International Journal of Fuzzy Logic and Intelligent Systems, vol. 10, no. 1, March 2010, pp. 19-24

An Auto Playlist Generation System with One Seed Song

Sung-Woo Bang, Hye-Wuk Jung, Jaekwang Kim and Jee-Hyong Lee*

Department of Electrical and Computer Engineering, Sungkyunkwan University
300 Cheoncheon-dong, Jangan-ku, Suwon, Gyunggi-do 440-746, Korea
Tel : +82-31-290-7987, Fax : +82-31-299-4637, E-mail : sistone81@skku.edu

Abstract

The rise of music resources has led to a parallel rise in the need to manage thousands of songs on user devices. So users have a tendency to build playlist for manage songs. However the manual selection of songs for creating playlist is a troublesome work. This paper proposes an auto playlist generation system considering user context of use and preferences. This system has two separated systems; 1) the mood and emotion classification system and 2) the music recommendation system. Firstly, users need to choose just one seed song for reflecting their context of use. Then system recommends candidate song list before the current song ends in order to fill up user playlist. User also can remove unsatisfied songs from the recommended song list to adapt the user preference model on the system for the next song list. The generated playlists show well defined mood and emotion of music and provide songs that the preference of the current user is reflected.

Key Words : Playlist generator, Recommendation, Intelligent System, Content-based analysis

1. Introduction

The digital music resources have been extremely growing in the digital world. Concurrently, the numbers of online music service providers are increasing with the explosive popularity of music recommendation systems. Also as there are thousands of songs waiting to be selected by users on servers or user devices, more efficient methods are needed to handle them. So users tend to build playlists of songs to either manage their music databases or allow using it for various occasions. Generally the playlists are filled with songs that users are desired, and selected as their context of use. Namely, users create several playlists by selecting songs on their device, and decide to choose one of these as they want. For instance, when users are working out, users may prefer to listen to the exciting and fast songs. Then users choose a playlist which contains such mood songs. When studying in library, additionally, a user will choose a playlist composed with relatively quiet songs for their concentration.

However users are dealing with the manual method for song selection in order to make a playlist. Users choose and assemble songs for adequate playlists of music compositions from music database. This process is cumbersome task with a time consuming and difficult activity. To solve these problems, a number of proposals have been made. Most of them approaches proposed the systems that created playlists automatically, according to some chosen criteria. John C. Platt offers a system for automatically generating music playlists by learning with a Gaussian process [1]. This system is shown to be more effective at predicting users' playlists than a reasonable designed system, but it does not care about user wanted musical mood and emotion.

Also auto generated playlist must be a collection of songs which a user wants to listen to. Users are usually select songs in the several musical genres. It means auto generated playlists need to care about not only context of use but also user musical preference like genres. To generate a playlist automatically with reflecting musical mood, emotion and user preference, one possibility of approach is with content-based music analysis. This music analysis method is concerning only music signals without any metadata of music in the recommendation process. With content-based music analysis, Deliège and Pedersen suggested a playlist management method using fuzzy lists [2]. But this management method does not consider the musical mood and emotion and also has a cold start problem that users are asked to build a playlist.

This paper proposes an auto playlist generator system with considering musical mood, emotion and user preference. Users need to choose just one seed song for reflecting their context of use. Then system recommends a candidate song list before the current song ends in order to fill up the user playlist. Users also can remove the unsatisfied songs from the recommended list then the system adapts user preference on system for the next one. With this operation is processed recursively, users can have a better recommendation result applied user preference. The generated playlists show well defined mood and emotion of music and provide songs that user preference is reflected. This proposed system is developed to achieve four goals:

1. To overcome cold start problems in generating playlists
2. To consider context of use (mood and emotion)
3. To reflect user preference
4. To generate playlist automatically

Manuscript received Nov. 27, 2009; revised Jan. 10, 2010.
* Corresponding author: jhlee@ece.skku.ac.kr
This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2009-0075109).

In the rest of this paper, we present our separated two systems; 1) mood and emotion classification system and 2) music recommendation system. After discussing related work and features selection in the next section, we describe our system for song selection approach. We then discuss our experiments and the result of generated playlists in Section 4 and 5.

2. Related Work

2.1 Feature selection

For comparing and classifying songs with content-based music analysis, the system extracts psychoacoustic music features. In our system, we extract a wide range of signal properties can be extracted for individual windows of song. Each song features can be extracted by calculating averages and standard deviations from these windows subsequently. We extract average and standard deviation values of several features, respectively; spectral centroid, spectral rolloff point, spectral flux, compactness, spectral variability, root mean square, fraction of low energy windows, zero crossings, strongest beat, beat sum, strength of strongest beat and LPC. These features have been found important for music signal processing and usually selected to find particularly related to mood and emotion of music perception.

