SoundScapify: Song Recommender Based on Soundscape

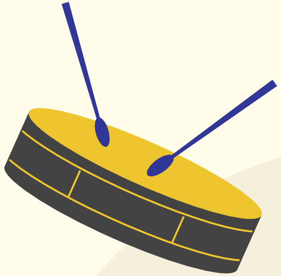
Adi H. Kusuma - DSI 28

6th July 2022

Table of contents

01

Introduction



"Music is the Soundtrack of Your Life."

—Dick Clark

Music

Stress ↓

Well-Being ↑

https://www.cell.com/trends/cognitive-sciences/fulltext/S1364-6613(13)00049-1

48%

of commuters in America listen to musics

In Singapore,

~40%

listen to music during their commute

Problem Statement

- Build a song recommender based on current ambience sound and mood
- Develop a classifier model to classify the acoustic scene
  - Target accuracy score > 80%
- Create criteria of Audio Feature Ranges as a metric for recommended

Scope of Data

Dataset	Description
fold1_train.csv	Original dataset from TAU Urban Acoustic Scenes 2022 Mobile, development dataset that contains filename and scene label for training purposes
fold1_test.csv	Original dataset from TAU Urban Acoustic Scenes 2022 Mobile, development dataset that contains filename and scene label for testing purposes
valence_arousal_dataset.csv	Dataset of songs from multiple genres that is scraped using Spotify API which includes the valence and energy value of the songs
recommend_criteria.csv	Dataset of criteria for the valence and energy range based on the label, which is extracted from *valence_arousal_dataset.csv*

