Dans ce doc on met le format des données qu' on utilise à chaque instant du code. Ça aidera pour la documentation. Et il faudra donc adapter le code pour qu'il respecte tout ça.

Interface (dans l'interface elle même, ce qui s'affiche à l'écran)

Dossier = string contenant l'adresse du dossier
les paramètres entrés par l'utilisateur sont, dans l'interface, de la forme :
nombre De Morceaux = entier positif
durée des morceaux = entier (en sec)
tonalité = string parmis ["A","A#","B","C","C#","D","D#","E","F","F#","G","G#"]
vitesse des morceaux = entier (en bpm)
type = string parmis ["Rythme seulement","Rythme et mélodie","Polyphonie"]

paramètres sauvegardés (ce qu'on a sauvegarder de ce que l'utilisateur à entrer dans l'interface)

Dossier = string contenant l'adresse du dossier
les paramètres entrés par l'utilisateur sont, dans l'interface, de la forme :
nombre De Morceaux = entier positif
durée des morceaux = entier (en sec)
tonalité = entier entre 0 et 11 (inclus)
vitesse des morceaux = entier (en bpm)
type = string parmis ["Rythme seulement","Rythme et mélodie","Polyphonie"]

table de conversion des tonalités
["A","A#","B","C","C#","D","D#","E","F","F#","G","G#"]
[9, 10, 11, 0, 1, 2, 3, 4, 5, 6, 7, 8]

Base de données (les fichier dans la base entrée par l'utilisateur

ensemble de fichiers au format MIDI

Traitement de la base de données (les fichiers de la base de données dans les codes du fichier extraire.py, ces codes sont appelés dans script.py)

conversion MIDI to CSV

les fichiers de la base de données sont au format CSV comme décrit par la documentation de la librairie miditocsv

pré-traitement des données

On récupère certaines informations du fichier CSV précédemment obtenu. On garde les informations suivantes qu'on stocke dans un objet de classe Morceau

filename = une string le nom du fichier
 # Header information
format = entier (0,1 ou 2)
nbTracks = entier (supérieur à 0)
division = entier (supérieur à 0)
 # SMPTE_offset information
smpteHour = entier
smpteMinute = entier
smpteSecond = entier
smpteFrame = entier
smpteFracFrame = entier
 # Time signature information
tsNum = entier
tsDenom = entier
tsClick = entier
tsNotesQ = entier
 # Key signature information
ksKey = entier
ksMinMaj = chaine de caractère (Major ou Minor)
 # Tempo information
tempoList = liste de chaîne de caractères (toujours en csv)
 # Tracks information
trackList = liste de listes contenant des notes_on
 # Dictionnaire temps vers note et liste des notes
time_to_note_dict = dictionnaire sous la forme
durée_note:représentation_note
liste_notes = liste contenant les différentes notes

Les informations suivantes ne sont pas conservées

Toutes les lignes relatives avec : Copyright, Text_t, Program_c, Control_c

On ne garde pas les lignes qui ne fournissent pas d'informations.

traitement des données pour le RNN

On applique un deuxième traitement sur ces mêmes données pour obtenir quelque chose d'utilisable par le RNN

On stocke donc les informations sous la forme d'une string au format suivant "a:b:c a:b:c"

chaque triplet a:b:c représente une note, Les triplets sont séparés par un espace et chaque terme du triplet est séparé par “:”

a = un entier qui représente le début de la note par rapport au début de la note précédente (en midiclock)

b = un caractère, la longueur de la note. La liste suivante donne la longueur en fonction de la lettre.

a	*ronde pointée*	*n*	*croche triolet*
b	*ronde*	*o*	*double croche pointée*
c	*blanche pointée*	*p*	*double croche*
d	*blanche*	*q*	*croche quintolet*
e	*ronde quintolet*	*r*	*double croche triolet*
f	*blanche triolet*	*s*	*triple croche pointée*
g	*noire pointée*	*t*	*triple croche*
h	*noire*	*u*	*double croche quintolet*
i	*blanche quintolet*	*v*	*triple croche triolet*
j	*noire triolet*	*w*	*quadruple croche pointée*
k	*croche pointée*	*x*	*quadruple croche*
l	*croche*	*y*	*triple croche quintolet*
m	*noire quintolet*	*z*	*quadruple croche triolet*

c = un entier qui représente la hauteur de la note (entre 1 et 128, chaque entier représente une touche d’un piano

Traitement dans le RNN (les strings passées en entré du RNN sont à nouveau modifiées deux fois)

Si on souhaite entraîner le RNN à générer des rythmes on peut lui donner les informations sous plusieurs formes.

- Soit on considère que toutes les notes sont jouées les unes à la suite des autres, sans pause. A ce moment la on peut convertir notre string de format “a:b:c a:b:c a:b:c” en “bbb” (on ne garde que la durée des notes et on colle tous les caractères)
- Soit on ne conserve que le rythme et on supprime le troisième caractère “a:b a:b a:b”

Si on souhaite entraîner le RNN à générer des mélodies on garde le “a:b:c a:b:c”

à venir : les tenseurs