ISSN: 2584-0495 Vol. 2, Issue.3, pp.658- 662

International Journal of Microsystems and IoT

ISSN: (Online) Journal homepage: https://www.ijmit.org

Music Recommender System

Shubha Vibhu Malige, Shruti J R

Cite as : Malige, S. V., & J R, S. (2024). Music Recommender System. International Journal of Microsystems and IoT, 2(3), 658–662. https://doi.org/10.5281/zenodo.11100774

Published online: 11 March 2024

DOI: https://doi.org/10.5281/zenodo.11100774

International Journal of Microsystems and IoT
Vol. 2, Issue.3, pp. 658-662; DOI: https://doi.org/10.5281/zenodo.11100774

Check for updates

Music Recommender System

Shubha Vibhu Malige[1], Shruti J R[2]

[1]Department of CSE(Cyber Security),MSRIT, Bengaluru, Karnataka, India
[2]Department of Information Science and Engineering, MSRIT, Bengaluru, Karnataka, India

ABSTRACT
Music Recommender is to assist users to filter and find out songs consistent with their choice. An ideal recommender will automatically detect preferences and accordingly generate playlists. The content based and collaborative based filtering are integrated to recommend the songs based on tracks, features, genres, and artists. The proposed work is to build a Music Recommender System which can recommend music to listeners based on users' taste and preferences. It considers into account the list of songs that users heard to or like in their playlist and interacting with devices. KMeans Clustering and BIRCH stands for Balanced Iterative Reducing and Clustering using Hierarchies, were incorporated while building Music Recommender system. The proposed work provides the recommendations based on Artist Wise, Genre Wise, and Mixed recommendations with respect to user choice and preferences. The model gives around 71% accuracy with respect to both KMeans clustering and BIRCH algorithm.

KEYWORDS
KMeans, BIRCH Clustering and Recommender System.

1. INTRODUCTION

There are millions of songs available exceeds the listening capacity of a private in their lifetime. It is tedious for an individual to sometimes choose between many songs and there is also an honest chance of missing out on songs which could are the apt for the occasion. These days all effort and research go in predicting the kind of songs that a specific user will like, using their entire song history. Here, people try to recommend songs without taking in mood as an element. A user may like party songs also as soothing songs, but recommending a song in anybody of those categories requires not just the history of the songs the user like but also capturing the present mood of the user. If an individual is within the mood to concentrate to a celebration song, he would not want the app to recommend a slow song, which statistically, consistent with his history is apt but just not suitable for the present mood. So, we will solve this problem by asking one particular question; given a song the user is currently taking note of, which song will the user want to hear next after the previous song? We try to seek out similarity within the song being played by the user with the list of songs available in our dataset by using clustering techniques. After generating a playlist of songs almost like the present song, we use popularity-based recommendation system to recommend subsequent song to the user.

In a world inundated with an ever-expanding ocean of musical choices, the role of the Music Recommender System emerges as a necessity.

Let us embark on a rhythmic journey, exploring why this digital virtuoso has become an integral part of our modern musical landscape.

The Paradox of Choice: In an era where virtually, every song ever recorded is at our fingertips, the sheer abundance of options can be overwhelming. Music Recommender Systems act as a musical compass, helping navigate this labyrinth of choices and presenting a curated selection that resonates with individual tastes. It's like having a knowledgeable friend who sifts through the noise and hands you a carefully curated bouquet of harmonies.

Discovering the Uncharted: The human tendency to gravitate towards the familiar can hinder our exploration of new melodies. Music Recommender Systems extend a friendly hand, introducing us to genres, artists, and songs that we might not have encountered on our own. It is a voyage of discovery, an avenue to broaden our auditory horizons and enrich our musical palette.

Tailored Musical Journey: Imagine embarking on a cross-country road trip without a map. Similarly, diving into the vast world of music without guidance can lead to a fragmented experience. Music Recommender Systems map out a personalized sonic journey, ensuring that each note flows seamlessly into the next, creating a cohesive and immersive musical adventure.

Time-Efficiency: Life's tempo is unrelenting, leaving us with limited time to hunt for new tunes. These systems act as efficient time-travelers, zipping through the archives and presenting us with selections that align with our preferences. They spare us the effort of scouring endless playlists, allowing us to bask in the joy of music without the chore of curation.

Unveiling the Hidden Gems: The vast expanse of the musical landscape often conceals hidden treasures – songs that may have slipped through the cracks or artists yet to gain mainstream recognition. Music Recommender Systems serve as treasure hunters, unearthing these gems and placing them gently in our auditory treasure chests.

Enhancing Emotional Connection: Music is a conduit for emotions, and the right song can uplift, comfort, or inspire. Recommender systems delve into the intricate labyrinth of musical emotions, ensuring that the notes that reach our ears resonate deeply within our hearts.

Supporting Artists and Industry: These systems are not just about us – they also play a vital role in sustaining the music industry. By connecting listeners with a diverse range of artists and genres, they contribute to the visibility and livelihood of musicians, fostering a dynamic and thriving musical ecosystem.