- Spectral centroid: This is the mean of the short time Fourier amplitude spectrum. It gives an indication of how "bright" a musical piece is.
- Spectral rolloff Point: This is the point where frequencies are getting smaller in amplitude and gives the shape of the spectrum. 95% of the total spectrum is within this rage.
- Spectral flux: This indicates how much the spectral shape changes form frame to frame.
- Compactness: This is a measure of the noisiness of a signal.
- Spectral variability: This is a measure of the standard deviation of a signal's magnitude spectrum.
- Root mean square (RMS): This is a good measure of the power of a signal.
- Fraction of low energy windows: This is a good measure of how much of a signal is quiet relative to the rest of a signal.
- Zero crossings: This features gives the number of times the signal crosses the zero line. It is a good indicator of the amount of noise.
- Strongest beat: This feature gives the strongest beat in a signal.
- Beat sum: This is a good measure of how important a role regular beats play in a piece of music.
- Strength of strongest beat: This is a measure of how strong the strongest beat is compared to other possible beats.
- LPC: This feature is one of the most useful methods for encoding good quality sound at a low bit rate.

See jAudio Tool Websites for more details [7]. We also extract Mel-Frequency Cepstral Coefficients (MFCC) in order to classify the musical genres. This feature will be explained more specifically in later section.

2.2 Mood and emotion recognition

Most previous works on mood and emotion of music recognition system categorize mood and emotion into a number of classes and apply the standard pattern recognition procedure to train a classification model. Typically, mood and emotion classes can be divided into the four quadrants in Thayer's energy-stress mood and emotion plane [3] see Fig. 1. Along the horizontal axis is the amount of stress in measured and along the vertical axis is the amount of energy. In music, we can think of energy as the volume or the intensity of sound. And stress can be translated as "busy to do many things", so the difference in beat and tempo would be a good mapping.

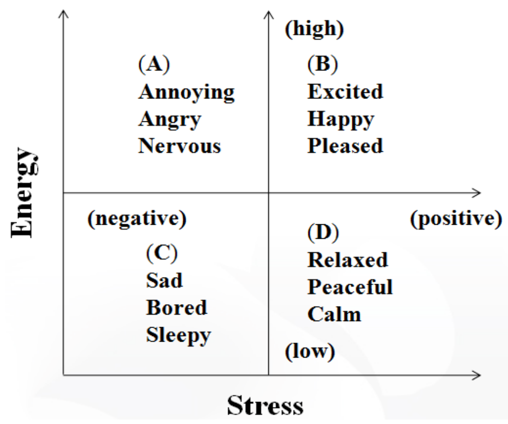


Fig. 1 Thayer's model of mood and emotion

Each quadrant A to D has song's mood; A {aggressive, angry, fiery, hostile, malevolent, visceral, volatile}, B {lively, bright, humorous, happy, cheerful, exciting, joyous, playful}, C {sad, melancholy, gloomy, ominous}, D {dreamy, smooth, peaceful, plaintive, relaxed, sentimental}. We collected songs in these categorized quadrants.

2.3 Musical genre classification

A traditional approach to determine musical genre is to use classical relational models such as Hidden Markov Model (HMM) and Gaussian Mixture Model (GMM). In this paper, MFCC's are extracted from song in features extraction part in order to build a music model. MFCCs are representative properties of sound waves usually used speech recognition and music analysis researches. After MFCCs are extracted, the HMM is employed to build a music model for evaluating the similarities among pieces, because it can model temporal patterns [4]. Many researches are suggesting HMMs to evaluate similarities between music. Xi. Shao et al. proposes a music genre classification using MFCC's and HMMs [5].

3. Proposed System

The proposed system consists of separated two main systems; 1) mood and emotion classification system and 2) recommendation system. Mood and emotion classification system decides song's mood class when users choose a song; it takes advantage of creating playlist. Users do not need to fill up their playlist at the begging of enjoyment music. The recommendation system then selects 3 songs in which are same class with a selected song until current song is end. Users are also able to remove unsatisfied songs in order to adapt their preferences to the recommendation system. Both systems adopt content-base music analysis technique. The diagram of system is shown in Fig. 2.

