An Approach to Analyzing Correlation between Songs/Artists Using iTMS Playlists

Dou, Yufeng
Graduate School of Information Science and Electrical Engineering, Kyushu University : Graduate Student

Ito, Eisuke
Computing and Communications Center, Kyushu University : Associate Professor : Information Science

Ikeda, Daisuke
Kyushu University Library : Associate Professor : Information Science

Hirokawa, Sachio
Computing and Communications Center, Kyushu University : Professor : Information Science

出版情報：Proceeding of the International Conference on Intelligent Agents, Web Technology andInternet Commerce (IAWTIC '2005), 2005-12
バージョン：
権利関係：

An Approach to Analyzing Correlation between Songs/Artists Using iTMS Playlists

Yufeng Dou Eisuke Itoh Sachio Hirokawa Daisuke Ikeda
Grad. School of ISEE, Compuging and Communications Center, Library,
Kyushu University.
Hakozaki 6-10-1, Higashi-ku, Fukuoka, 812-8581, Japan.
{itou@cc, hirokawa@cc, g-tou@i, daisuke@lib}.kyushu-u.ac.jp

Abstract

Digital audio devices have been changing music entertainment environment. Those devices are bundled with music jukebox software, such as Apple's iTunes, Sony's CONNECT player. Jukebox software not only enables us to recode, play, search, purchase music on PC, but also to manage playlist. Anybody can make his/her own playlists, and play music according to the list.

In this paper, we focus on iTMS (iTunse Music Stroe) playlists and use them as the data minig resources for a music recommendation system, then developing correlation measuring methods. We have retrieved about 13,000 playlists, and analyzed the frequency of artists/songs and the co-occurrence of artists/songs in the playlists. Through the result data, we found out that all graphs we drew are follow the Zipf's law. Furthermore, we have analyzed the hierarchical relation between songs according to their popularity. We proposed a basic idea of popularity measuring method.

1. Introduction

With the expansion of the Internet, the web pages which include listing materials are increasing too. Using Web information extraction technology [2, 6], we can extract list data from such pages. Such a list include items, such as products, pieces of music, books, and so on. Also as web communities are expanding by day and day, list data, such as Playlists of songs, favorites in SNS (Social Networking Site), Listmania in amazon.com, etc, are also increasing rapidly, and people can share their interests with each other.

Apple Computer has made a big success in digital audio field. They have many big user communities. Much research focused on their communities has been conducted up to now. These kinds of research can apply to many fields, such as "social practices". In this paper, we focused on the playlists from Apple iTMS. The playlists we refer to are lists of music that are created in a digital jukebox software. At iTMS, there is a user community and every registered user can publish his/her own playlists to the iTMS on the web. Thus the users can share their music taste by publishing playlists, and can also rate other playlists. There are also a charts in iTMS, so users can see a high rating playlist, to discover new music. iTMS has a large amount of playlist data, with 332,560 playlists as of 26 th Aug. This is a big reason why we chose iTunes playlists for our analysis data. We retrieved nearly 13,000 playlists from iMix (the playlist publishing site in iTMS) for our research.

We can expect that songs or artists in a playlist have some close relationship. Thus if we can measure the correlations between songs/artists, we can use them for a music recommendation system. These developed technologies will be also applicable to other fields, such as market analysis, decision making, and so on.

For building a recommendation system, much research has been conducted in the field of Information Retrieval since early times. Basically, a recommendation system is either content-based or collaborative filtering based. The former type is used to define the similarity of items from the contents. Miura *et. al.* takes an acoustic based approach and they have proposed a music raw data analyzing system [3]. The latter type uses subjective information. Collaborative filtering algorithms function in a manner such that administrator has user A to create his/her own profiles, then recording to the similarity of the profiles to search for another user B, and then recommend user B's favorite items. Our approach is similar to the collaborative filtering approach.

The question arises: to conduct a recommendation, users must create detailed information for their profiles. For creating an accurate recommendation, more categories in the profiles is cosidered better. Thus profile creating would be

hard work for users. Concerning this question, some attempts have been made to use information from the web or log data [1], but data from the web are not made from a common set, so it is meaningless to measure their distance. Thus we attempt to analyze the items which should be recommended, not considering the similarities of the profiles.

