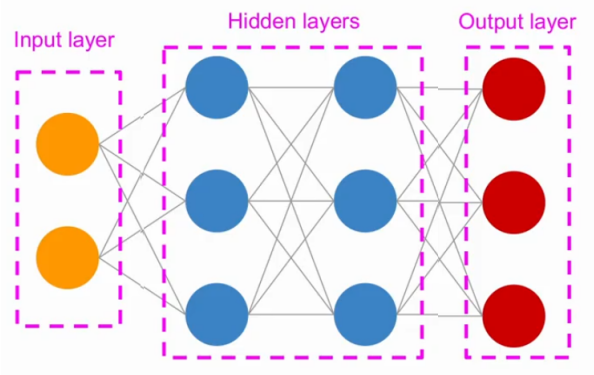


FIGURE 3.3 – réseau de neurones

3.5 Neurone artificiel

Les neurones artificiels sont les unités de calculs de base des réseaux de neurones. Un neurone est tout simplement une unité qui prend en paramètre des inputs en appliquant à chacun un poids , calcule leur somme et la passe à une fonction dite d'activation et retourne l'output , qui peut être l'entrée d'un autre neurone. Le but général de l'entraînement d'un réseau de neurones est avoir les poids optimums pour des pertes minimales à la sortie .

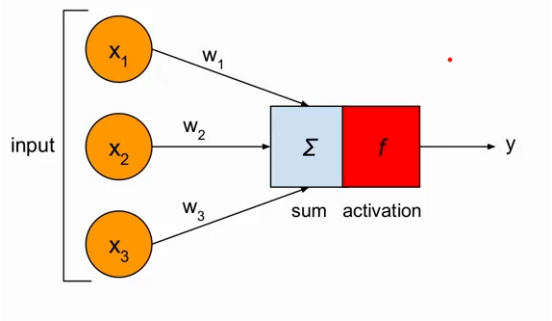


FIGURE 3.4 – Neurone artificiel

3.6 Algorithmes de Deep Learning

On présente les deux algorithmes les plus utilisée pour la classification en Deep Learning

3.6.1 Modèle 1 : MultiLayer Perceptron

Description du Modèle

Le modèle le plus communément utilisé des réseaux de neurones est le multiLayer perceptron qui est constitué de plusieurs couches successives. Ce réseau est toujours totalement connecté c'est à dire que les différents neurones de chaque couche sont connectés à tous les neurones des couches adjacentes . Pour notre modèle on a utilisé un feed forward network qui est composé en addition de la couche d'entrée et la couche de sortie de trois couches internes totalement connectées avec une fonction d'activation ReLU et avec la fonction de perte cross entropy loss . L'entrée dans notre modèle était 1D lors de l'utilisation des spectrogrammes MFCC , nous avons aplati les données, À chaque couche, nous avons appliqué une activation de la fonction ReLU à la sortie de chaque nœud, en suivant la formule :

$$ReLU(x) = \begin{cases} x & x >= 0 \\ 0 & x < 0 \end{cases}$$

À la fin, nous construisons une distribution de probabilité des 10 genres en exécutant les sorties via une fonction softmax

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

Pour optimiser notre modèle, nous avons minimisé la perte cross entropy loss :

$$CE(\theta) = -\sum_{x \epsilon X} y(x) log(\hat{y}(x))$$

3.6.2 Modèle 2 : Réseau de neurones à convolution CNN

Description

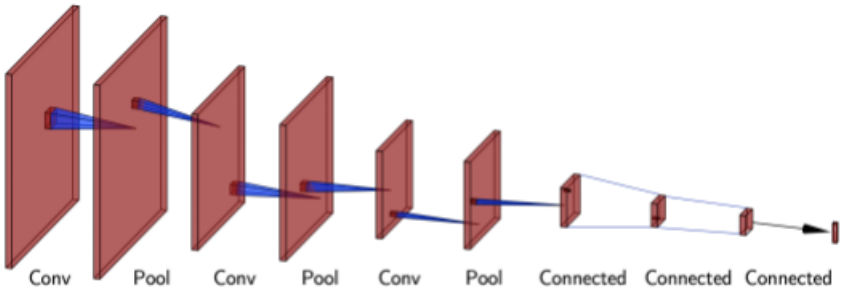


FIGURE 3.5 – Modèle CNN

Les spectrogrammes peuvent être considérés comme «images» et fournies en entrée à un CNN, qui a montré de bonnes performances sur les tâches de classification d'images. Chaque bloc d'un CNN se compose de ces trois opérations suivantes :