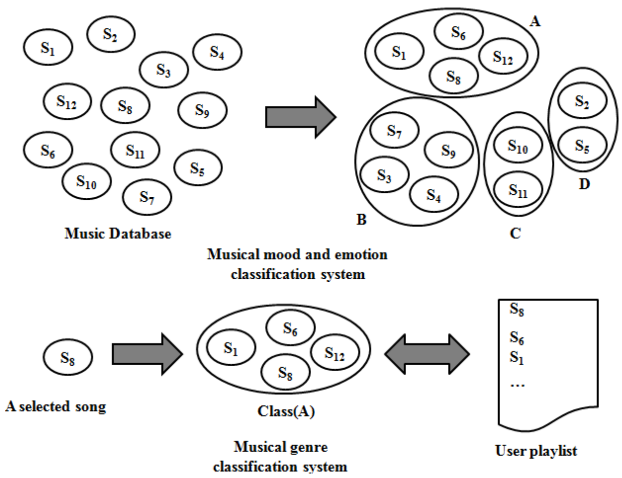


Fig. 2 The diagram of system

3.1 Mood and emotion classification system

The k-NN classification algorithm classifies an object based on proximity to neighboring objects in a multidimensional space. The algorithm is trained with a corpus of approximately 241 songs. The result of the k-NN training is a multidimensional feature space that is divided into specific regions corresponding to the four labeled classes of the training set. The number of nearest neighbor, k, used in this system is determined through cross-validation of the data, in which an initial subset of the data is analyzed and later compared with the analysis of subsequent subset. Through this process it was determined that the best number of nearest neighbors for this is k = 5. When a new song is added into the system its class is unknown, then the multidimensional space of its position is compared to that of its nearest neighbors using the measure of Euclidean distance, and the song is assigned a class based on its proximity to its 5 nearest neighbors.

3.2 Music recommendation system

Music recommendation system has two modules, musical genre classification module and recommendation module. To analyze musical genre songs, abstracting a model from the data is necessary. To create a model of a song, our system extracts a sequence of MFCCs from the songs and builds an HMM with the sequence. Then the similarity between songs is evaluated by comparing the Markov models [6]. With HMM, the song similarity was defined as shown in Eq. (1).

$$S_{ij} = \frac{\frac{1}{N_i}\left[\log p(O_i \mid H_i) - \log P(O_j \mid H_i)\right]}{2} + \frac{\frac{1}{N_j}\left[\log P(O_j \mid H_j) - \log P(O_i \mid H_j)\right]}{2} \qquad (1)$$

In (1), S_{ij} is the similarity between the i th and the j th song, Hi and H_j are the HMMs of the i th and the j th song respectively, O_i and O_j are the most likely observation sequences of H_i and H_j respectively, and N_i and N_j are the lengths of O_i and O_j respectively.

The process in recommendation module is analyzing user playlist. This process creates groups of songs in the playlist of user. The songs are grouped according to the similarity which is meaning how similar genre is. When system creates recommendation playlist, it uses grouping result in order to reflect user genre preference. In analyzing user playlist process, the system creates groups of pieces in the playlist of user. The pieces are grouped according to the similarity. For the grouping, the system generates a similarity graph that represents the similarity between pieces in the users' playlist. Then, the system clusters pieces using the MCL graph clustering method [7]. A music similarity graph represents a similarity relationship between pieces in the playlist. The system clusters the pieces that a user has listened to with the similarity graph. Since a user may listen to different types of music, similar pieces need to be grouped together in order to enhance the accuracy of the analysis. In this paper MCL graph clustering algorithm was applied to efficiently create music groups. The algorithm was proposed by S. V. Dongen and is a type of graph clustering method utilizing the intensity of edges and the number of connective edges between nodes [8].

4. Experimental Setup

We collected 241 songs and each experimental song was first down-sampled into a uniform format 16000 Hz, 16 bits, mono channel, wav file. Then 51 acoustical features were extracted from jAudio Tool [9]. jAudio is a famous for feature extraction tool calculating features from an audio signal. The specific description about jAudio tool is introduced in jAudio website. After features were extracted, we calculated statistical coefficients and normalized these as preprocess using (2), (3):

$$StatisticalCoefficient(SC_{kl}) = \frac{\sigma(F_{kl})}{ave(F_{kl})} \qquad (2)$$

$$Nomalization(SC_{kl}) = \frac{SC_{kl}}{\max(SC_l)} \qquad (3)$$

SC_{kl} is the statistical coefficient, F_{kl} is the feature value, k is a song number and l is a feature number. Feature values are normalized between 0 and 1 for experiment. Also songs' tag information such as genre and mood were collected from All-Music website which provides a music retrieval service.

5. Results and Evaluation

We used k-NN algorithm to decide song's mood. Table 1, k-NN classifier is used with k from 1 to 10 with 145 songs as training set and 96 songs as validation data.

