

Music Playlists via FM Radio Music Programming

Tushar Wattamwar¹, Siddhesh Pore², Prof. Amruta Patil³

Students, Department of Computer Engineering^{1,2}

Guide, Department of Computer Engineering³

Smt. Kashibai Navale College of Engineering, Vadgaon Bk, Pune, Maharashtra, India

Abstract: *The online music libraries or what we can call as music streaming services have overflowed their streaming platforms with hundreds and thousands of music playlist and music catalogs which has put the users in a struggle to find their best suitable music playlists according to their needs and their musical tastes. In this paper, we have aimed at designing a approach for generating playlists which suits the users musical taste. In this approach we analyze FM radio music programming, transform the songs into their audio features, and use that data to generate playlist for the users.*

Keywords: Playlist, Songs, FM radio, music programming, Spotify API.

I. INTRODUCTION

Until about two decades ago, people chose the music and soundtracks of their needs and various activities by purchasing various individual music or music album CDs. Eventually a few years later the world of music started shifting from these CDs to digital music devices like iPods which introduced us to the concept of Internet music distribution. With the technology rapidly advancing, there has also been an advancement in streaming technologies. With the rise of these giant music streaming platforms, we no longer have to keep buying music as a good, but just pay for the services provided on the platforms of companies like Spotify, Apple Music, Amazon and YouTube Music, etc. We pay just a few Rupees a month to one of these services and have access to a plethora of songs at our fingertips.

But however, as these music streaming platforms have been flooded with an immense catalog of music, it has become a tedious and a time-consuming task of us users to choose what to listen to. To make this easy, playlists were introduced on these platforms which are nothing but a sequence of songs for addressing the user's musical tastes. There are various approaches for doing this like selecting songs based on popularity, ratings, audio features, high-level metadata while some also analyse the data generated by social media platforms. Nowadays, user-tailored playlists have been into consideration. For example, Spotify Mega Hit Mix playlist or Dailymix playlists which reduce the time for choosing what to listen to.

But this does not take into consideration the user's have different music requirements throughout the day: usually the morning, afternoon and evening music is different from each other and so user find it odd to play those playlists. We have also observed that people like to use playlists in the form of soundtracks for their various activities. But when we look at FM radios, we can see that the songs aired or their music scheduling is never random. All that is done by music experts with deep knowledge of music and music selection (e.g., a classic song will never be played alongside rock music). It's all about the song selection and when we talk about playlists, song selection is only one of its problems. Most of the papers and studies have focused on this song selection problem with the help of recommendation systems which take the help of AI. Not only song selection, but also song ordering is equally important.

II. LITERATURE SURVEY

G. Bonnin et al. [1] have review and categorized the various approaches to automated playlist generation.

N. Lin et al. [2] have recommended music based on artist popularity with 8.33% better recommendation than Spotify radio but does not create time tailored/sensitive playlists.

A. N. Hagen [3] analyzed how users describe and make sense of practices and experiences of creating, maintaining and using personal playlists.

A. Vall [4] helps learning how people listen to music and construct playlists.

M. Pichl et al. [5] understands how users organize music nowadays by analyzing user-created playlists on Spotify, but fails to provide search facilities capable of finding music fitting a special situation.

S. Ikeda et al. [6] analyzed what kind of song sequence users feel smooth in music transition.

M. Furini et al. [7] proposes a single playlist to a user on accessing a music platform like Spotify based on listening habits like time-sensitivity.

III. RELATED WORKS

Pichl et al. [5] has analyzed the user-created playlists in order to understand their characteristics. The information about user and their playlist data has been taken from the Twitter platform and here the results suggest that almost 91% of people consider some kind of genre while creating their playlists while the other don't because diverse genres are already present in the same playlist. Hagen [3] has made observation and analyzed the user's habits of creating and maintaining playlists with the help of online reports and interviews with heavy users of Spotify. Zangerle alongside Pichl [5] has analyzed over 11,000 playlists created by the users and identifies all the possible audio features for characterizing them (*i.e., danceability, energy, acousticness, etc.*).

Flexer et al. [?] in "Playlist generation using start and end songs." In 2008 has worked on an approach for smooth transitions between the end of one song and the start of the next one based on some low-level audio features without requiring any meta data. Out of the very few, Ikeda et al. [6] has also proposed smooth transitions and playlist generation but based on acoustic audio feature. Some researchers have also approached the playlist generation based on artist similarity by gathering data of users likes to artists pages from social media platforms like Facebook.

However, this approach is different from the rest mentioned above. Our model is generating multiple time specific playlists based on the data gathered and analyzed from FM radio. This is because, the songs aired throughout the day on FM radio are not random and it's a work done by specialized people.

IV. PROBLEM STATEMENT

We implement radio music programming and generate customized playlists for users that are intended as an ordered sequence of songs associated with a specific time window with the help of Spotify API.

V. ARCHITECTURE

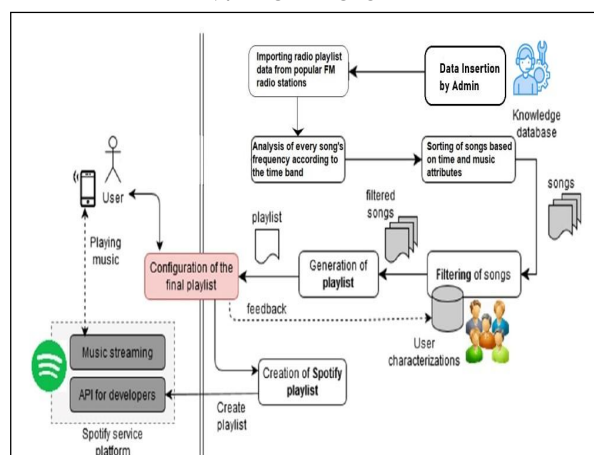


Figure 1: System Architecture.