- **Convolution** : Cette étape consiste à faire glisser un filtre matriciel (disons taille 3x3) sur l'image d'entrée qui est de (largeur d'image) x (hauteur de l'image). Le filtre est d'abord placé sur la matrice d'image, puis nous calculons une multiplication par élément entre le filtre et la partie de chevauchement de l'image, suivi d'une sommation pour donner une valeur de propriété. Nous utilisons beaucoup de ces filtres, les valeurs sont «apprises» lors de l'entraînement du réseau de neurones via rétro-propagation.
- **Pooling** :Il s'agit d'un moyen de réduire la dimension de la carte d'entités obtenue à partir de l'étape de convolution. Par exemple , par Max Pooling avec une taille de fenêtre 2x2, nous ne retenons que l'élément avec la valeur maximale parmi les 4 éléments de la carte d'entités qui sont couvert par cette fenêtre. Nous continuons de bouger fenêtre sur la carte d'entités avec un pas prédéfinie.
- **Activation Non linéaire** L'opération de convolution est linéaire et afin de rendre le réseau de neurones plus puissant, nous devons introduire une certaine non-linéarité. Dans ce but, nous pouvons appliquer une fonction d'activation telle que ReLU sur chaque élément de la carte.

3.7 Le Machine Learning trditionnel contre Le Deep Learning

Le deep Learning est un sous domaine qui est né à cause de l'insuffisance et la non adéquation du machine learning traditionnel avec des problèmes réels , car avec la croissance exponentielle de la quantité des données ; les méthodes traditionnelles se sont trouvées insuffisantes , et ce pour des multiples raison :

- Les méthodes traditionnelles nécessitent un prétraitement énorme pour l'extraction des propriétés selon lesquelles la machine prends les décision alors que le deep learning se préoccupe du tout et ne nécessite pas ce prétraitement ce qui le rend adéquat pour le traitement des images et du son.
- Les méthodes traditionnelles ne peuvent pas travailler avec des quantités énorme de données.
- Les méthodes traditionnelles sont parfaites pour les problèmes simples .

3.8 Conclusion

Dans cette section on a présenté une vue générale du domaine du machine learning , en explorant ses différents types ainsi que son sous domaine le deep learning , ensuite on a fait une comparaison entre le deep learning et le machine learning classique pour savoir qui est le plus adéquat à notre projet , on a conclus en s'appuyant sur des recherches scientifiques que le deep learning a fait preuve de bonnes performances surtout dans les domaines de la classification et le traitement du son ainsi que des images.

Chapitre 4

Réalisation

Dans ce chapitre nous présentons les différents outils de travail ainsi que l'environnement et finalement la réalisation de notre projet.

4.1 Preparation du Dataset

4.1.1 Description

Pour la première recherche scientifique décrite dans 1.4. [1], ils ont utilisé un data-set qui continue à grandir jusqu'à aujourd'hui et qui est connue pour sa qualité ainsi que la précision d'étiquetage, elle est connue sous le nom GTZAN et elle est ouverte gratuitement. Ce data set nous fournit 1000 clips audio de 30 secondes , tous étiquetés par un des 10 genres possibles et présentés sous forme de fichiers .au La classification est malgré nous soumise à la subjectivité de l'être humain et il y a des morceaux que chacun d'entre nous classifiera différemment d'où la nécessité d'une normalisation de cette classification et une limitations du nombre des classes pour des résultats cohérents , ainsi ,les différents classes de musiques qu'on utilisera sont comme décrit dans le dataset :

- Classical
- Country
- Disco
- HipHop
- Jazz
- Rock
- Blues
- Reggae
- Pop
- Metal

Les sectogrammes suivant montres les differents figures des genres musicaux representées à l'echelle du Mel.

