Project PTNM

Personal Taste Navigation Model

Team : Find a Secret Key  
Dongguk Univ.

# Overview ::

Recently, hybrid recommendation systems are widely used to recommend content using a combination of Collaborative Filtering (CF) and Item Based Recommendation.

For example, "Spotify" provides hybrid recommendation service using three models, and is provided as an Open API.   
 This is the recommended way to do Spotify.  
- Collaborative Filtering Model  
- NLP through tags that characterize the music source  
- Classification of new songs via CNN

It is true that these models provide a sufficiently high level of satisfaction. However, as time goes by, users become less satisfied with receiving only similar music recommendations.

Therefore, we aimed to create a model that would suit users' tastes and recommend new music that I've never experienced before.

PTNM started with an idea from CNN. CNN is the extraction of frequency features through the Convolution Layer and classification or prediction through the Neural Network.

PTNM used vectors that extracted eight features instead of the Convolution Layer, and for ways to reflect user evaluations of frequency features on behalf of the Neural Network, see Page Rank Algorithm. Based on user evaluation, weights are obtained for frequency characteristics and then re-recommended by ranking them.

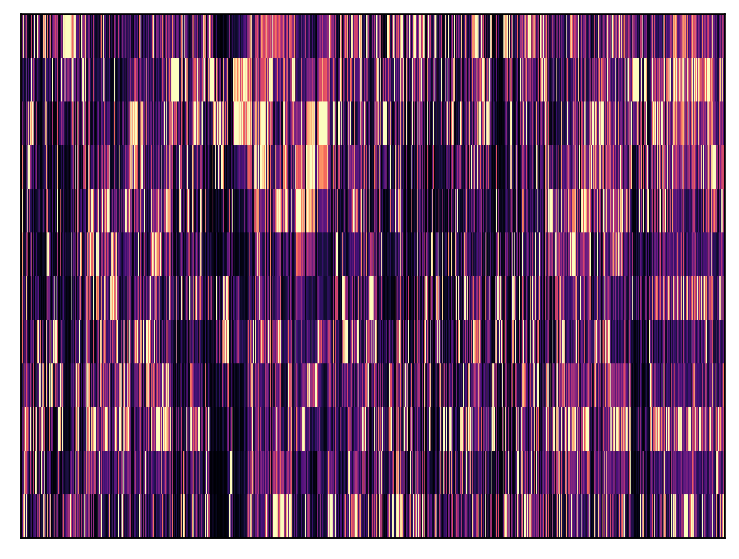
# Feature Extraction ::

Characteristic Extraction: Extract eight characteristics from the sound source. It is similar to the role of Convolution Layer on CNN.

Usually, these resulting images are entered into CNN. However, we compared similarities with sample groups through histogram similarity analysis. These results are summarized into one axis for each feature.

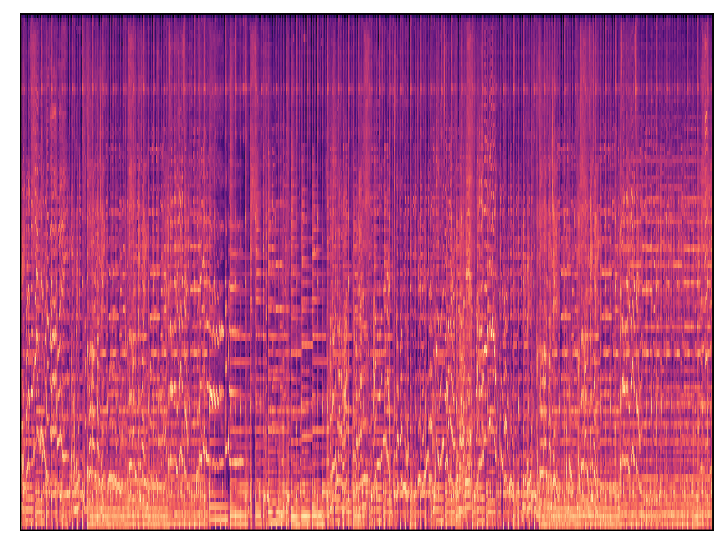
Frequency feature extraction mainly used the 'Librosa' package.

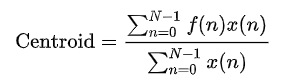
* **Chromagram** : A chromogram is a frequency power by dividing the frequency by 12 bin. In music, the octave-spaced notes are perceived to be similar, so knowing the Chroma distribution without absolute frequency can also provide useful information.  
  <https://labrosa.ee.columbia.edu/matlab/chroma-ansyn/>



