

Sujet de BE – Systèmes d'indexation et de classification sonore

L'objectif de ce BE est de développer d'une part un système d'indexation sonore, et d'autre part, un système de classification sonore.

Ces systèmes seront développés en langage Python en utilisant le module numpy.

Présentation des données fournies

Les données consistent en un certain nombre d'extraits d'une seconde de titres musicaux. Chaque extrait est identifié par son nom, et caractérisé par son appartenance à un genre musical.

Les données sont divisées en deux sous-ensembles : un sous-ensemble d'apprentissage et un sous-ensemble de test, reconnaissables par l'extention *_app* et *_test*, respectivement pour les données d'apprentissage et de test.

Pour chacun de ces sous-ensembles, les données suivantes sont fournies :

- noms : vecteur contenant le nom des titres musicaux
- classes : vecteur contenant un entier identifiant le genre musical de chaque titre
 - 1 : classique
 - 2 : disco
 - 3 : jazz
 - 4 : rock
- donnees : matrice contenant les données audio. Chaque ligne correspond à un titre musical, et les valeurs en colonnes correspondent aux valeurs d'amplitude brutes du signal sonore.

Partie 1 : Extraction de descripteurs

Afin de représenter les signaux audio, un certain nombre de descripteurs seront extraits.

Dans un premier temps, le travail consistera donc, à partir des signaux audio numériques, à calculer leur enveloppe d'énergie comme cela a été vu en cours. Pour cela, la fonction *enveloppe_energie* devra être développée, prenant en paramètres un vecteur contenant les valeurs d'amplitudes brutes d'un signal sonore, ainsi que la taille de la fenêtre temporelle utilisée (caractérisée par son nombre d'échantillons). Cette fonction renverra l'enveloppe d'énergie du signal passé en paramètre, consistant en un vecteur contenant les valeurs de *RMS* calculées pour chaque fenêtre. Un taux de recouvrement de chaque fenêtre de 50% pourra être utilisé.

A partir de cette enveloppe d'énergie, un certain nombre de descripteurs sera calculé :

- le centroïde temporel (fonction *centroïde_temporel*)
- la durée effective (fonction *duree_effective*)
- l'énergie globale (fonction *energie_globale*)

Un quatrième descripteur sera calculé à partir de la représentation brute du signal : le taux de passage par zéro (fonction *zcr*).

Partie 2 : Indexation sonore

L'objectif de ce système d'indexation est de permettre l'interrogation d'une base de titres musicaux à partir d'une requête par l'exemple, de manière à extraire les titres les plus semblables à l'exemple.

Chaque titre de la base sera représenté par son vecteur de descripteurs calculés précédemment. A partir du vecteur de descripteurs, l'objectif de ce système d'indexation et d'extraire de l'ensemble des titres, le nom des 5 titres les plus semblables. Pour cela, une fonction *distance_descripteurs* pourra être développée, permettant de renvoyer la distance euclidienne entre les deux vecteurs de descripteurs passés en argument.

Le système pourra être évalué en utilisant en requête des titres de l'ensemble de test, la base étant constituées des titres de l'ensemble d'apprentissage.

Partie 3 : Classification sonore

L'objectif de ce système de classification est de déterminer le genre musical des titres musicaux. Pour cela, une fonction de classification *classification_kppv* devra être développée, permettant de renvoyer un entier indiquant la classe (et donc le genre) du titre musical dont le vecteur de descripteurs a été donné en argument. La distance mise en œuvre pourra être la distance euclidienne. Cette fonction utilisera un deuxième argument, qui est le nombre k de voisins considérés.

Afin d'évaluer les performances de ce classifieur, le taux de reconnaissance sur l'ensemble de test devra être calculé. Il faudra également étudier l'influence du paramètre k sur le taux de reconnaissance.

Travail à rendre

Ce travail d'indexation et de classification est à rendre pour le lundi 6 novembre, sous forme d'une archive .zip à déposer sur le site moodle. Cette archive contiendra un compte-rendu au format numérique (.pdf) ainsi que le programme complet (ensemble des fichiers .py). Le compte-rendu devra contenir notamment les explications concernant le principe des fonctions que vous avez programmées ainsi que leur code commenté (copier-coller des fichiers Python). Les expérimentations que vous aurez réalisées devront être décrites et les résultats commentés.