02
Exploratory Data Analysis

TAU Urban Acoustic Scene 2022 dataset

Scene Label

Initial label: 10 nos

Label to be used: 4 nos

park <-

street_traffic <-

metro <-

bus <-

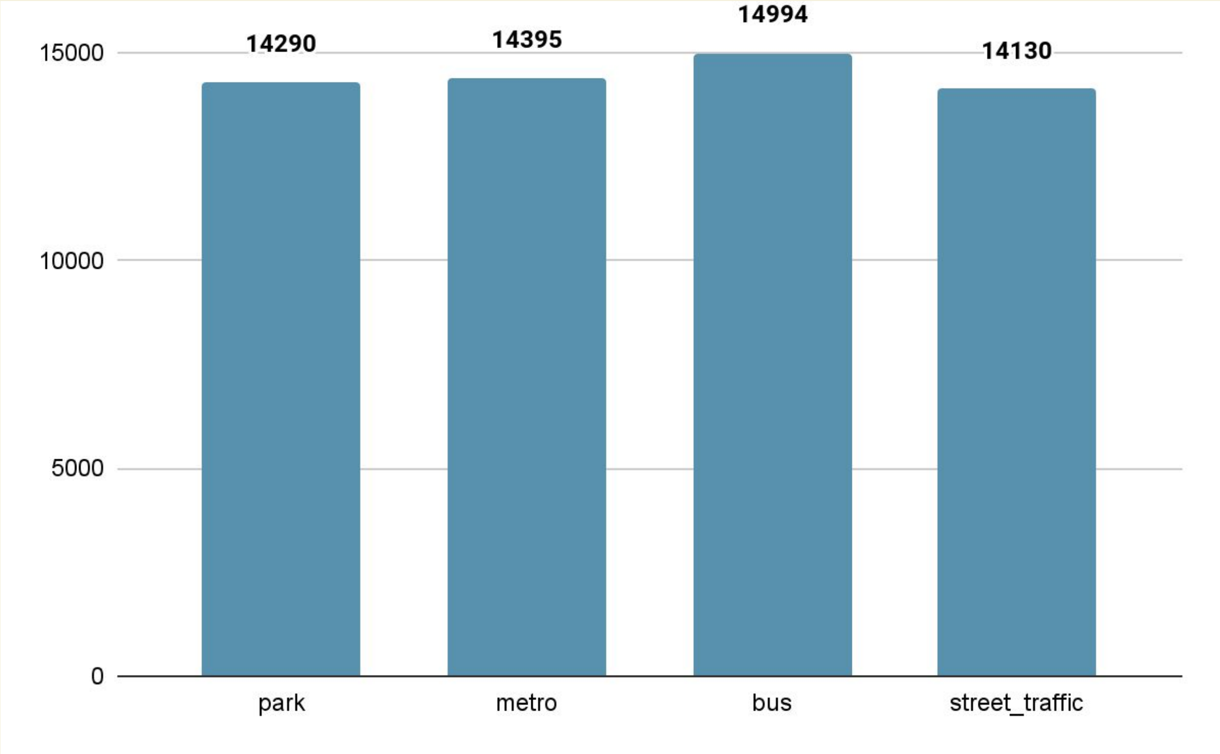
Audio Files

1-second audio clips for 10 different countries in Europe

Singapore Context

Added recordings of bus and MRT

Bar Chart of Scene Label

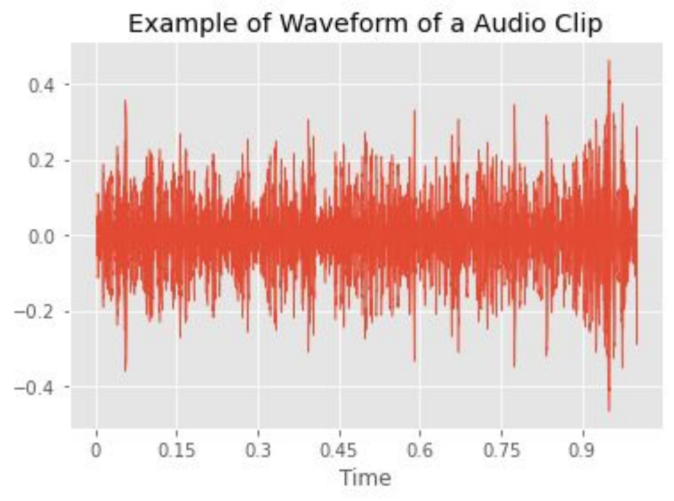


Preprocessing Audio File: Waveform

Audio File

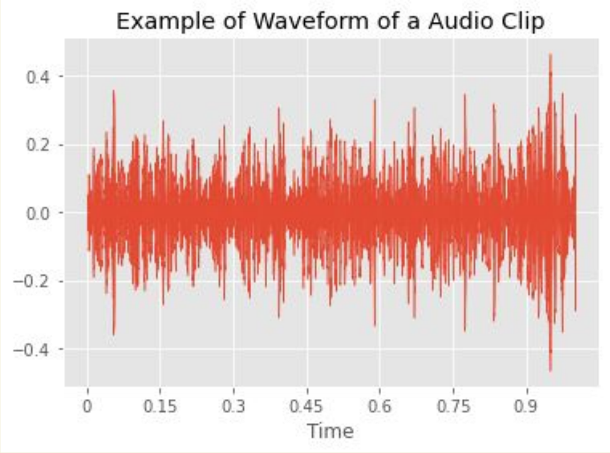
Load .wav files with
librosa package

Waveform



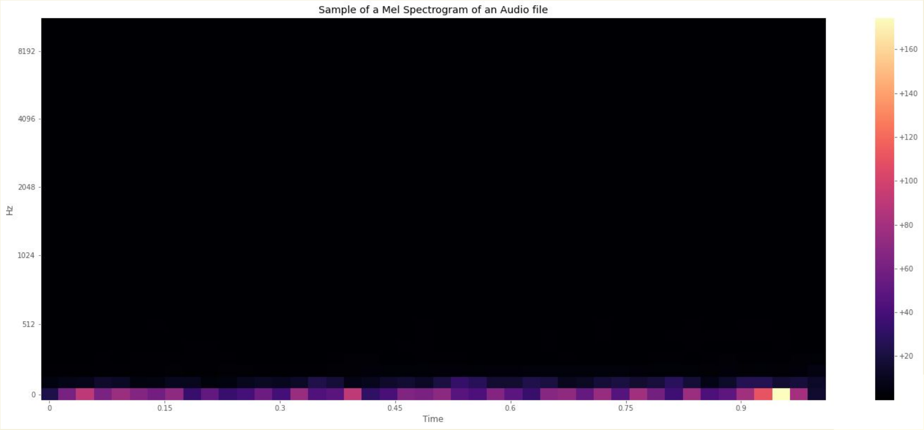
Preprocessing Audio File: Mel-Spectrogram