Table 1. Best k in k-NN

Value of k	%Error Training	%Error Validation
1	0.00	39.95
2	19.60	36.05
3	27.39	34.42
4	25.88	35.05
5	**22.37**	**32.29**
6	25.88	36.68
7	26.63	35.05
8	31.40	36.05
9	25.88	32.79
10	28.14	34.42

The most correct recognizable emotional space is quadrant A and the poorest space is quadrant D in Table 2. The highest result in k-NN classifier is achieved when k is 5. The recognition error rate of validation is 32.29% with best k=5. Songs are also classified into the following 6 genres: electronic, metal, jazz, pop, rock and others. Table 3 shows user genre preference and users' playlist analysis result through our recommendation system. The recommended result may explain that our system creates well defined musical genre and provides meaningful recommendation to user. For example, in case of electronic user, our system recommended mainly electronic songs with 69.6% accuracy and jazz users can fill up their playlist with jazz songs which has 73.3% accuracy.

Table 2. Error Report in k-NN with k=5

Error Report			
Class	#Cases	#Errors	%Error
A	26	4	15.38
B	21	7	33.33
C	27	11	40.74
D	22	9	40.91
Overall	96	31	**32.29**

Table 3. Result of User Playlist

Playlist	Genre (%)					
	Elec.	Metal	Jazz	Pop	Rock	Others
User Playlist	87.5	72.5	84.5	84.0	83.5	-
Recommended on Playlist	69.6	52.8	73.3	55.1	63.2	-

In the following example is particular playlist generating processes which is accomplished by a rock user.

Step 1 (user selects one seed song);

	Song Name	Artist	Class	Genre
User Playlist	Bring The Noise	Anthrax	A	Metal

A user selects one seed song in order to reflect his or her context of use from music database in step 1.

Step 2 (system recommends 3 songs in class A);

	Song Name	Artist	Class	Genre
System Playlist	Bring The Pain	Method Man	A	Pop
	Fade To Black	Metallica	A	Metal
	My Generation	The Who	A	Rock

The system then recommends 3 songs from user database randomly. A user removes 2 songs not to prefer genre, pop and metal.

Step 3 (user feedback and song recommendation);

	Song Name	Artist	Class	Genre
User Playlist	Bring The Noise	Anthrax	A	Metal
	My Generation	The Who	A	Rock

Now user playlist contains 2 songs.

Step 4 (user feedback and song recommendation);

	Song Name	Artist	Class	Genre
System Playlist	Seasons in the abyss	Slayer	A	Rock
	In Bloom	Nirvana	A	Rock
	Personality Crisis	The New York Dolls	A	Metal

Next the system recommends 3 songs again based on user playlist analysis. Also a user removes metal song from the playlist.

Step 5 (user feedback and song recommendation);

	Song Name	Artist	Class	Genre
User Playlist	Bring The Noise	Anthrax	A	Metal
	My Generation	The Who	A	Rock
	Seasons in the abyss	Slayer	A	Rock
	In Bloom	Nirvana	A	Rock
	Song Name	**Artist**	**Class**	**Genre**
System Playlist	Welcome to the Jungle	Gun's and Roses	A	Rock
	Fight the Power	Public Enemy	A	Rock
	DU HAST	Rammstein	A	Metal

Step 6 (user feedback and song recommendation);

	Song Name	Artist	Class	Genre
User Playlist	Bring The Noise	Anthrax	A	Metal
	My Generation	The Who	A	Rock
	Seasons in the abyss	Slayer	A	Rock
	In Bloom	Nirvana	A	Rock
	Welcome to the Jungle	Gun's and Roses	A	Rock
	Fight the Power	Public Enemy	A	Rock
	DU HAST	Rammstein	A	Metal
	Song Name	**Artist**	**Class**	**Genre**
System Playlist	Paradise City	Gun's and Roses	A	Rock
	Street Fighting Man	Public Enemy	A	Rock
	Bring the Noise	Public Enemy	A	Rock

Through step 5 and step 6, a user is able to adopt his or her preference such as musical genre. Seeing the last system playlist, the system recommends three rock songs for a rock user. This meaningful recommendation shows that a user can fill up playlist with desired musical genre songs.

Finally, a user playlist is completed with rock songs that a user wants to listen to.