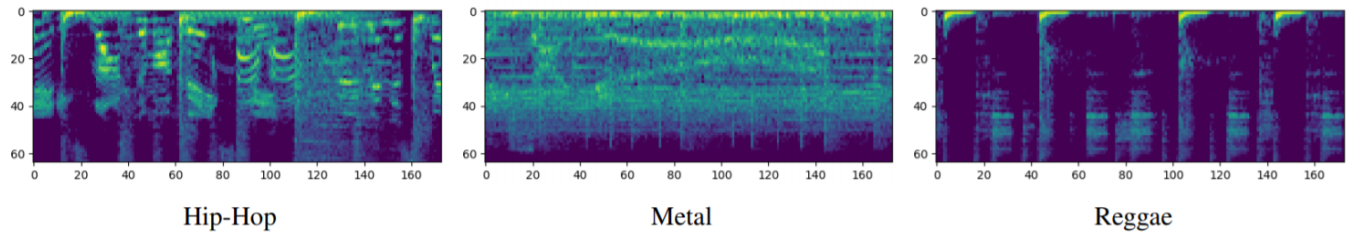


FIGURE 4.1 – exemple des spectogramme de Mel pour les classes musicales

4.1.2 Prétraitement du dataset pour le deep Learning

Avant de travailler avec les clips audio il est nécessaire de les transformer en une forme utilisable par notre algorithme , pour celà on transforme le dataset à un fichier json contenant les informations utiles pour chaque piste.

Bibliothèque utiliseé : Librosa

FIGURE 4.2 – Librosa Logo

Librosa est une bibliothèque Open source largement utilisée dans le traitement du son en python, elle propose beaucoup de fonctionnalités afin de faciliter le traitement du son comme le calcul du MFCC , les spectogramme des frequences ainsi que les spectogramme de Mel.

Structure du data Généré après extraction des caractéristiques

On a utilisé Librosa pour extraire les caractéristiques cités dans 2.5 pour généré un fichier csv contenant pour chacun des morceaux la moyenne et la variance de chaque ca-

ractéristique extraite

	meanZCR	stdZCR	meanSpecCentroid	stdSpecCentroid	meanSpecContrast	stdSpecContrast	meanSpecBandwidth	stdSpecBandwidth
0	0.083045	0.027694	1784.165850	360.241675	20.526699	8.760242	2002.449060	293.057608
1	0.056040	0.038046	1530.176679	613.066125	20.676128	8.237203	2039.036516	462.432433
2	0.076291	0.031731	1552.811865	395.559911	22.197517	8.725292	1747.702312	276.141616
3	0.033309	0.020561	1070.106615	429.366909	21.426268	7.961446	1596.412872	407.972419
4	0.101461	0.044205	1835.004266	586.003361	21.466338	7.936384	1748.172116	297.397392

Structure du data Généré pour Deep Learning

Un algorithme du machine learning a besoin d'une grande quantité de data , notre dataset contient 10 genres avec 100 pistes par genres ce qui donne 1000 pistes au final ce qui n'est pas suffisamment grand.À partir de chaque clip, nous avons échantillonné une fenêtre contiguë de 2 secondes à quatre emplacements aléatoires, augmentant ainsi nos données à 10000 clips de deux secondes chacun. À l'issue de notre prétraitement on génére un fichier json contenant un seule objet , constitué de trois listes :

— Mapping : elle contient les genres musicaux , elle sert à donner à chaque genre un nombre utilisable par la suite
— Labels : c'est elle qui contient le genre musical de chaque morceau sous forme d'un nombre
— MFCC : cette dernière liste contient pour chaque morceau une liste contenant une liste des MFCCs par chaque intervalle du temps

```
{
  "mapping": [
    "nom_des_genres" * pour chaque genres
  ],
  "labels": [
    "numero du genre pour chaque piste" * nombre de morceaux
  ],
  "mfcc": [
        [
            [MFCC * nbr_mfccs]*nbre_de fenetre

        ]* nombre de morceaux
    ]
}
```

4.2 Outils de travail

Le machine learning est un domaine qui nécessite l'implémentation de beaucoup d'algorithmes très compliqués , d'où la nécessité des outils qui facilite cette implémentation et qui présentent beucoup de fonctionnalités du monde de l'intelligence artificielle.

4.2.1 Python

FIGURE 4.3 – python logo

Python est le langage le plus populaire dans le monde de l'intelligence artificielle. Python est orienté objet et se veut relativement facile d'accès. Il est très utilisé au sein de la communauté scientifique et particulièrement dans le domaine de l'intelligence artificielle. Les principaux frameworks de machine learning et deep learning sont effectivement disponibles en Python.

