Music Genres Classification and Recommendation System

Jo-Chen Ma, Ting-Yu Wong, Kai-Ting Tsao, Kuan Lin
Mentor:Ting-Wei Su

Introduction

- **Classifying the multi-type of the audio-recordings and build up the recommendation model to generate the list of songs with similar genres.**

- **We aim to accurate distinguish the genres of the songs and customize optimal playlist recommendation.**

- **While user input a single song, we can provide three songs that includes high-related genres.**

Track [203] : [12] [66] [4] [1235] [18]

[12] Rock

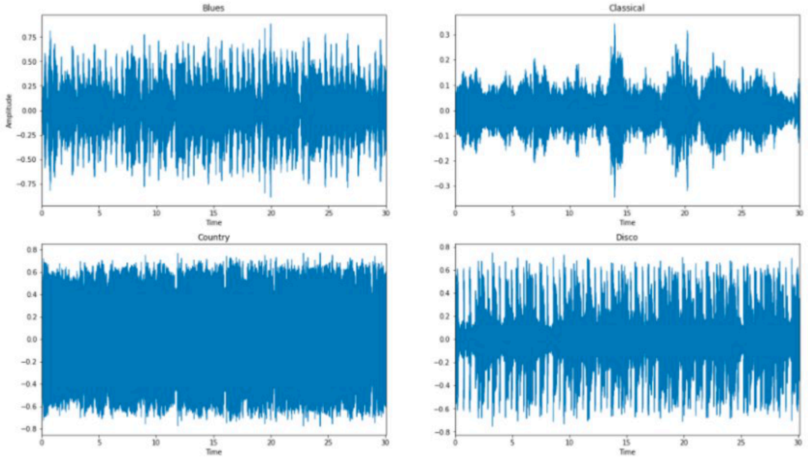
[85]Garage | [66]Indie-Rock | [70]Industrial | [58]Psych-Rock

[25]Punk | [26]Post-Pock | [27]Lo-Fi | [45]Loud-Rock

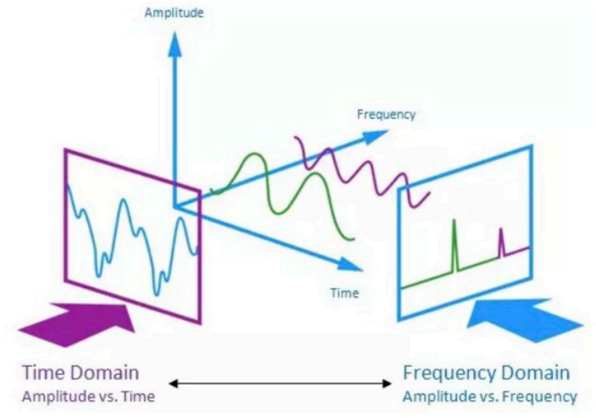
genre_id	#tracks	parent	title	top_level
2	3,738	0	International	2
3	1,250	0	Blues	3
4	1,647	0	Jazz	4
5	2,027	0	Classical	5
9	1,642	0	Country	9
14	1,235	0	Soul-RnB	14
18	3,151	1235	Soundtrack	1235
21	6,737	0	Hip-Hop	21
25	7,033	12	Punk	12
26	1,345	12	Post-Rock	12
27	3,821	12	Lo-Fi	12
33	1,165	17	Psych-Folk	17
42	2,491	15	Ambient Ele	15
45	1,704	12	Loud-Rock	12
58	1,644	12	Psych-Rock	12
66	4,331	12	Indie-Rock	12
70	1,106	12	Industrial	12
76	4,869	10	Experimenta	10
85	3,041	12	Garage	12
103	3,144	17	Singer-Songw	17
107	2,813	1235	Ambient	1235
181	1,457	15	Techno	15
182	1,252	15	House	15
183	1,154	15	Glitch	15
236	2,061	15	IDM	15
286	1,271	15	Trip-Hop	15
297	1,783	15	Chip Music	15
362	1,456	10	Synth Pop	10
495	1,488	15	Downtempo	15