Waveform



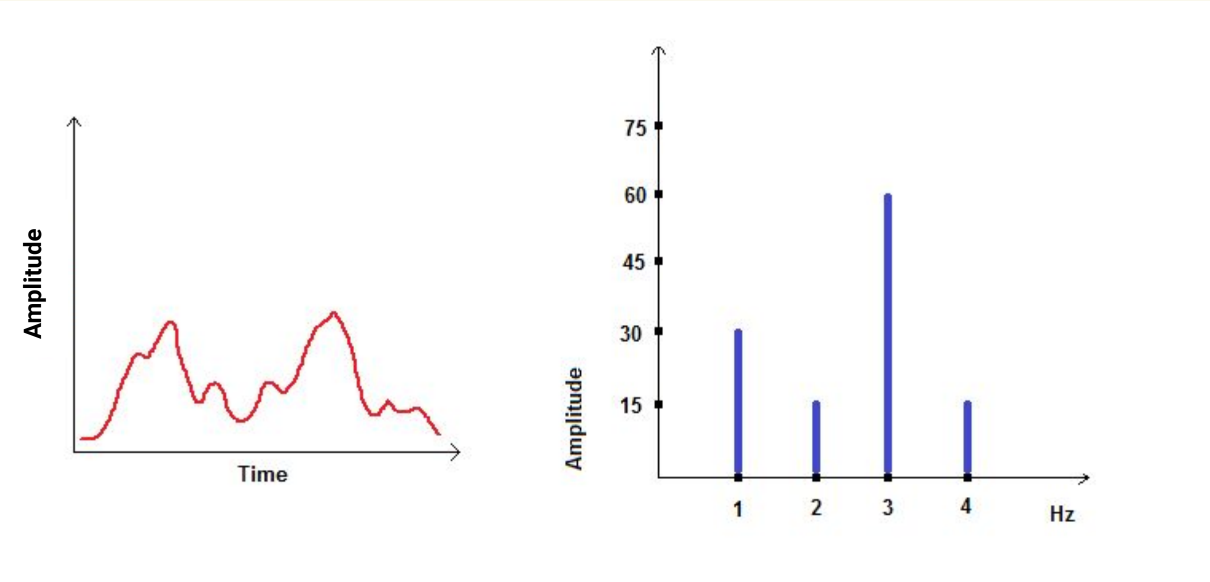
STFT & No of mels

Mel-Spectrogram

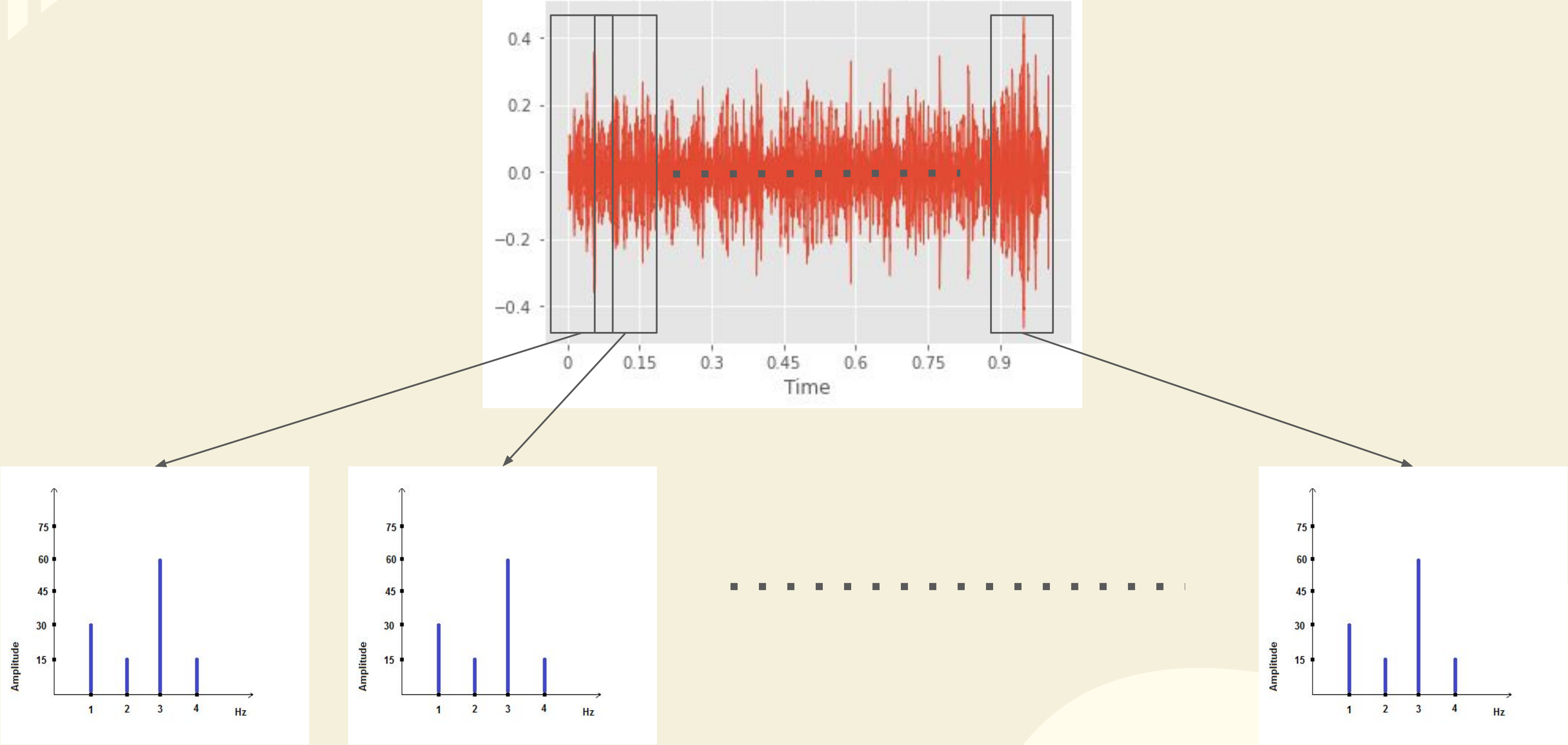


Fourier Transform

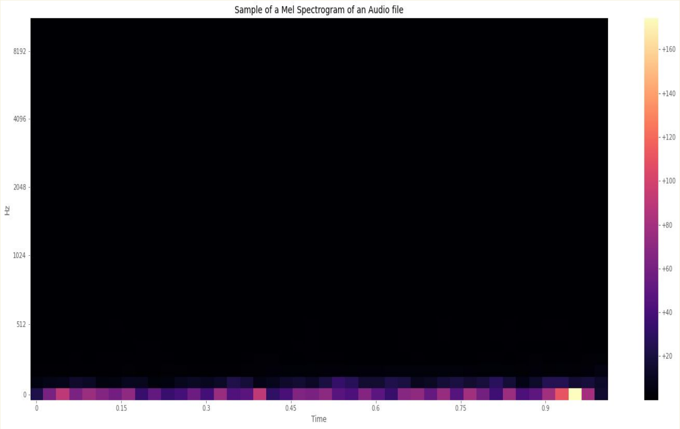
Representation of waveform based on the frequency and amplitude