4.2.2 TensorFlow

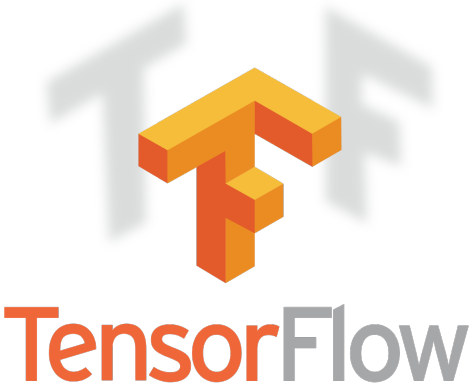


FIGURE 4.4 – TensorFlow logo

TensorFlow est une plateforme open source developpée par Google de bout en bout pour le machine learning. Il dispose d'un écosystème complet et flexible d'outils, de bibliothèques et de ressources communautaires qui permet aux chercheurs de faire évoluer l'état de l'art en machine learning et aux développeurs de créer et de déployer facilement des applications propulsées par ML, elle propose des librairies qui facilite la création des réseaux de neurones ainsi que leurs entrainements .

4.2.3 Google Colab

Le machine learning nécessite beaucoup des ressources que se soit en terme de mémoire ou de traitement ce qui rend nécessaire l'utilisation les plateformes du cloud pour l'entrainement des modèles.

FIGURE 4.5 – google colab logo

Google Colab ou Colaboratory est un service cloud, offert par Google gratuitement basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l'apprentissage automatique. Cette plateforme permet d'entraîner des modèles de Machine Learning directement dans le cloud , et facilite ainsi la collaboration de plusieurs membres.

4.3 Résultats des Modèles

4.3.1 Modèle 1 : KNN

Pour ce modèle on a essayé plusieurs valeurs pour le paramètre K et on a trouvé une précision maximale avec K=4

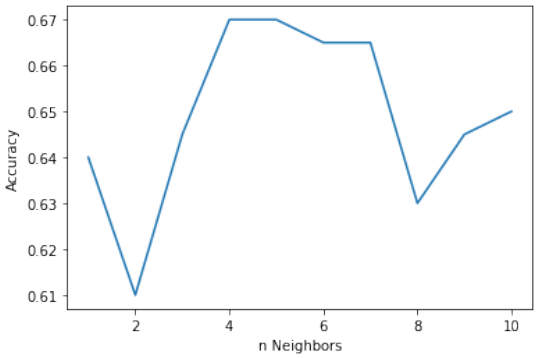


FIGURE 4.6 – Matrice de confusion KNN

Résultats

On a obtenu une précision de 0.78 pour les données d'entraînement mais pour les données de teste que le modèle n'a jamais vu avant on a obtenu une précision de 0.67 .

Matrice de confusion

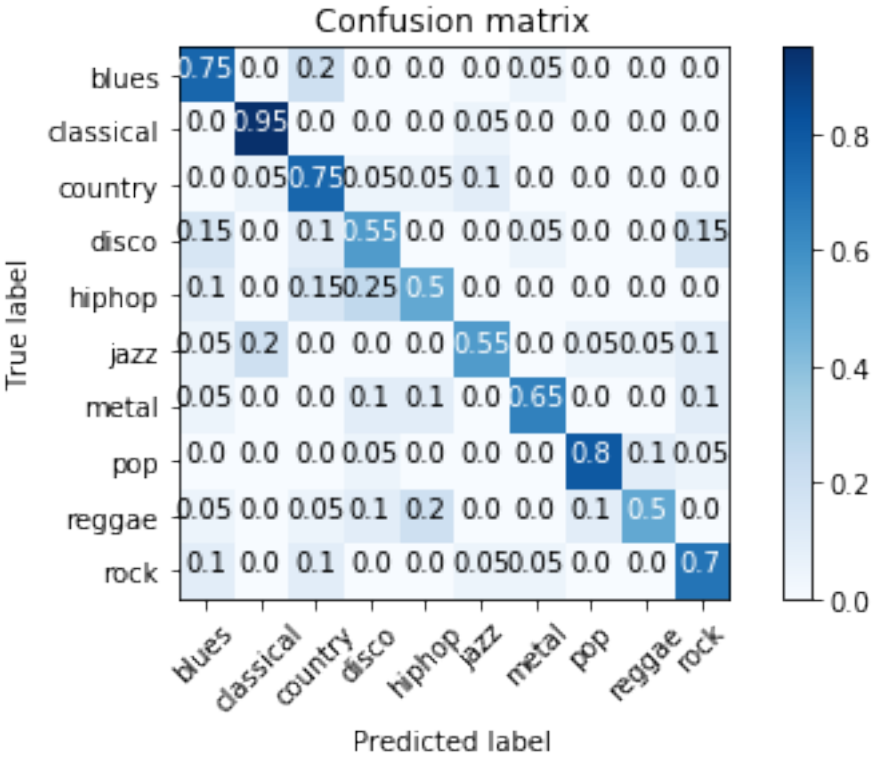


FIGURE 4.7 – Matrice de confusion KNN

4.3.2 Modèle 2 : SVM

Pour ce modèle on a utilisé le Kernel RBF (Radial Basis Function) qui a montré sa performances avec les données non séparables linéairement