Dataset

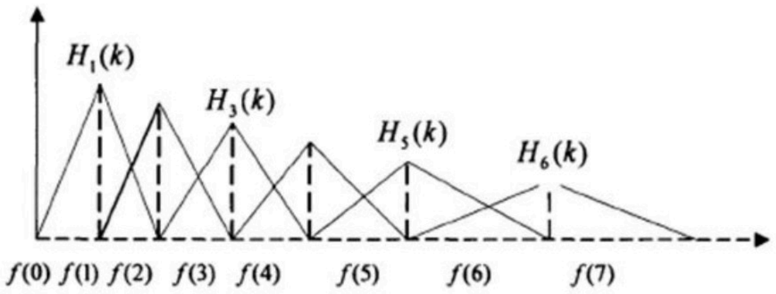
- The datasets used in our project were collected International by Society for Music Information Retrieval Conference (ISMIR) and are available at https://github.com/mdeff/fma .
- tracks.csv: per track metadata such as ID, title, artist, genres, tags and play counts, for all 106,574 tracks.
- genres.csv: all 163 genres with name and parent (used to infer the genre hierarchy and top-level genres).
- features.csv: common features extracted with librosa.
- echonest.csv: audio features provided Spotify for a subset of 13,129 tracks.



Audio Time Series



Fourier Transform



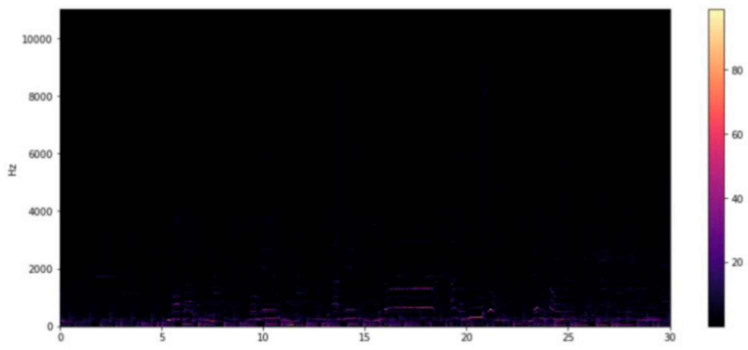
Triangular Bandpass Filters

Dataset

- **By Librosa Package**

```
stft = librosa.stft(data)
stft_db = librosa.amplitude_to_db(abs(stft))
plt.figure(figsize=(14, 6))
librosa.display.specshow(stft, sr=sr, x_axis='time', y_axis='hz')
plt.colorbar()
```

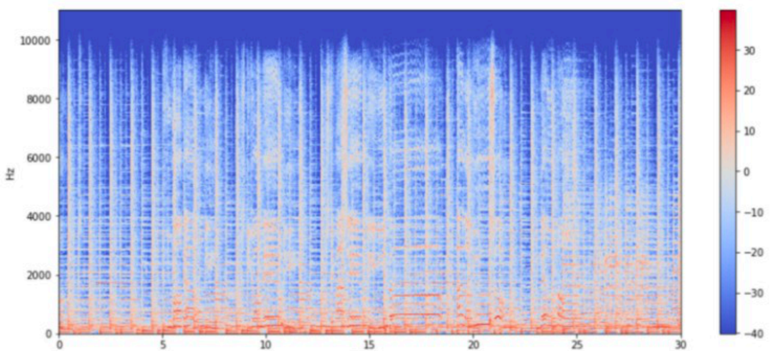
```
<matplotlib.colorbar.Colorbar at 0x7fa45036d490>
```



MFCC Spectrogram before transfer

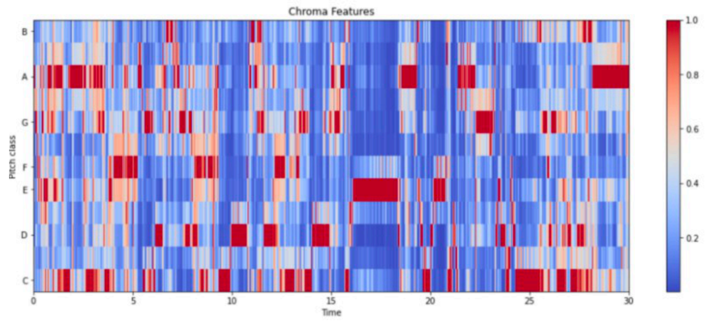
```
stft = librosa.stft(data)
stft_db = librosa.amplitude_to_db(abs(stft))
plt.figure(figsize=(14, 6))
librosa.display.specshow(stft_db, sr=sr, x_axis='time', y_axis='hz')
plt.colorbar()
```

```
<matplotlib.colorbar.Colorbar at 0x7fa44f047160>
```