Funkadelic - Can’t Stand the Strain / Chromagram

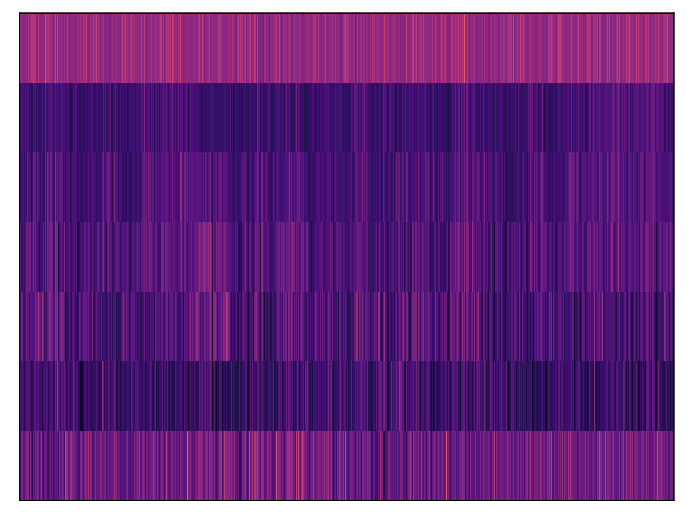
* **Mel-spectrogram** : Mel-scaled spectrogram

  
Funkadelic-Can’t Stand the Strain / Mel-spectrogram

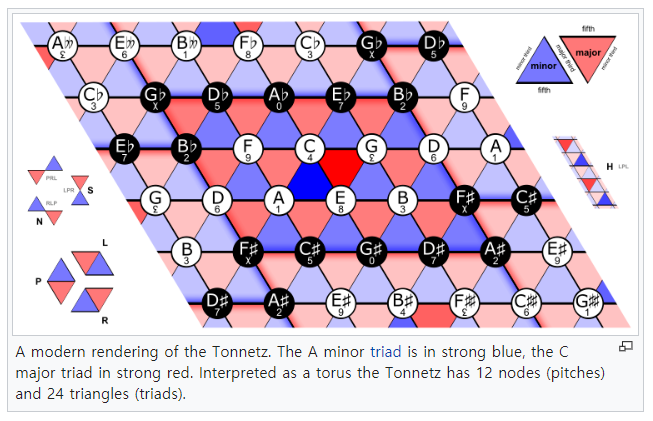
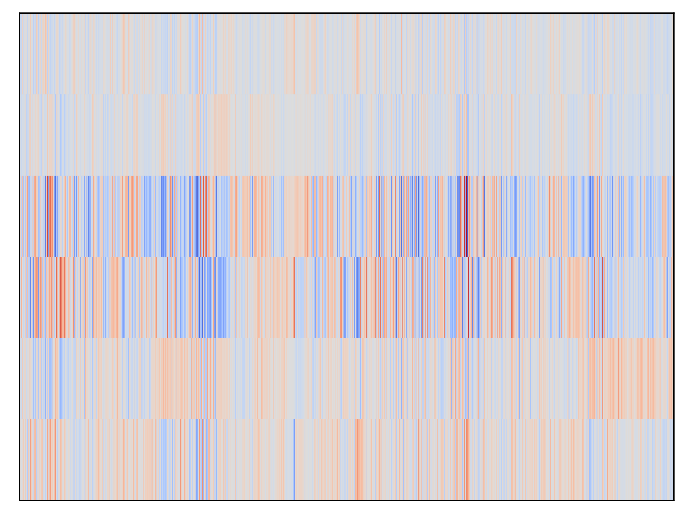
* **Spectral\_centroid** : It is related to a Brightness of sound, indicating where the center of the current frequency spectrum is.  
     
     
  Funkadelic - Can’t Stand the Strain / Spectral Centroid
* **Spectral bandwidth** : Compute p’th-order spectral bandwidth.

(sum\_k S[k, t] \* (freq[k, t] - centroid[t])\*\*p)\*\*(1/p)  
   
Funkadelic-Can’t Stand the Strain / Spectral Bandwidth

* **Spectral contrast** : It is a method of spectral contrast based on octave.

   
Funkadelic-Can’t Stand The Strain / Spectral Contrast

* **Tonnetz** : the Tonnetz (German: tone-network) is a conceptual lattice diagram representing tonal space first described by Leonhard Euler in 1739. It expresses the relationship between tonality and tonal space in a graphic way.  
  <https://en.wikipedia.org/wiki/Tonnetz>

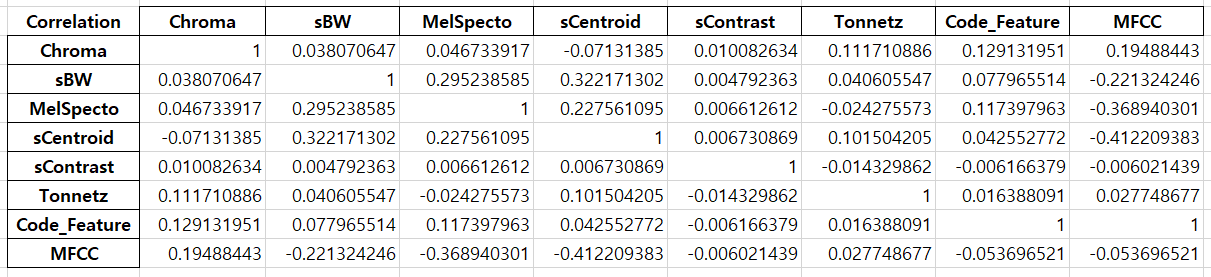
Funkadelic-Can’t Stand the Strain / Tonnetz

* **MFCC** : A 12-dimensional vector is extracted and the cos similarity with the unique vector is analyzed for each dimension.
* **Melody Feature** : First, using 'Melodia' to separate the vocal notes from the music, then convert them into musical notes to create a 12-dimensional vectorization. And organized by analyzing the cos similarity between the unique vectors for each dimension.