User playlist;

	Song Name	Artist	Class	Genre
User Playlist	Bring The Noise	Anthrax	A	Metal
	My Generation	The Who	A	Rock
	Seasons in the abyss	Slayer	A	Rock
	In Bloom	Nirvana	A	Rock
	Welcome to the Jungle	Gun's and Roses	A	Rock
	Fight the Power	Public Enemy	A	Rock
	DU HAST	Rammstein	A	Metal
	Paradise City	Gun's and Roses	A	Rock
	Street Fighting Man	Public Enemy	A	Rock

6. Conclusion and Future Work

In this paper we proposed an auto-playlist generator considering user context of use and musical preferences, and how it can be helped to generate playlist. We presented a reasonable result of musical mood and emotion classification to reflect user context of use, and the recommended playlist showed high accuracy of genre preference of user. In contrast to the previous playlist generators which take content-based approach but had several inconvenient practical uses, the proposed system generated playlist automatically by selecting one seed song.

In the future, we will combine this system with mobile devices or web services. This may be more useful to enjoy music anytime and anywhere. Also the system will be able to recognize user's external information such as time, position and context. Then the context of use, we define in this paper, is recognized automatically. It means users do not need to select even one song.

References

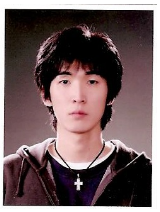
[1] John C. Platt, C. Burges, S. Swenson, C. Weare and A. Zheng, "Learning a Gaussian Process Prior for Automatically Generating Music Playlists," *In Proc. NIPS*, vol. 14, pp. 1425-1423, 2002.

[2] F. Deli`ege, T. B. Pedersen, "Using Fuzzy Lists for Playlist Management," *In Proc. MMM,* pp. 198-209, 2007.

[3] R. E. Thayer, *The Biopsychology of Mood and Arousal*, Oxford University Press, 1989.

[4] L. R. Rabiner, *Fundamentals of speech recognition*, Prentice-Hall, 1993.

[5] Xi. Shao, C. Xu and M. S. Kankanhalli, “Unsupervised Classification of Music Genre Using Hidden Markov Model,” *IEEE International Conference on Multimedia and Expo*, vol. 3, pp. 2023-2026, 2004.
[6] K. Kim, D. Lee, T. Yoon and J. Lee, “A music Recommendation System based on Preference Analysis,” *In Proc. ICADIWT*, pp. 102- 106, 2008.
[7] J. J. Aucouturier and Francois Pachet, “Music Similarity Measures: What’s the Use?,” *International Symposium on Music Information Retrieval*, pp. 157-163, 2002.
[8] S. V. Dongen, “Graph Clustering by Flow Simulation,” *PhD thesis, University of Utrecht*, 2000.
[9] http://jaudio.sourceforge.net

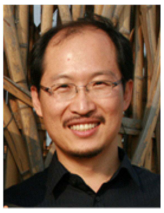


Sung-Woo Bang
Sung-Woo Bang received his B.S. in the Department of Information and Communication Engineering from Sungkyunkwan University, Suwon, Korea, in 2009. He is currently Master course in the Department of Electrical and Computer Engineering, Sungkyunkwan University. His research interests include intelligence system, user modeling and music recommendation, etc.

Hye-Wuk Jung
Hye-Wuk Jung received her B.S. in the Department of Information and Communications Engineering from Hansung University, Seoul, Korea, in 1999. She also received her M.S. degree in the Department of Information and Communication Engineering from Sungkyunkwan University, Suwon, Korea, in 2005. She is currently a Ph.D. Student in the Department of Electrical and Computer Engineering, Sungkyunkwan University. Her research interests include finger printing, intelligent system, game A. I. and information security, etc.

Jaekwang Kim
He received a B.S., M.S. degree from Sungkyunkwan University at Suwon, Korea in 2004, and 2006 respectively. Jaewkang’s research interests include networks, security, and intelligent system. He is currently a Ph.D. student at the Department of Electrical and Computer Engineering, Sungkyunkwan University, and has also worked as a researcher of the Institute of Information and Communications Technology, Sungkyunkwan University.

Mr. Kim is a member of the Korean Institute Intelligent Systems and a student member of the IEEE. Mr. Kim was a pioneer of the system consultant group at Sungkyunkwan University School of Information and Communication Engineering.



Jee-Hyong Lee
Jee-Hyong Lee received his B.S., M.S. and Ph.D. degrees in Computer Science from KAIST (Korea Advanced Institute of Science and Engineering), Taejon, Korea, in 1993, 1995 and 1999 respectively. From 2000 to 2002, he was an International Fellow of SRI International, Menlo Park, CA. He served as the Publicity Chair of FUZZ-IEEE in 2009 and the Treasurer of Ubicomp in 2008. He is currently an Associate Professor in the School of Information and Communication Engineering, Sungkyunkwan University, Suwon, Korea. His areas of research include data mining, user modeling and intelligent information systems.