MFCC Spectrogram after transfer

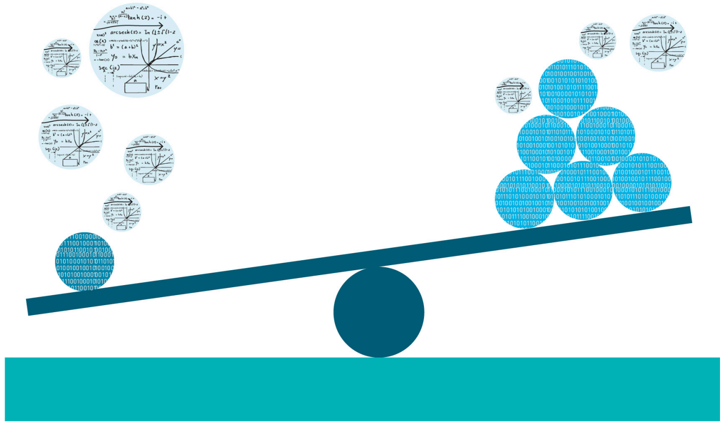
```
import librosa.display as lplt
chroma = librosa.feature.chroma_stft(data, sr=sr)
plt.figure(figsize=(16,6))
lplt.specshow(chroma, sr=sr, x_axis='time', y_axis='chroma', cmap='coolwarm')
plt.colorbar()
plt.title("Chroma Features")
plt.show()
```

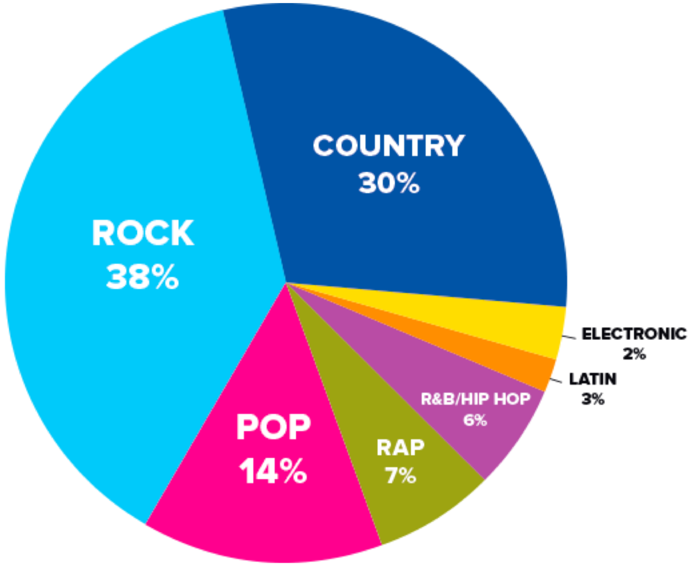


Chroma Spectrogram

Data Preprocessing

- Split and shuffle the data into train, valid , and test set (70%, 20%, and 10%)
- We choose 32 genres[1] that accounted for the greatest proportion of the dataset as the multi true label of our model.
- Balance the amount of songs for different genres so that the weight of each genre can be more easier to learn.





[1] International, Blues, Jazz, Classical, Country, Experimental Pop, Synth Pop, Punk, Post-Rock, Lo-Fi, Loud-Rock, Noise-Rock, Psych-Rock, Indie-Rock, Industrial, Garage, Post-Punk, Hardcore, Soul-RnB, Ambient Electronic, Techno, House, Glitch, IDM, Trip-Hop, Chip Music, Downtempo, Psych-Folk, Singer-Songwriter, Hip-Hop, Soundtrack, Ambient.