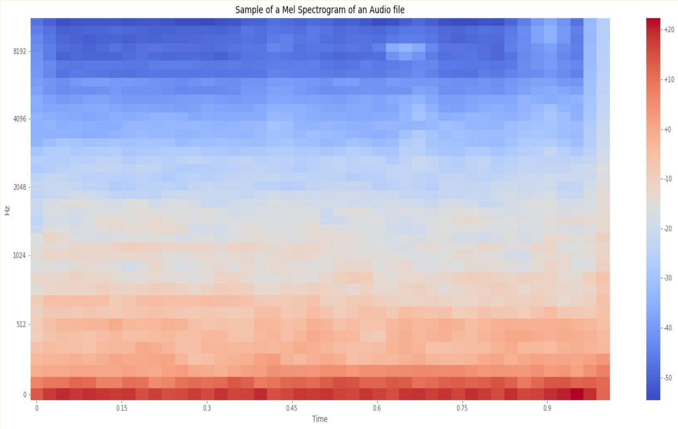
Short-Term Fourier Transform



Preprocessing Audio File: Convert to dB scale



dB scale

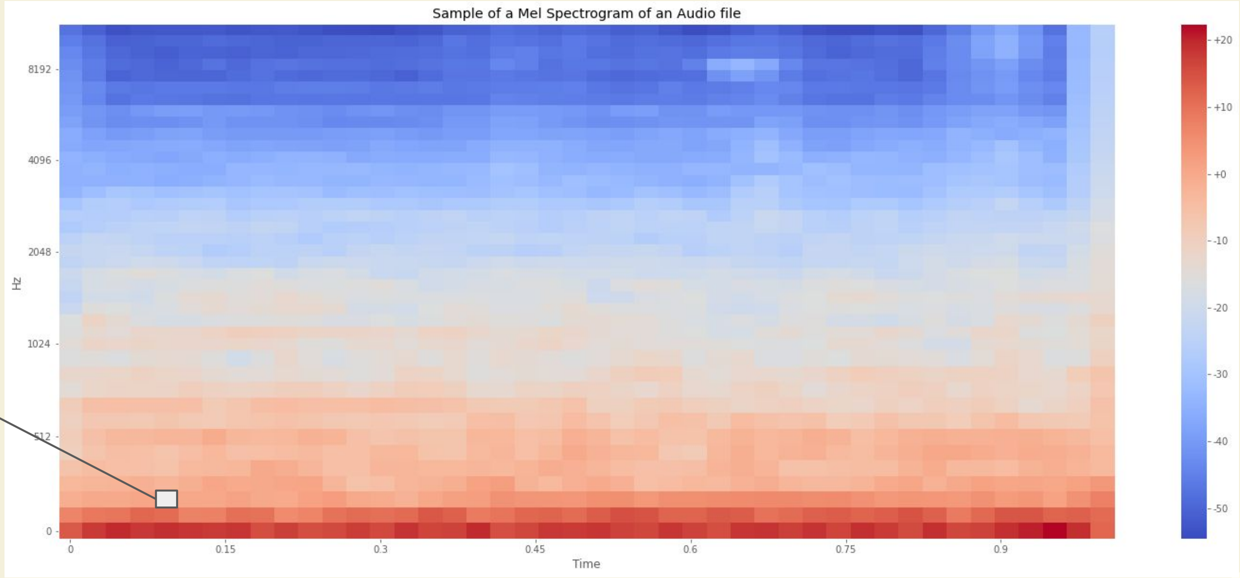


Mel-Spectrogram

1 mel

window_size

40 mels



44 windows

Valence Arousal Dataset Scraping Process

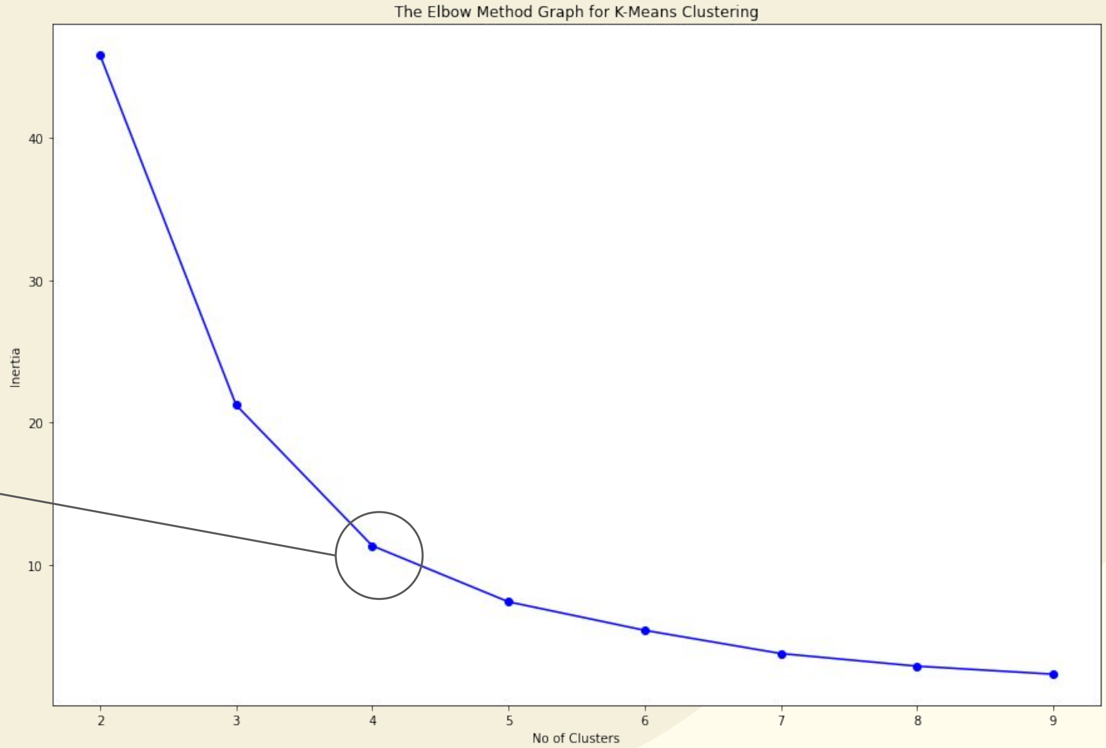
All genres available in Spotify → Spotify Recommender → Track information:

- Id
- Track name
- Artist name
- Valence
- Energy

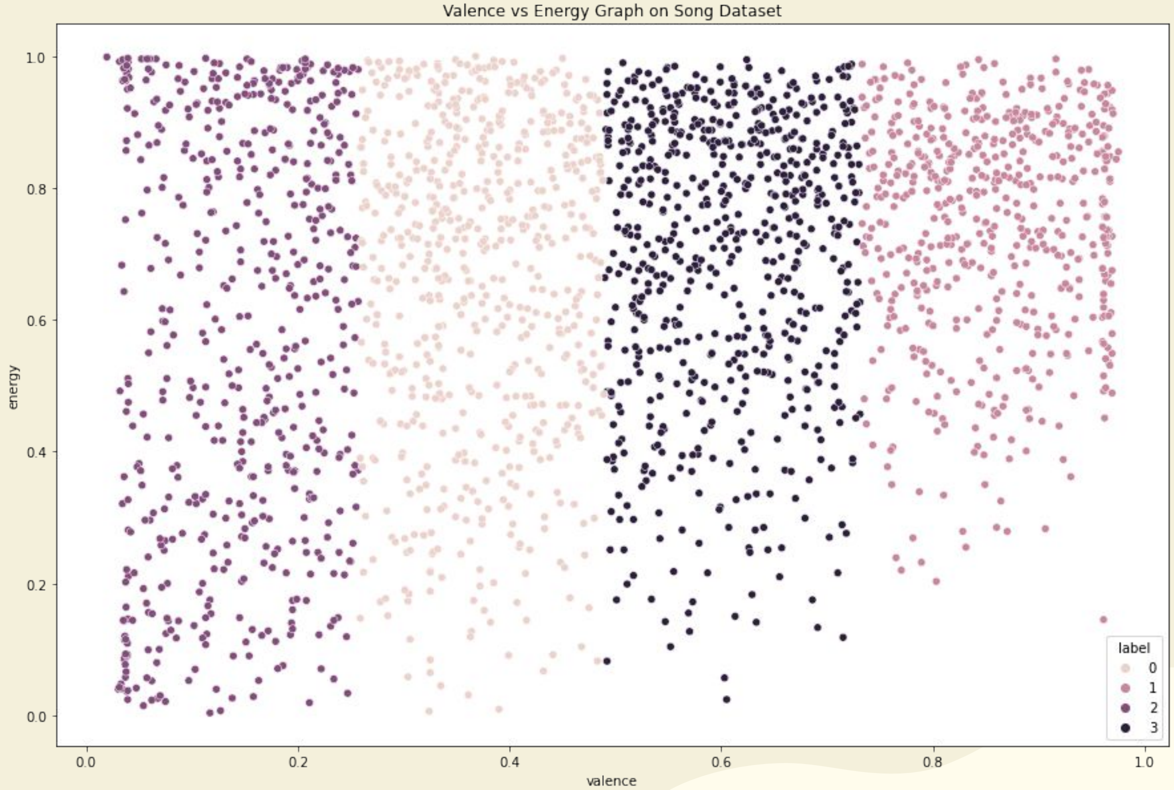
K-Means Clustering: Elbow Graph

To check the inertia of the cluster and find optimal cluster number

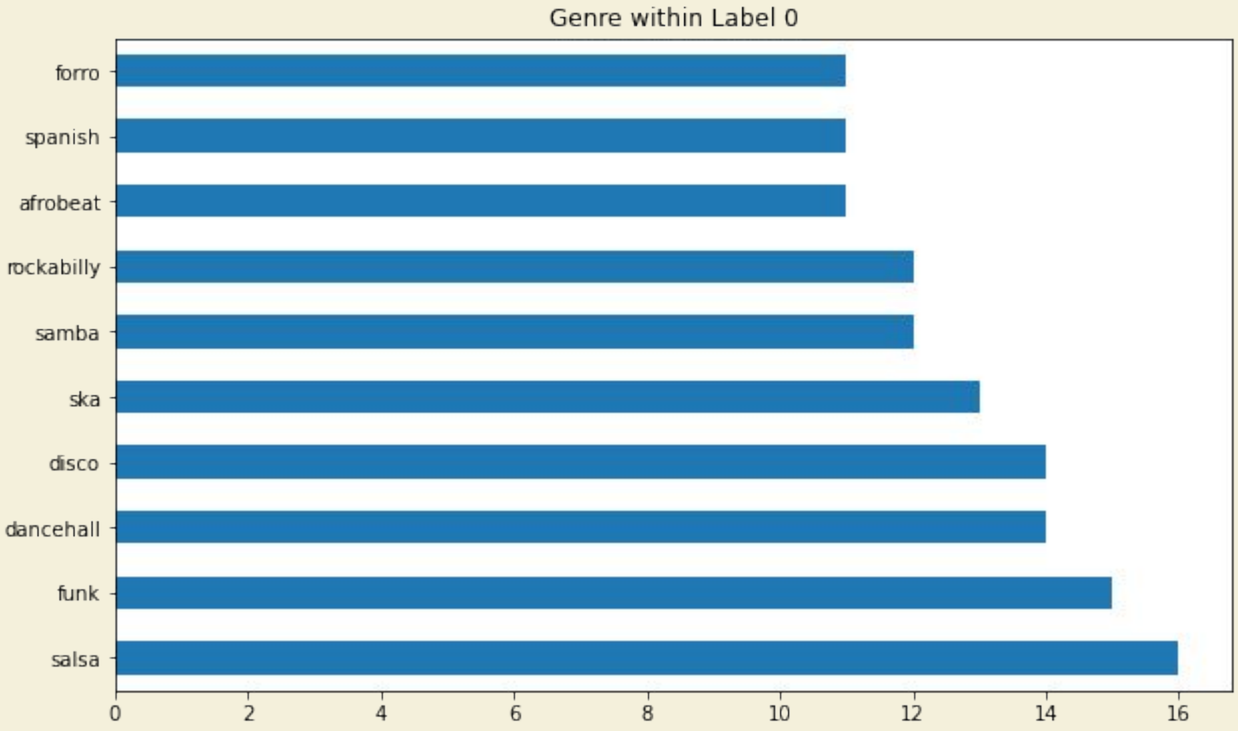