Résultats

Pour le SVM on a réussi à avoir une précision de 0.7 pour le teste set avec 0.99 pour le traning set.

Matrice de confusion

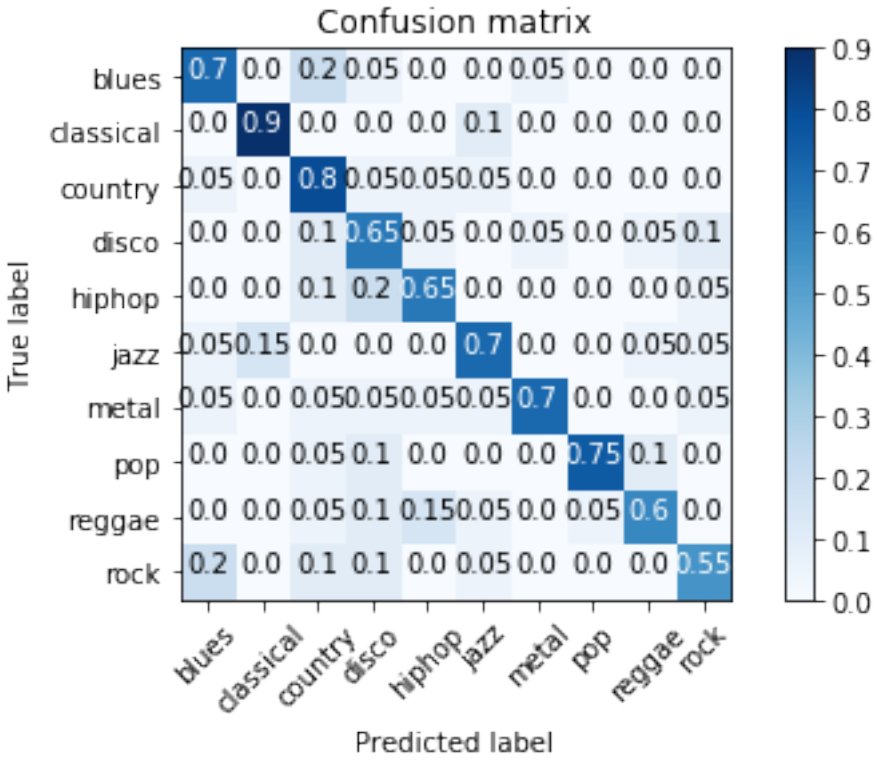


FIGURE 4.8 – Matrice de confusion SVM

4.3.3 Modèle 3 : MultiLayer Perceptron

Résultats


```
219/219 - 1s - loss: 0.0343 - accuracy: 0.9886
94/94 - 0s - loss: 3.1736 - accuracy: 0.6299

Test accuracy: 0.6298766136169434

Train accuracy: 0.988566517829895
```

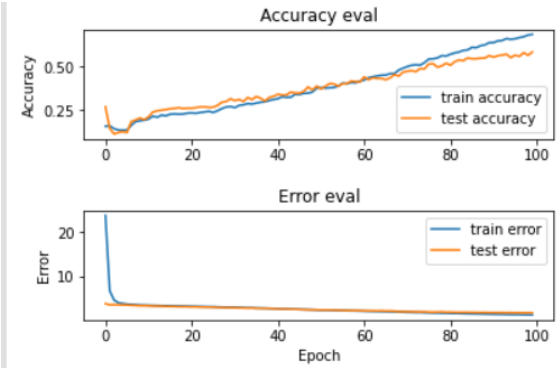
FIGURE 4.9 – résultats du modèle du MLP

On a obtenu une précision de 0.93 pour les données d'entraînement mais pour les données de teste que le modèle n'a jamais vu avant on a obtenu une précision de 0.58 , ce qui montre une grande différence entre les l'entraînement et le teste ce qui prouve qu'il y a le phénomène du Overfitting