Algorithm and Training

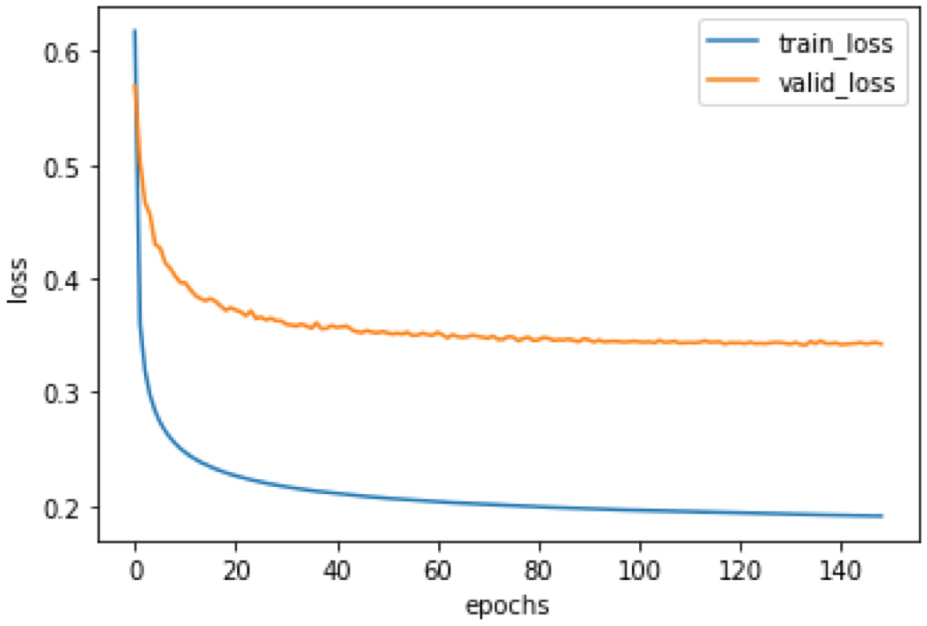
- We choose deep neural network (DNN) as our model with Keras.
- By monitoring val_loss and val_auc, we use EarlyStopping and ModelCheckpoint to prevent our model from overfitting.
- We got 79% accuracy by applying Adagrad optimizer (learning_rate= 5e-4, epsilon= 1e-6)

```
genres_list =
["2","3","4","5","9","76","362","25","26","27","45","58","66","70","85","14","
42","181","182","183","236","286","297","495","33","103","21","18","107"]
```

```
[0.1162 0.0187 0.0359 0.0007 0.1931 0.2323 0.1404 0.0413 0.0196 0.1058
 0.0284 0.0271 0.2207 0.0814 0.0567 0.1143 0.1488 0.0641 0.0302 0.0182
 0.01   0.0237 0.0149 0.1049 0.0522 0.2001 0.2317 0.0036 0.0048]
```

```
prediction = np.asarray(prediction > 0.15, dtype=int)
```

```
[0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1]
```



Testing and Recommendation System

- By using the train model, we could give our test data predicted multi-label.
- A song could be successfully tagging with its genres, and our recommendation system will use set to match top three highest related songs to the users.