As the first step for analyzing the correlations between pieces of music or artists, in this paper, we measured the frequency of songs, artist, and genre listed in playlists. We retrieved nearly 13,000 playlists from the site iMix. We then analyzed the frequency of artists/songs in playlists. Furthermore, we analyzed the hierarchical relation between songs according to their popularity.

Schedl *et. al.* has proposed a web-based approach for music simiality calculation algorithm based on co-occurances in [5]. They have made a complete distance matrix for 224 artists and 14 genres. Complexity is quadratically proportional number of artists. They also developed a visualization system in [4].

Our approach is similar with Schedl's method in [5]. Their resources were heterogeneous web documents, and the amount of documents is not excessive. Our resource is completely homogeneous playlists, but the amout of playlists is massive.

2. Collection of Playlists

Our target playlists are Apple's iTunes playlist, because Apple's iPod is most major, and iMix offers huge ammount of playlists, A playlist is created and submitted by an iTunes user. Every iTunes users can create own playlist by drag and drop operation, and he/she is able to submit own playlist to iMix.

iMix offers 332,560 playlists as of 26th Aug., 2005. One can download playlists using following operations on iTunes:

- Run iTunes.
- On iTunes, click [iTunes Music Store] $\rightarrow$ [iMix], then iMix front page will be displayed on iTunes.
- 30 playlists are avaliable on the front page.
- Select a playlist, and save it.

Each playlist is given a unique ID number. Right button click on playlist icon on iMix page, "copy URL" menu will be appeared. If you copy and paste the URL, you can get a URL string such as `http://phobos.apple.com/WebObjects/MZStore.woa/wa/viewPublishedPlaylist?id=471475` . Surfix part string is the ID of the playlist. "471475" is the ID, in this case.

MS-Windows user can type "`start` *PlaylistURL*" in MS-DOS prompt, then iTunes will run automatically, and the playlist will displayed on iTunes.

We tried to get 15,000 playlists for this reearch. At first, generate 15,000 ramdom integer between 1 to 300,000. Concatenate the Apple's playlist URL and the number, and get playlist one by one.

Finally, we got 13,480 playlists. In other words, 1,520 ID numbers are invalid. Some playlists may be removed by playlist creator or administrator of iMix. Because we randomly generated Playlist ID numbers, 1,520 invalid ID number are included.

3. Frequency Analysis

In this section, we describe some analysis of frequency of songs/artists that appear 13,480 playlists.

3.1. Number of songs in a playlist

At first, we analyzed number of songs in each playlist. How many songs does a playlist(PL for short) has. Fig.1 shows a rank–frequency plot of the number of songs in decreasing order. Both x and y axes are in log-scale. Only 2 PLs have 249 songs, which is the maximum number of songs among collected lists.

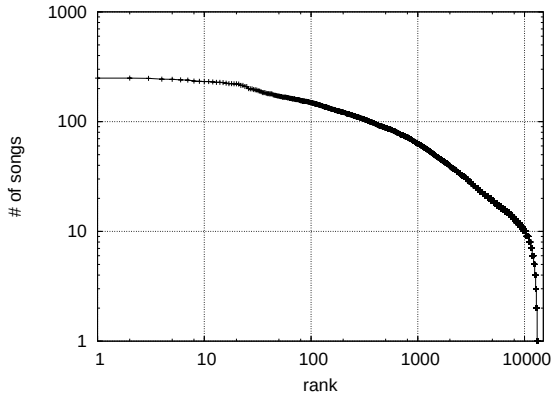

Figure 1. Number of songs in PL

For each positive number x, we plot the number of PL files which contain x songs (Fig.2). We find that most of playlists has 5 to 20 songs. Actually, the total number of PL files which have 5 to 20 songs is 7,919, and they account for 60% of all PL files. 12 songs are contained in 678 PLs, which is maximum. On average, a PL contain 24 songs.