Résolution du Overfitting

Les méthodes qu'on a utilisé pour résoudre l'Overfitting sont les deux méthodes les plus utilisées pour ce problémes qui sont :

- **DropOut** : qui consistete à enlever un neurone d'une couche avec une probabilité donnée .
- **Regularisation L2** : celle ci consiste à ajouter une pénalité à la fonction de perte sous la forme $\lambda \sum_i \omega i$ pour pénaliser les poids grands




```
219/219 - 1s - loss: 0.9225 - accuracy: 0.7879
94/94 - 0s - loss: 1.6557 - accuracy: 0.5832

Test accuracy: 0.583194375038147

Train accuracy: 0.7879090905189514
```

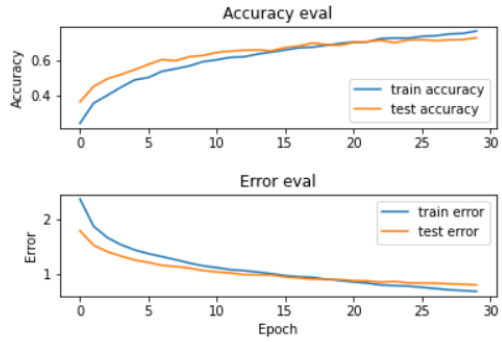
FIGURE 4.10 – résultats après résolution du overfitting

Matrice de confusion

FIGURE 4.11 – Matrice de confusion MLP

4.3.4 Modèle 2 : Réseau de neurones à convolution CNN

Résultats




```
79/79 - 1s - loss: 0.7904 - accuracy: 0.7275
188/188 - 3s - loss: 0.4746 - accuracy: 0.8464

Train accuracy: 0.8464232087135315

Test accuracy: 0.727491021156311
```

FIGURE 4.12 – Résultats de CNN

Matrice de confusion

FIGURE 4.13 – Matrice de confusion CNN

On a obtenu des résultats satisfaisants avec une précision de 0.73 qui est très bonne et sans phénomène d'Overfitting ce qui montre la puissance des CNN pour relever les motifs des images et des audio.

4.4 Déploiement du modèle CNN

Dans cette section on presentes les étapes nécessaires pour rendre le modèle utilisable dans une application.

4.4.1 Flask

FIGURE 4.14 – Flask logo

Flask est un framework open-source de développement web en Python. Son but principal est d'être léger, afin de garder la souplesse de la programmation Python, associé à un système de templates.