OUR MODEL

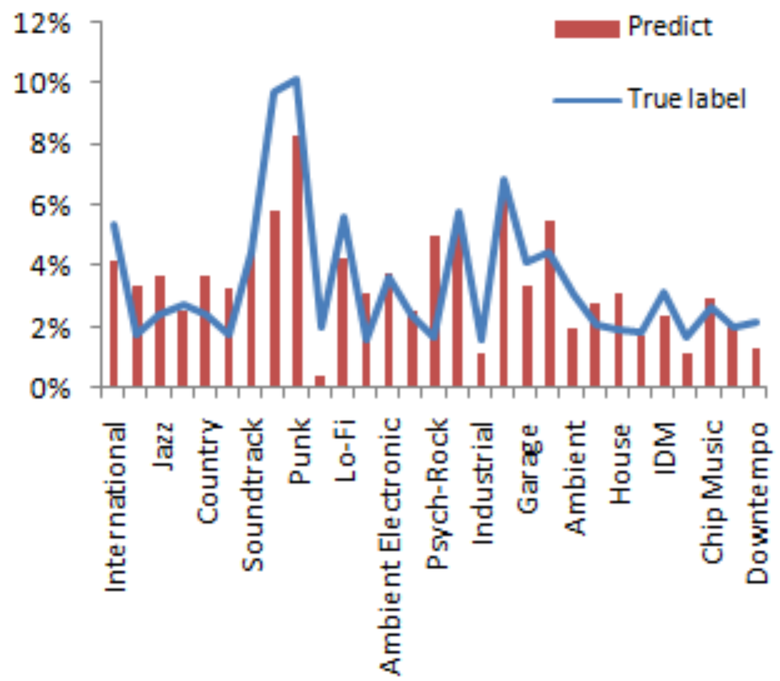
SONGS WITH MULTI-LABELS

Input track

{'103', '17', '33', '3'}

System Recommend

{'33', '3', '103'}



Discussion and Conclusion

- There are more than 32 genres in our data set, but the rest of the genres can not accurately predict due to insufficient numbers of tracks.
- Because of a large amount of audio features(518), the genres can be more precise with more variety of tracks.
- We can includes more type of genres to neural network by enhancing the data set.
- Let our recommendation learn user's preference by the feedback of songs it recommends to them.

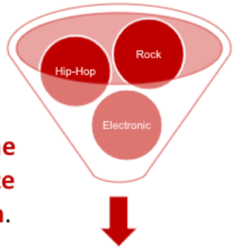
Music Genres Classification and Recommendation System

Jo-Chen Ma, Ting-Yu Wong, Kai-Ting Tsao, Kuan Lin

Mentor:Ting-Wei Su

Introduction

- Classifying the multi-type of the audio-recordings and build up the recommendation model to generate the list of songs with similar genres.
- We aim to accurate **distinguish the genres of the songs** and **customize optimal playlist recommendation**.
- While user input a single song, we can provide three songs that includes high-related genres.



Dataset

- The datasets used in our project were collected International by Society for Music Information Retrieval Conference (ISMIR) and are available at https://github.com/mdeff/fma .
- tracks.csv: per track metadata such as ID, title, artist, genres, tags and play counts, for all **106,574 tracks.**
- genres.csv: all **163 genres** with name and parent (used to infer the genre hierarchy and top-level genres).
- features.csv: common features extracted with librosa.
- echonest.csv: audio features provided Spotify for a subset of 13,129 tracks.

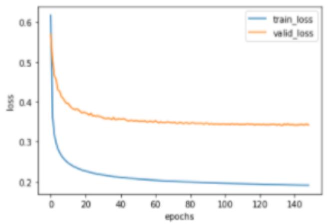
Methodology and Results

Data Preprocessing

- Split and shuffle the data into train, valid , and test set (70%, 20%, and 10%)
- We choose 32 genres[1] that accounted for the greatest proportion of the dataset as the multi true label of our model.
- Balance the amount of songs for different genres so that the weight of each genre can be more easier to learn.

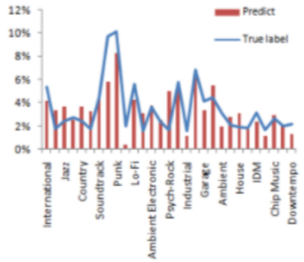
Algorithm and Training

- We choose deep neural network (DNN) as our model with Keras.
- By monitoring val_loss and val_auc, we use EarlyStopping and ModelCheckpoint to prevent our model from overfitting.
- We got 79% accuracy by applying Adagrad optimizer (learning_rate= 5e-4, epsilon= 1e-6)



Testing and Recommendation System

- By using the train model, we could give our test data predicted multi-label.
- A song could be successfully tagging with its genres, and our recommendation system will use set to match top three highest related songs to the users.



[1] International, Blues, Jazz, Classical, Country, Experimental Pop, Synth Pop, Punk, Post-Rock, Lo-Fi, Loud-Rock, Noise-Rock, Psych-Rock, Indie-Rock, Industrial, Garage, Post-Punk, Hardcore, Soul-RnB, Ambient Electronic, Techno, House, Glitch, IDM, Trip-Hop, Chip Music, Downtempo, Psych-Folk, Singer-Songwriter, Hip-Hop, Soundtrack, Ambient.

Discussion and Conclusion

- There are more than 32 genres in our data set, but the rest of the genres can not accurately predict due to insufficient numbers of tracks.
- Because of a large amount of audio features(518), the genres can be more precise with more variety of tracks.

Future Work

- We can includes more type of genres to neural network by enhancing the data set.
- Let our recommendation learn user's preference by the feedback of songs it recommends to them.

References

FMA: A Dataset For Music Analysis
Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, Xavier Bresson. International Society for Music Information Retrieval Conference (ISMIR), 2017.

stanCode
標準程式教育機構