When we try to define some relationships between songs in PL's for recommendation, the number of songs per PL will be an important factor. Based on the authors experience, we selected about 20 songs for one PL. If you make a playlist which has more than 40 songs, you may be irresponsible for song selection. And, it is easy to select only

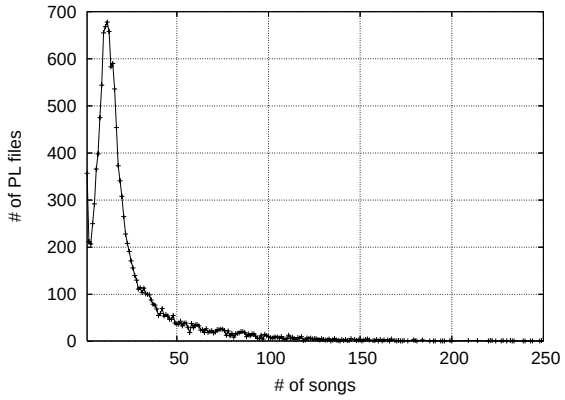

Figure 2. Number of PL files for each songs

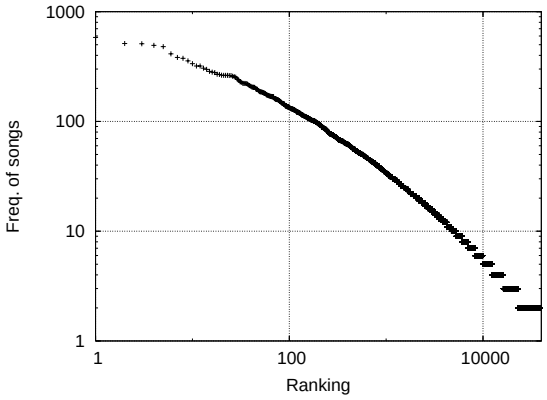

Figure 3. Freqency of songs

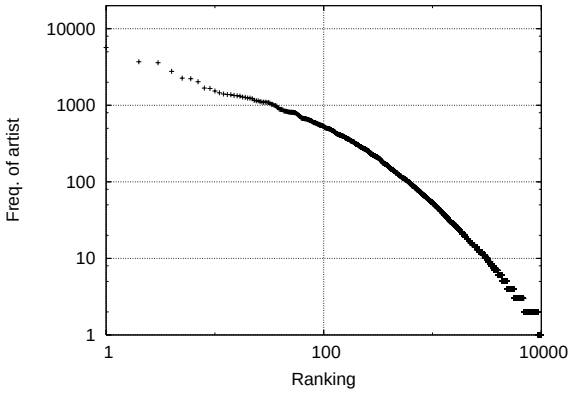

Figure 4. Freqency of artists

few songs. The number of songs in an album CD is about 10 to 20.

Therefore, in case of calculation of relationships between songs, weight of relationships between songs in PL's, which has about 5 to 20 songs, should be heavier than other.

3.2. Frequency of songs/artists

Secondly, we count frequency of songs/artists in all PL's. There are 320,345 song records in 13,480 PL files. The number of unique songs is 100,2004.

For information retrieval from documents, TF (term frequency) and DF (document frequency) are used for weighting of index words. Given a doument set $D = \{d_1, d_2, ...d_n\}$, TF value and DF value for a word w_i are defined as follows:

$$tf(w_i, d) = \text{frequency of term } w_i \text{ in document } d\ ,$$
$$df(w_i) = \text{number of documents which has term } w_i \text{ in } D.$$

We apply TF/DF to playlists, where, substitute d_j for a playlist p_j, and term w for a song/artist. We plot $tsf(s)$ for each song s in Fig.3. Definition of $tsf(s_i)$ is

$$tsf(s) = \sum_{j=1}^{n} tf(s, p_j).$$

In Fig.3, x and y are in logscale. It shows power law, so songs of playlists follow Zipf's law. Table.1 is top 20 songs for $tsf(s_i)$. Most appeared song is "American Idiot" by "Green Day".

Next, we count frequency of artists. The number of unique artist name is 18,582. Fig.4 shows plotted data of $tsf(a)$ for each artist a. Table.2 has top 20 aritst names. First artist name is "Green Day", and second is "Eminem".