4.5 Architecture

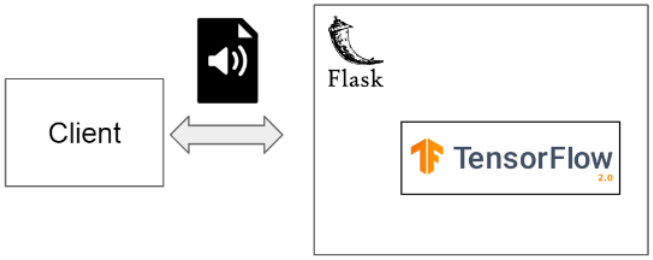


FIGURE 4.15 – Architecture client serveur

4.5.1 Étapes du pipeline

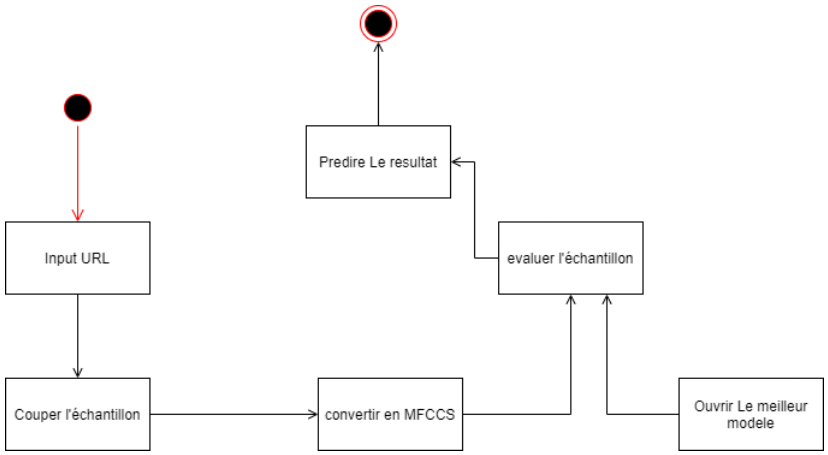


FIGURE 4.16 – Étapes du pipeline

1. L'utilisateur saisit l'URL de la chanson YouTube dans l'interface du programme et il télécharge la chanson.

2. Morceau coupé à un échantillon et converti en tableau numpy

3. Le son est normalisé puis converti en MFCC

4. Le meilleur modèle est chargé, puis le niveau de confiance prévu de chaque genre est calculé et affiché

Capture d'écran de l'interface d'accueil

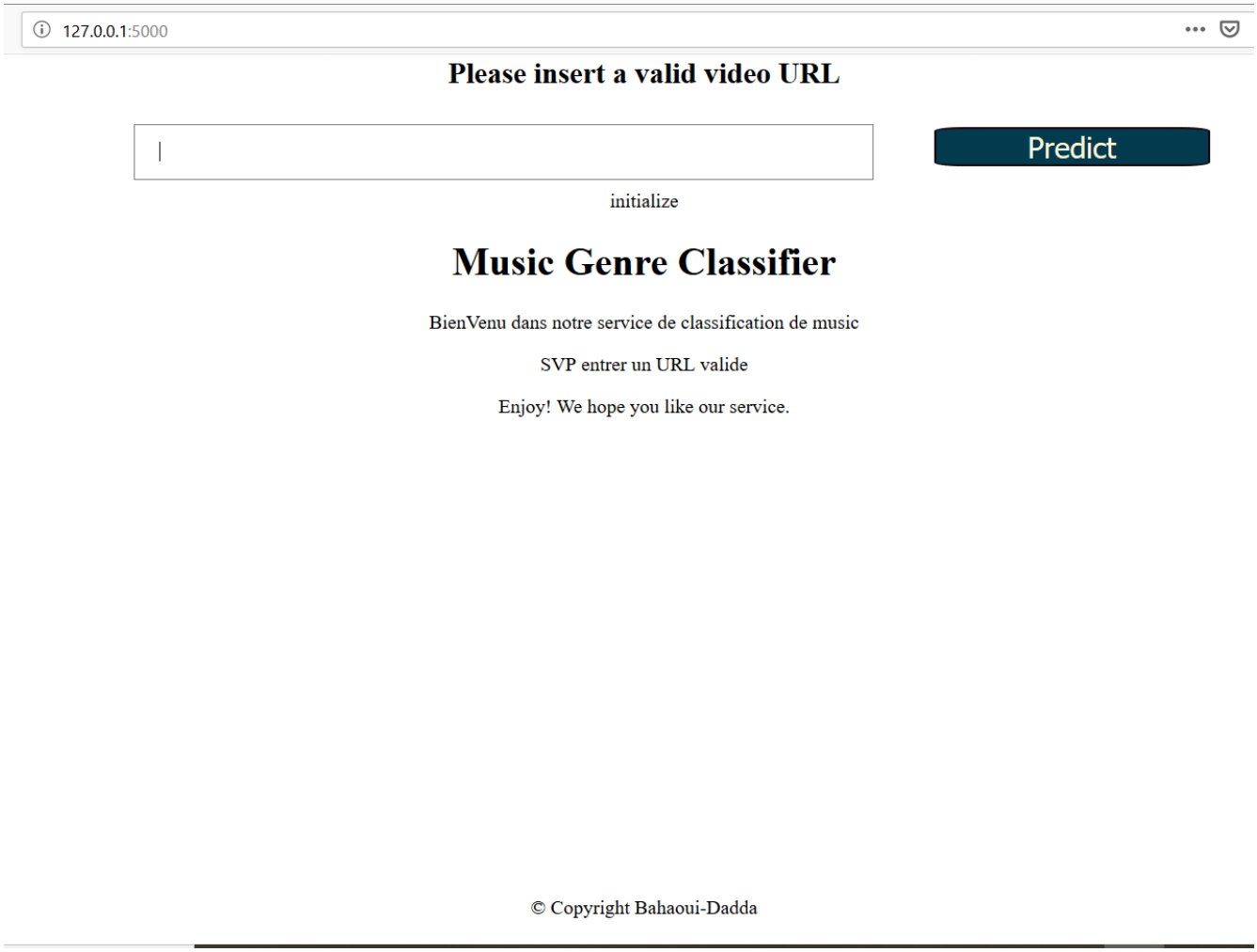


FIGURE 4.17 – Interface d'accueil

Capture d'écran de l'interface de prédiction

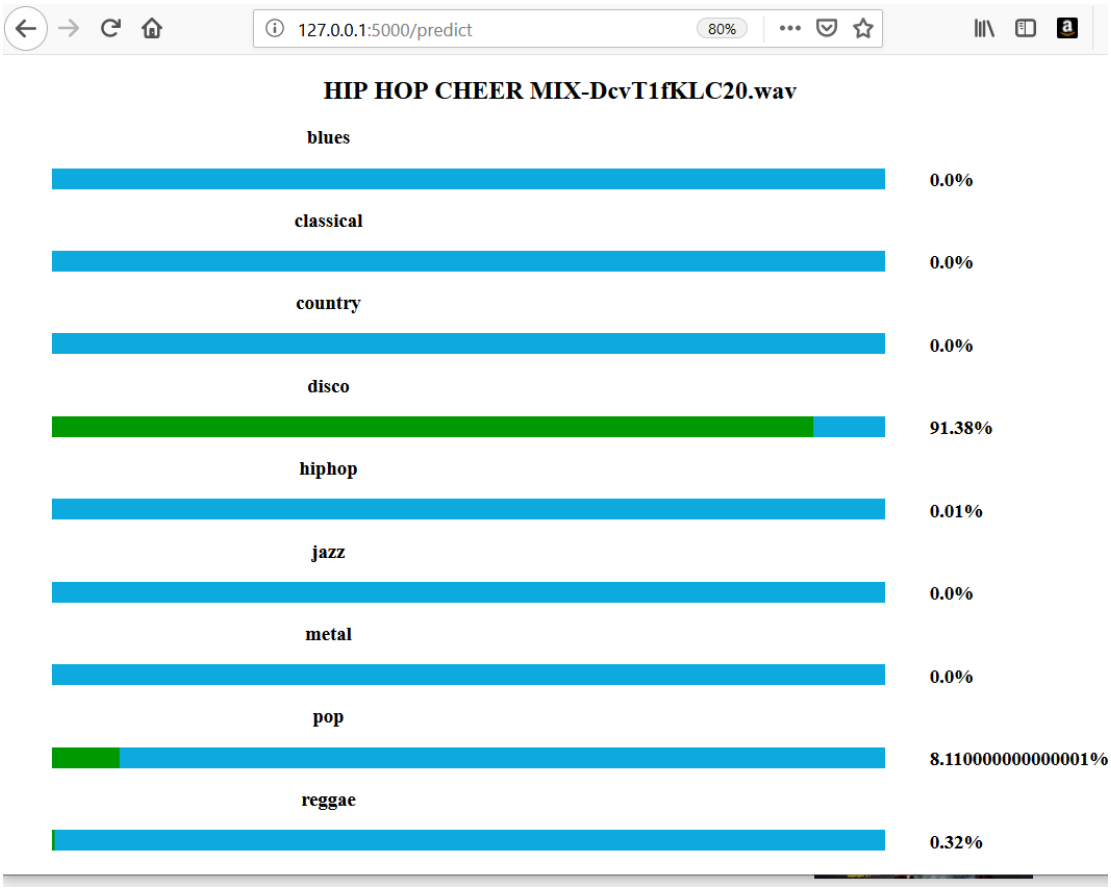


FIGURE 4.18 – interface de prédiction

Bibliographie

[1] Musical Genre Classification of Audio Signals
George Tzanetakis, Student Member, IEEE, and Perry Cook, Member, IEEE (2002).
https://pdfs.semanticscholar.org/4ccb/0d37c69200dc63d1f757eafb36ef4853c178.pdf

[2] Automatic Genre Classification of Music Content
Nicolas Scaringella, Giorgio Zoia, and Daniel Mlynek (2005).
https://sci-hub.tw/https://ieeexplore.ieee.org/document/1598089

[3] WaveNet : A Generative Model for Raw Audio
Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu (2016).
https://arxiv.org/abs/1609.03499

[4] Audio Spectrogram Representations for Processing with Convolutional Neural Networks
L. Wyse(2017).
https://arxiv.org/abs/1706.09559

[5] Music Genre Classification using Machine Learning Techniques
Hareesh BahuleyanUniversity of Waterloo, ON, Canada (2018).
https://www.researchgate.net/publication/324218667_Music_Genre_Classification_using_Machine_Learning_Techniques