Optimal Cluster : 4



K-Means Clustering on the Dataset



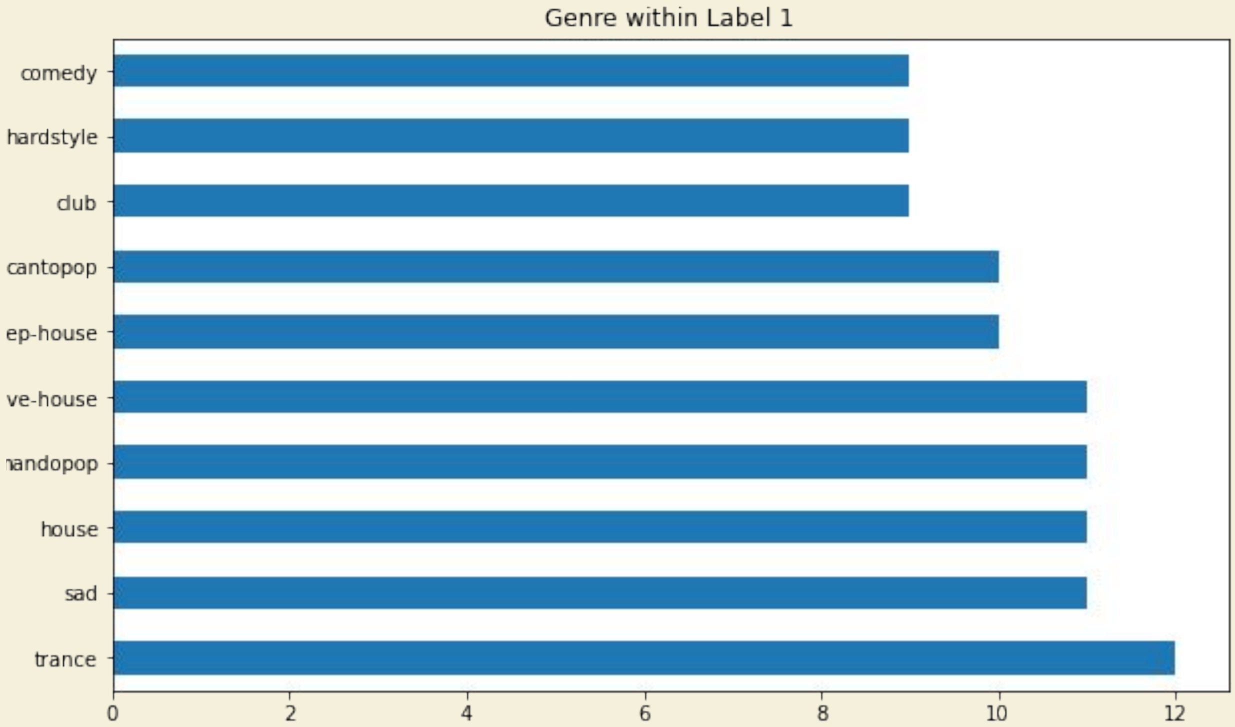
Label 0 as Metro



Based on the sample music heard, the songs which has uptension beat. This work well the soundscape of metro