3.3. Frequency of genres

Every song is given a genre text, so we analyzed about "genre". Total number of unique genre names is only 29. Compared with total number of songs/artists, there are only few genres in playlists.

Genre names and frequency are described in Table.3, and Fig.5 shows plot of the tatle.

3.4. Co-occurrence of artists

We analized co-occurrece of songs/artists in PLs. Fig.6 shows a rank–frequency plot of the number of co-occurred artists in decreasing order. Table.4 shows the top 20 artists pairs. We also analyzed pair of songs, but there is not enough space to show them. Most of pairs in Table.4 are major artists, who appeared in Table.2.

Rank	songName	genre	artistName	freq.
1	American Idiot	Alternative	Green Day	691
2	This Love	Rock	Maroon 5	629
3	The Reason	Alternative	Hoobastank	527
4	Boulevard of Broken Dreams	Alternative	Green Day	493
5	Are You Gonna Be My Girl	Rock	Jet	481
6	Hey Ya!	Hip-Hop/Rap	OutKast	450
7	Somebody Told Me	Alternative	The Killers	420
8	My Band	Hip-Hop/Rap	D12	398
9	Drop It Like It's Hot	Hip-Hop/Rap	Snoop Dogg & Pharrell Williams	386
10	Float On	Alternative	Modest Mouse	386
11	Let's Get It Started (Spike Mix) [Bonus Track]	Hip-Hop/Rap	Black Eyed Peas	376
12	Just Lose It	Hip-Hop/Rap	Eminem	341
13	Dirt Off Your Shoulder	Hip-Hop/Rap	Jay-Z	334
14	I Believe in a Thing Called Love	Rock	The Darkness	330
15	Holiday / Boulevard of Broken Dreams	Alternative	Green Day	320
16	Clocks	Alternative	Coldplay	320
17	Hey Mama	Hip-Hop/Rap	Black Eyed Peas	319
18	Harder to Breathe	Rock	Maroon 5	304
19	Welcome to My Life	Alternative	Simple Plan	286
20	Toxic	Pop	Britney Spears	280

Table 1. Top 20 songs

Rank	artistName	Freq.
1	Green Day	5683
2	Eminem	3704
3	U2	3600
4	Blink-182	2776
5	Maroon 5	2259
6	50 Cent	2225
7	Nirvana	2027
8	Coldplay	1678
9	OutKast	1659
10	Sum 41	1536
11	Guns N' Roses	1455
12	Black Eyed Peas	1404
13	Britney Spears	1385
14	Good Charlotte	1375
15	Yellowcard	1344
16	Avril Lavigne	1332
17	Beastie Boys	1319
18	R.E.M.	1278
19	Modest Mouse	1266
20	Jet	1241

Table 2. Top 20 artists

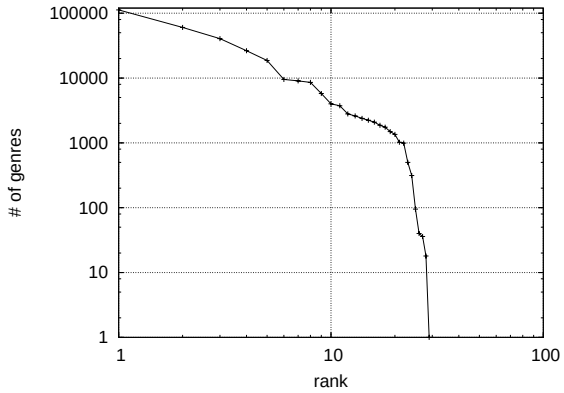

Figure 5. Freqency of genres

4. Relation Analysis of Songs

We proposed several method and showed analysis the popularity of songs. We plan to investigate the relationship between songs listed in the same playlists. In this section, we describe basic idea of relationship calculation algorithm.

We consider dependency of 2 songs. Popular song (or artist) are frequentry appear in many playlists, but, do two popular songs, such as "American Idiot (Green Day)" and "This Love (Maroon 5)", are co-occured frequently?