As shown above, each feature vector extracted has 6 to 12 dimensions. Thus, it takes the process of reducing dimensions to make it one axis.

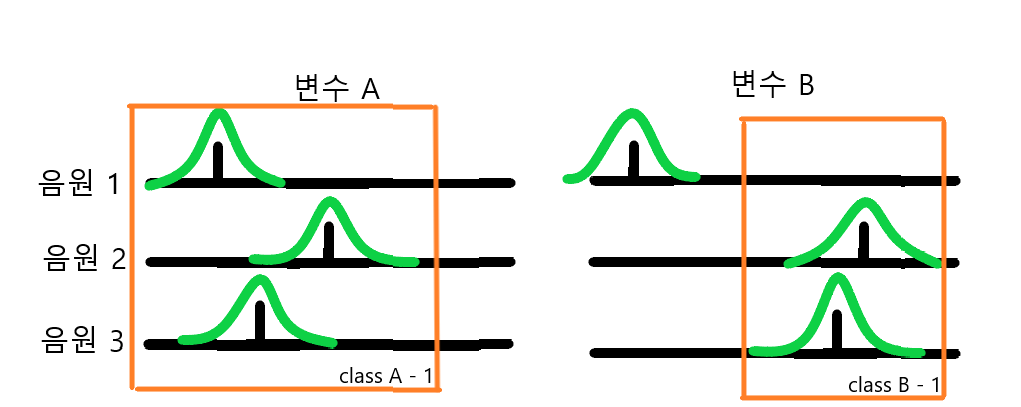
Because the ratio of similarity to each group is important, we normalized the feature vector, and then arranged into one axis using Arithmetic Coding.

- Correlation of the final organized features



# Recommendation Algorithm ::

1. Enter a song that the user previously preferred.
2. Find the variable value for the song.



Assuming that the user's preference probability around the selected music has a Gaussian normal distribution, if the probability range of other songs overlaps within the range of 80% probability, it is grouped into one class to indicate the user's preference range for that variable.

1. Calculate user dependencies for each variable and weights for each class.
2. Calculate total weights.
3. Recommend the top five songs and apply the feedback back to the variable axis space.

# Conclusion ::

The test was conducted with a total of 5,602 songs. Overall, we were able to achieve satisfactory results. About three songs were recommended very similar, and two songs were recommended in a new genre. Sometimes songs that are completely opposite to taste are recommended, but as feedback is applied, we have even confirmed that we are finding the right direction of recommendation again.

PTNM can be used in various ways for music recommendations. In case of weather or moody music recommendations, you can continue to recommend new songs by entering some of the songs that you previously preferred. Based on the songs I've already listened to, rather than the songs selected by other people, I can get my own customized service depending on the weather.

Finally, the biggest advantage of PTNM is that the limits of taste are likely to be extended. Traditional music classification methods were all based on genre, as PTNM listed the songs by frequency features. Even similar music can have access to different genres of music, and this process allows users to access more diverse songs.

# Reference ::

‘How Does Spotify Know You So Well?’ Sohia Ciocca, Medium /

<https://medium.com/s/story/spotifys-discover-weekly-how-machine-learning-finds-your-new-music-19a41ab76efe>

Librosa Documentation /

<https://librosa.github.io/librosa/feature.html>

Chromagram /   
Ellis, Daniel P.W. “Chroma feature analysis and synthesis” 2007/04/21

Spectral Centroid /  
<https://en.wikipedia.org/wiki/Spectral_centroid>

Spectral Contrast /  
Jiang, Dan-Ning, Lie Lu, Hong-Jiang Zhang, Jian-Hua Tao, and Lian-Hong Cai. “Music type classification by spectral contrast feature.” In Multimedia and Expo, 2002. ICME‘02. Proceedings. 2002 IEEE International Conference on, vol. 1, pp. 113-116. IEEE, 2002.

Tonnetz /  
Harte, C., Sandler, M., & Gasser, M. (2006). “Detecting Harmonic Change in Musical Audio.” In Proceedings of the 1st ACM Workshop on Audio and Music Computing Multimedia (pp. 21-26). Santa Barbara, CA, USA: ACM Press. doi:10.1145/1178723.1178727.

And ‘<https://en.wikipedia.org/wiki/Tonnetz>’

Page Rank Algorithm /

Sergey Brin and Lawrence Page “The Anatomy of a Large-Scale Hypertextual Web Search Engine”

결정을 돕는 추천 알고리즘/

<https://blog.naver.com/happyrachy/221292844469>