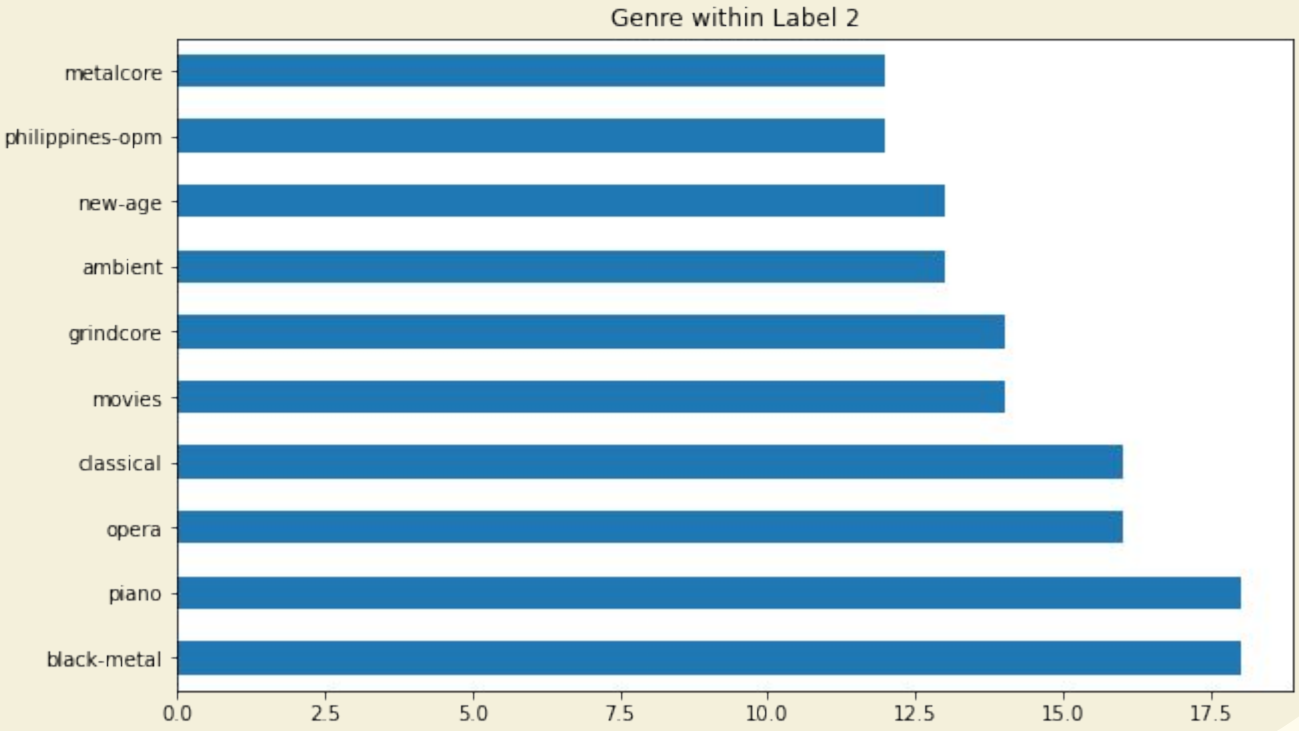
The genre also give the same vibe

Label 1 as Bus



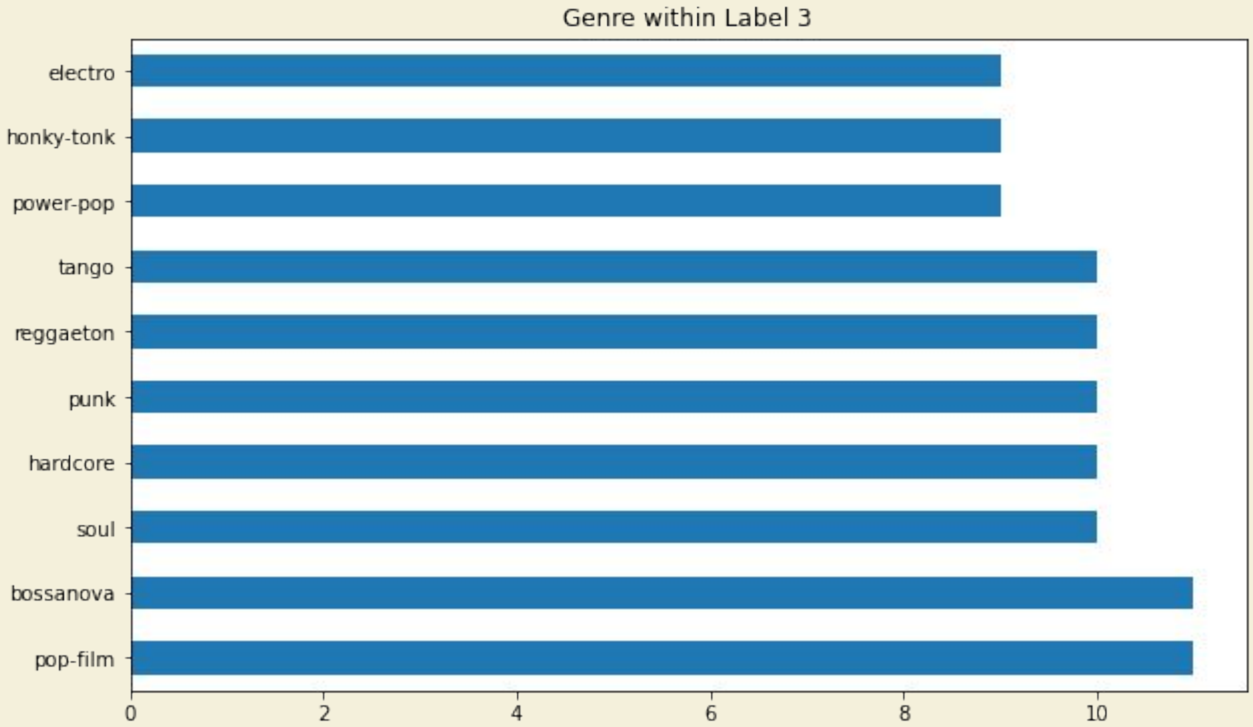
The genre within label 1 has layback vibe to them, which makes them resonates well with driving/riding bus

Label 2 as Park



The sample songs that are presented and give similar ambience of park.

Label 3 as Street Traffic



The sample songs and top genres in label 3 give similar ambience of traffic sound.

Criteria Value Range

Label	Valence_min	Valence_max	Energy_min	Energy_2nd	Energy_3rd	Energy_Max
Metro	0.2590	0.489	0.00591	0.337273	0.668637	1.000
Bus	0.7330	0.975	0.14500	0.4288667	0.712333	0.996
Park	0.0196	0.257	0.00341	0.335273	0.667137	0.999
Street_Traffic	0.4900	0.731	0.02380	0.347533	0.671267	0.995

03

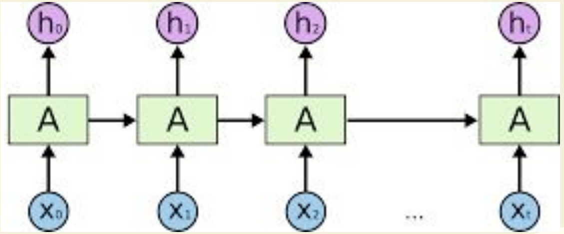
Modelling & Result

Model Input & Output Variable Preprocessing

1. Set input and output variable
2. Label Encoding the output variable
3. Train test split the dataset
4. Check train sample size vs batch size
5. Initiate DataGenerator

Long Short-Term Memory (LSTM) Neural Network

- Part of Recurrent Neural Network
- LSTM Neural Network is able to capture the previous time sequence model data and use the memory on the next time sequence model to have a better classification.