If song s_1 and s_2 are indevendent, then probability of co-occurrence in playlists will be $Pr(s_1) * Pr(s_2)$, where $Pr(s)$ is appearance probability of song s, and $Pr(s_1) = tsf(s_1)/\#\{PL\}$, If song s_1 depends on s_2, then probability of co-occurrence will not be $Pr(s_1) * Pr(s_2)$. So, check probability for all co-ocurred song pairs, then dependency of songs will be solved. Complexity is less than $O(n^2)$, where n is the number of unique songs.

We are implementing a system for such analysis. Fig.7 is a screen shot obtained from 138 playlists that contain

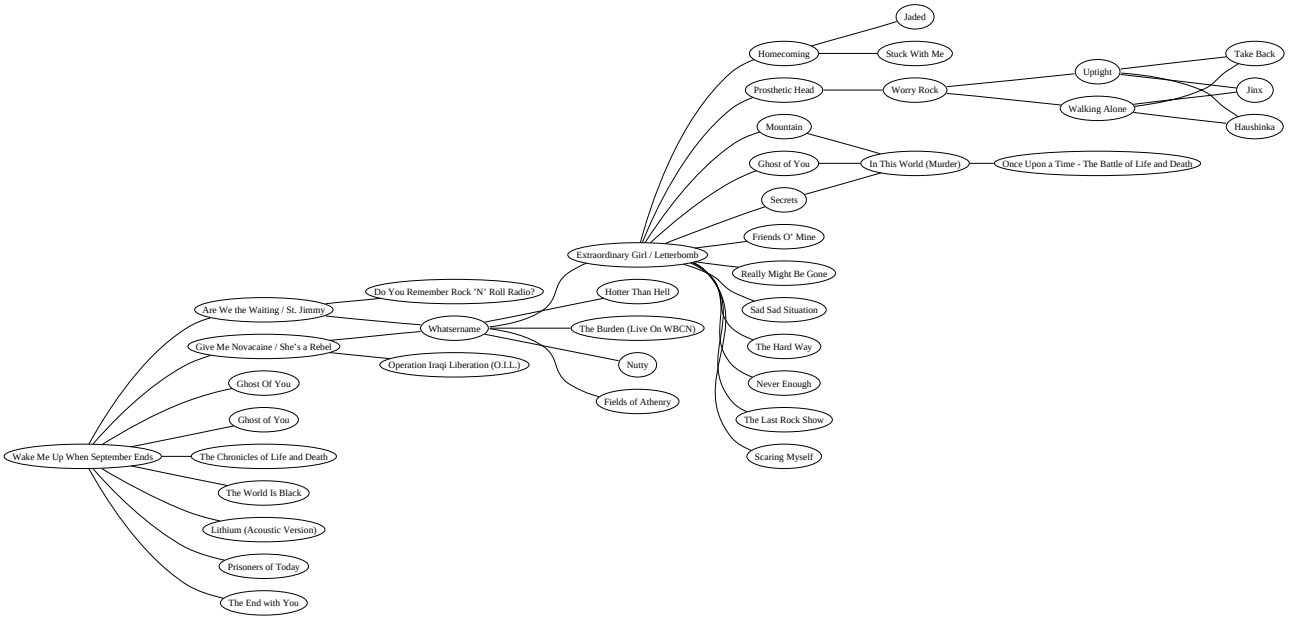

Figure 7. Popularity Diagram of "Wake Me Up When September Ends"

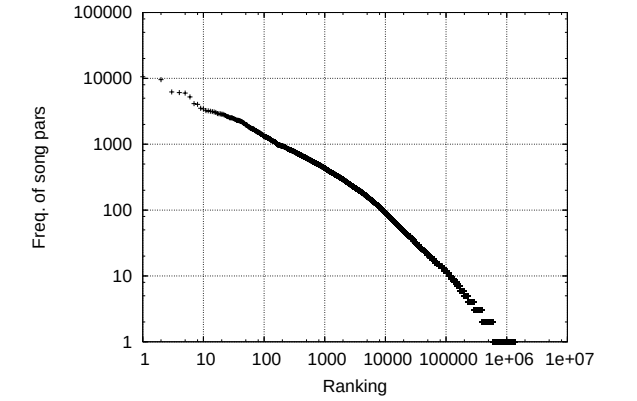

Figure 6. Number of songs pairs in PL

"Wake Me Up When September Ends".