According to our proposed and logical architecture, we can divide the system in 3 stages which essentially are, collection of songs aired on the FM radio station, analysis of these songs to transform them into their audio feature with the help of Spotify API, and then finally generating time sensitive playlists of ordered songs.

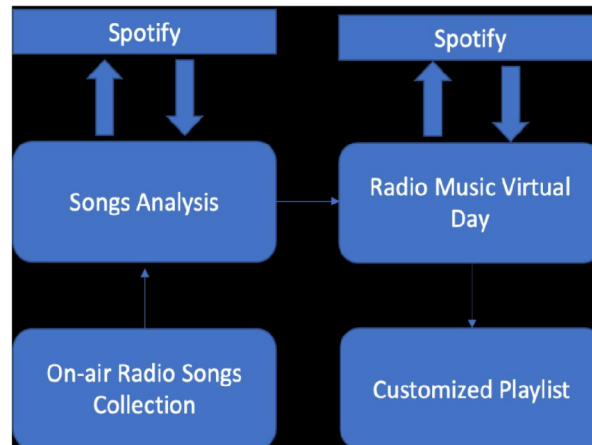


Figure 2: Logical Architecture.

We first collect the data of songs aired on the radio station for our further analysis. This data can either be collected from the web or directly from FM radio station.

In the next stage, every one of these songs is transformed into a vector of audio features (E.g., *acousticness*, *danceability*, *energy*, *instrumentalness*, *liveliness*, *loudness*, *speechiness*, *tempo*, *time_signature* and *valence*). All this is done with the help of the Spotify API which allows the transformation of a song into something like this (0,00177, 0,473, 0,881, 5,71E-05, 0,202, -4.229, 0,0744, 74.966, 4, 0,576). This transformation helps in the analysis of trends of all considered audio features.

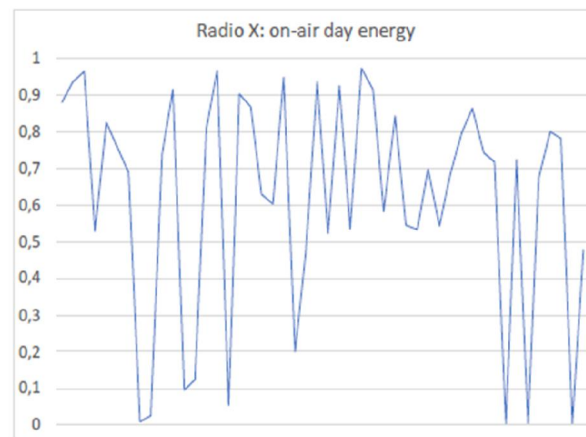


Figure 3: Energy of songs aired on Radio X in a day.

After this analysis is done, with the help of Spotify API, an array of multiple playlists for the users all specifically based on various audio features which will be specific to specific time windows throughout the entire day. We also issue a request for fetching each song that has been collected from FM radio and analyzed so as to play the songs. At any given time window when user tries to access a music from our array generated playlists, only that particular playlist specific to that time window will be displayed or made accessible to the user (E.g. at evening time playlist with high danceability/tempo/energy will be played for the user).

V. CONCLUSION

As we have seen how time-consuming and tedious it could be for user for choosing desired music or a music playlist from these music streaming services which have a plethora of songs and playlists in their catalog, we focused on an approach of producing time-sensitive playlists as ordered sequence of songs based on their audio features while relying on FM radio music programming with the help of Spotify API. We have created multiple playlists, each specific to a time window based on audio features for soothing the user's musical taste at any time of the day.

In the future, it would be interesting to create only one, time sensitive playlist instead of multiple, which can keep changing its list and order of songs according to any time of the day.

REFERENCES

- [1]. G. Bonnin and D. Jannach, "Automated generation of music playlists: Survey and experiments," *ACM Comput. Surv.*, vol. 47, no. 2, pp. 26:1–26:35, Nov. 2014.
- [2]. N. Lin, P. C. Tsai, Y. A. Chen, and H. H. Chen, "Music recommendation based on artist novelty and similarity," in *IEEE International Workshop on Multimedia Signal Processing*, Sept 2014, pp. 1–6.
- [3]. A. N. Hagen, "The playlist experience: Personal playlists in music streaming services," *Popular Music and Society*, vol. 38, no. 5, 2015.
- [4]. A. Vall, "Listener-Inspired Automated Music Playlist Generation", 9th ACM Conference on Recommender systems, Sept. 2015, p. 387-390.
- [5]. M. Pichl, E. Zangerle, and G. Specht, "Understanding playlist creation on music streaming platforms," in *2016 IEEE International Symposium on Multimedia (ISM)*, Dec 2016, pp. 475–480.
- [6]. S. Ikeda, K. Oku, and K. Kawagoe, "Analysis of music transition in acoustic feature space for music recommendation," in *International Conference on Machine Learning and Computing*, ser. ICMLC 2017, 2017, p. 77–80.
- [7]. M. Furini, J. Martini, and M. Montangero, "Automated generation of user-tailored and time-sensitive music playlists," in *IEEE Annual Consumer Communications Networking Conference (CCNC)*, Jan 2019.