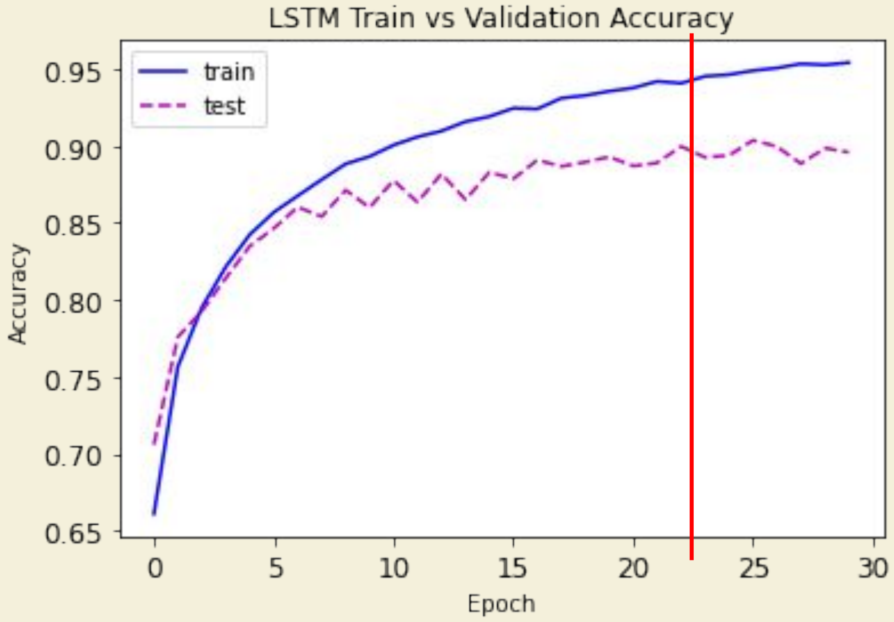


T0 T1 T2 Tt

Model Layer Summary

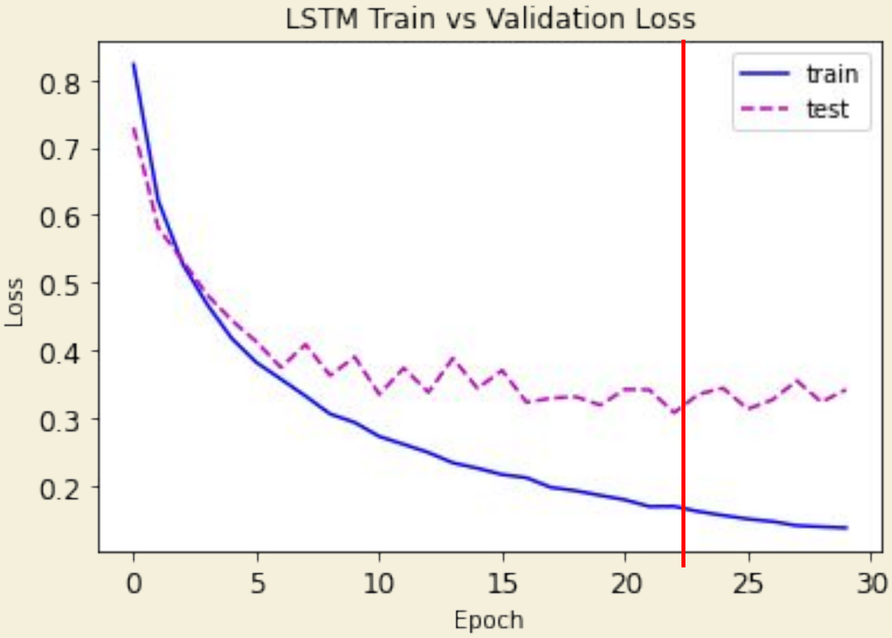
Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	(None, 44, 40, 1)	0	[ ]
batch_norm (LayerNormalization)	(None, 44, 40, 1)	80	['input_1[0][0]']
reshape (TimeDistributed)	(None, 44, 40)	0	['batch_norm[0][0]']
td_dense_tanh (TimeDistributed)	(None, 44, 64)	2624	['reshape[0][0]']
bidirectional_lstm (Bidirectional)	(None, 44, 64)	24832	['td_dense_tanh[0][0]']
skip_connection (Concatenate)	(None, 44, 128)	0	['td_dense_tanh[0][0]', 'bidirectional[0][0]']
dense_1_relu (Dense)	(None, 44, 64)	8256	['skip_connection[0][0]']
max_pool_1d (MaxPooling1D)	(None, 22, 64)	0	['dense_1_relu[0][0]']
dense_2_relu (Dense)	(None, 22, 32)	2080	['max_pool_1d[0][0]']
flatten (Flatten)	(None, 704)	0	['dense_2_relu[0][0]']
dropout (Dropout)	(None, 704)	0	['flatten[0][0]']
dense_3_relu (Dense)	(None, 32)	22560	['dropout[0][0]']
softmax (Dense)	(None, 4)	132	['dense_3_relu[0][0]']

Model Accuracy across Epochs



- First few epochs, train set underfitting (accuracy < 0.8)
- After 5 epochs, the accuracy difference between train and validation set widens.
- Both the train and validation set has accuracy higher than 0.8 after epoch 5

Model Loss across Epochs : Best Model = Epoch 23



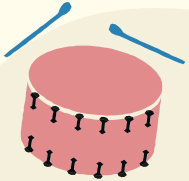
- The best model is determined by the smallest validation loss
- The movement of the value across epochs is inverse to accuracy

Model Accuracy on Train and Validation Set (Epoch 23)

Dataset	Accuracy	Loss
Train set	0.945338	0.161300
Validation set	0.892643	0.334957

Confusion Matrix of Prediction on Unseen Dataset

- Street_traffic scenes have the best accuracy
- A lot of park acoustic scenes are misclassified as street_traffic
- Bus and metro are misclassified with one another as well



04
WebApp Live Demo

Scripts for the webapp:

- app.py
- authorization.py
- spotify.py

05

Conclusion

Limitations

1. Limited number of acoustic scenes that are able to be classified
2. Time limitations to fine tune the model or explore different types of deep learning model
3. Limited dataset on valence and energy in relation to acoustic scene
4. The microphone and machine not able to detect the ambience sound

Future Works

1. Improve the model accuracy by tuning the current model or introduce different type of deep learning model
2. Increase the number of acoustic scene to be trained and classified
3. Collect data that represent valence and energy of acoustic scenes
4. Develop an Android apk to deploy the app to utilize a better microphone

Thanks

"Music is life itself"
– Louis Armstrong