5. Conclusion

In this paper, we described the frequency analysis of song/artist in iPod/iTunes playlists. Most of the playlists have 5 to 20 songs. The number of songs should be an important factor for estimating the relationships between songs. We also analized frequency of artist/song names in playlists, and showed that songs/artists in playlists follow Zipf's law. and throught the result of co-occurred artists data, we found analyzing 3 or more artists' co-occurrence would be more usefule than analyzing pairs only.

Frequent pattern mining is the research field to develop algorithms to find frequent patterns, such as specific items, from given massive data set. Being *frequent* is defined by a minimum support, which is given by a user of the mining algorithm in advance. This way lead the reader of this paper may expect that a frequent pattern mining algorithm is a good option for the recommendation system. However, it is difficult to decide a good minimum support, because, as we showed in this paper, the artists/songs frequency distributions follow Zipf's law, and the distributions are scale-free.

Instead, we analyzed hierarchical relation between songs according to their frequencies. It is a challenging future work to develop an algorithm that evaluates similarity of songs using such hierarchical relationship. This similarity measurement will be a core of the music recommendation system.

References

[1] W. W. Cohen and W. Fan. Web-collaborative filtering: Recommending music by crawling the web. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 33:685–698, June 2000.

[2] N. Kushmerick. Wrapper induction: Efficiency and expressiveness. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 118:15–68, 2000.

[3] T. Miura and I. Shioya. Similarity among melodies for music information retrieval. In *Proc. of CIKM*, pages 61–68, 2003.

[4] M. Schedl, P. Knees, and G. Widmer. Discovering and visualizing prototypical artists by web-based co-occurrence anal-

Rank	Genre	Freq.
1	Rock	112321
2	Alternative	60256
3	Hip-Hop/Rap	40531
4	Pop	26376
5	R&B/Soul	18644
6	Country	9506
7	Soundtrack	9047
8	Electronic	8576
9	Jazz	5774
10	Inspirational	4000
11	Vocal	3739
12	Dance	2794
13	Folk	2602
14	Comedy	2383
15	Holiday	2236
16	Latin	2088
17	Reggae	1868
18	Classical	1746
19	World	1491
20	Blues	1350
21	Disney	1022
22	New Age	991
23	Children's Music	500
24	French Pop	314
25	Spoken Word	95
26	Opera	40
27	German Pop	36
28	German Folk	18
29	Audiobooks	1

Table 3. Freqency of genres

ysis. In *Proc. of the Sixth Int'l Conf. on Music Information Retrieval (ISMIR'05)*, 2005.

[5] M. Schedl, P. Knees, and G. Widmer. A web-based approach to assessing artist similarity using co-occurrences. In *In Proc. of the Fourth Int'l Workshop on Content-Based Multimedia Indexing (CBMI'05)*, 2005.

[6] Y. Yamada, D. Ikeda, and S. Hirokawa. Automatic wrapper generation for multilingual web resources. In *LNCS 2534, Proc. 5th Int. Conf. on Discovery Science (DS2002)*, pages 332–339, 2002.

Rank	artistName	artistName	freq.
1	GreenDay	Blink-182	48368
2	Eminem	50cent	35471
3	GreenDay	Sum41	32007
4	GreenDay	GoodCharlotte	22212
5	SarahMclachlan	Barenakedladies	16565
6	Eminem	D12	12602
7	GreenDay	NewFoundGlory	11633
8	Eminem	Ludacris	10588
9	GreenDay	Nirvana	10001
10	Blink-182	sum41	9917
11	Eminem	Gunit	9634
12	Sum41	GoodCharlotte	9575
13	Korn	Slipknot	9254
14	GreenDay	Simple Plan	8872
15	Twiztid	Insane Clown Posse	8313
16	Kidz Bop Kids	William Hung	7994
17	50cent	Gunit	7717
18	sum41	NewFoundGlory	7588
19	Eminem	Nelly	7434
20	Iron Maiden	Hammerfall	7402

Table 4. Top 20 co-occurred artists