In a symphony of possibilities, the Music Recommender System steps in as the conductor, orchestrating a harmonious blend of familiarity and novelty. It transcends mere convenience and becomes an essential companion in our auditory exploration, enhancing our musical experience and enriching the tapestry of sounds that color our lives. As the world of music continues to evolve, the role of these systems becomes even more vital, ensuring that we never miss a chance to dance to the melody of the universe.

2. RELATED WORK

This section presents literature review on several research areas related to Music Recommender System. Music Recommender System [1] Artificial Intelligence depicts the designing and implementing a song recommendation system. To understand from the previous listening user. Music Recommendation using Deep Learnings & Collaborative Filtering [2] The approach here is to enhance the Recommender system by using the similarity content with Deep Learning. The older model and album

art of the music will be used to suggest new songs. Unique labels will be scanned for by the hybrid RS. Machine learning algorithms that are widely used are KNN model, matrix factorization similarities-based algorithm and Deep learning. The Million Song Dataset Challenge [3] A large dataset personalized music recommendation challenge, and consequences where the objectives are predicting the music that a user will listen to, with the user's previous listening list or attribute in the given playlist . Inferring User Expertise from Social Tagging in Music Recommender Systems for Streaming Services [4] In memory-based Collaborative filtering methods, the predictions for the active user are based on his/her nearest neighbors using KNN algorithm.

Chatbot with Music and Movie Recommendation Based on Mood [5] To build a model that recommends music and movies depending on the user's mood. It implements two different algorithms, i.e. Haar Cascade Algorithm for recognize the users face in each stages of the webcam feed and the Convolutional Neural Networks Algorithm to extract the facial expression to recognize the mood of the user. Music Bot Evaluating Critiquing-Based Music Recommenders with Conversational Interaction [6] To build a system for music recommendations, and it identified with two different basic critiquing techniques, user-initiated critiquing method and system-suggested critiquing. A Music Recommendation Algorithm based on Clustering and Latent Factor Model [7]

To build a music recommender system by using Kmeans clustering and latent factor model and check the comparison in terms of accuracy while recommending Music to users.

BIRCH: an efficient data clustering method for very large databases [8] A comparison of Birch algorithm with other algorithm and how it performs in terms of forming clusters and how it is better than another clustering algorithm.

3. DATASET

Kaggle is large platform to allows users to find out and bring out data sets, also will be wont to traverse and developed models in web-based datasets with same platform and is among the foremost widely used audio streaming platforms available to users. The Kaggle is one among the foremost popular websites among Data Scientists and Machine Learning Engineers. The datasets are free to do download off of Kaggle.com. Kaggle site has a wide variety of music to developers. The web API like GET PUT allows for access to user information about the music, that consist of their attributes like album artist genre. It also presents metadata regarding music artists, albums, tracks directly from the Kaggle dataset. A total of around 10,000 unique tracks from the Kaggle site and all the necessary metadata related to the track has been collected. The meta data contains a wide range of information about the track and the artist. The website also renders acoustic attribute related with the tracks like their dance ability, different kind of euphonizes for the tracks present in the catalogue. The dataset was constructed using the data fetched from the API and used to train the models.

4. DESIGN

The diagram shown in Fig 1 depicts songs commonly listened by the user are labelled based on the user preference, then various clusters are formed by using k means algorithm's songs are recommended for the user based on the collaborative filtering. The required recommendation results are presented to the user.

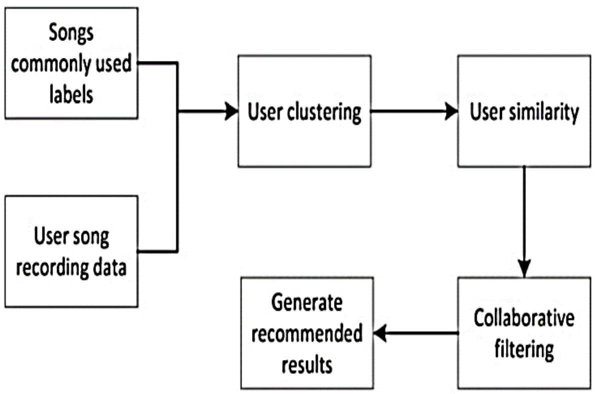

Fig. 1 Design of Music Recommender System

5. TECHNIQUES USED IN BUILDING MUSIC RECOMMENDER SYSTEM

Creating a music recommender system involves understanding user preferences and patterns in music consumption. By combining K-Means clustering and BIRCH algorithms, we can craft a powerful music recommender that captures both global and local music preferences. Here is how this fusion unfolds:

Data Collection and Preprocessing:
Gather user data, including listening histories, ratings, or interactions with songs/artists.
Preprocess the data, handling missing values, transforming categorical features (genres, artists) into numerical representations (one-hot encoding), and creating a user-item matrix.
Feature Extraction and Representation:
Utilize techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or Word2Vec to represent songs or artists as numerical vectors.
Transform user interactions into a user-item matrix where rows represent users, and columns represent songs or artists, with numerical values indicating the level of interaction or preference.
Applying BIRCH Clustering:
Use the BIRCH algorithm to cluster songs or artists based on their features or characteristics.
Create a hierarchical structure of clusters that captures different musical styles, genres, or themes.
K-Means Refinement:
Within each BIRCH cluster, apply K-Means clustering to further group similar songs or artists.
Compute cluster centroids that represent local preferences within each refined cluster.
User Profile Creation:
For each user, identify the BIRCH cluster that best matches their overall taste in music.
Within the chosen BIRCH cluster, find the nearest centroid using K-Means, representing the user's local music preferences.
Recommendation Generation:
Retrieve songs or artists that are popular or highly rated within the chosen BIRCH cluster.
Combine these recommendations with songs or artists that are favored by users who have similar local preferences (nearby K-Means centroids).
Evaluation and Optimization:
Evaluate the system's performance using metrics like precision, recall, or user satisfaction.
Fine-tune hyperparameters of BIRCH and K-Means algorithms for optimal clustering and recommendation quality.
Real-time Adaptation:
Periodically update clusters using BIRCH to adapt to evolving musical trends and user preferences.
Re-calculate K-Means centroids within each cluster to ensure up-to-date recommendations.
By intertwining the hierarchical clustering capabilities of BIRCH with the fine-grained grouping power of K-Means, this innovative music recommender system orchestrates a harmonious blend of global and local musical tastes. It not only suggests popular tracks but also surfaces hidden gems that align with users' specific preferences, resulting in a melodious journey through the vast landscape of music.

6. RECOMMENDATIONS ALGORITHMS

KMeans Clustering Algorithm

The first characteristics of clusters explains that the points inside a cluster should be similar to each other. Therefore, our goal is to reduce the distance between the points inside a cluster. The k-means clustering technique is an algorithm that tries to do this with their centroid. K-means is where the distances to assign a point to a cluster are calculated. This is because it is a centroid-based algorithm, or a distance-based algorithm. Here, each cluster is associated with a centroid. Minimizing the sum of distances between the points and their respective cluster centroid is the major goal of this algorithm.
The K-means algorithm for clustering the user preference matrix U in the previous article are as follows:
Input: user preference matrix U, number of clusters

centers k [7].
Step 1: Process the user preference matrix U and convert it into a two-dimensional matrix form Ulist that can be directly clustered.
Step 2: Initialize k empty clusters, denoted as
A {a1, a2 ak}
Step 3: Enter the two-dimensional matrix Ulist and the number of clusters centers k, and then run the K-means algorithm to get a one-dimensional list L. Each element Li in the list represents a cluster assigned to the user Ui by the clustering algorithm.
Step 4: According to the user allocation list L, each empty cluster ai is assigned a corresponding user element Uj. Finally, the cluster output is used for the next step. For project clusters, the numerical value of K is essential. This is due to the fact that when the value of K is too less, it is usually not possible to accomplish the efficient division of the project set. Also, there are fewer similar projects and noisy project elements in the cluster, which brings errors to the algorithm prediction. When the value of K is too big as opposed to the number of elements in the cluster being too small, and the item data of the scoring reference is insufficient, which in turn affects the accuracy of the algorithm. The result is produced based on the Genre Wise, Artist Wise, Mixed Wise recommendations.

BIRCH Algorithm

Nearest neighborhood method is applicable to smaller dataset and it is hard to process huge datasets with a few numbers of resources like CPU Main memory and platform environment So, the algorithms don't scale clearly in the main aspect of time period and quality as the size of the dataset surges. That's where BIRCH clustering is used for comparing the suggestion. The more popular algorithm is a Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) clustering algorithm which will cluster large datasets by first creating a small and miniscule gist of the huge dataset that retains as much information as possible. This smaller gist is then clustered instead of grouping the larger dataset. BIRCH is typically used to endorse other clustering algorithms by generating a gist of the dataset that the other clustering algorithm can now use. However, BIRCH has one prominent demerit – it can only comply with metric attributes. A metric property is any attribute whose values are frequently shown in Euclidean space i.e., no categorical attributes should be available.
BIRCH is an unsupervised literacy algorithm used to perform hierarchical clustering (8). Kmeans works best when the clusters are linearly divisible (by hyperplanes) and the clusters are compact. Whereas, hierarchical clustering styles are best suited operations where clusters are more spread out and have a tree like structure. Now, druggies belonging to this cluster may prefer pop gemstone or indie gemstone in particular and pop in general. It is also likely that since songs have multiple stripes, the druggies clustered into this cluster might like specifically faddish gemstone but because the song also belonged 5 to pop order, they were clustered with druggies who do not have similar specific tastes.

7. RESULTS

Genre Wise Recommendation

The table 1 shows the Genre wise recommendation by incorporating KMeans clustering algorithm. By considering genre Rock different album title, track-id, artist name and track title are displayed.

Table. 1 Genre wise recommendation

Track_id	Album Title	Artist name	Genre	Track Title
153	Arc and Sender	Arc and Sender	Rock	Hundred-Year Flood
154	Arc and Sender	Arc and Sender	Rock	Squares And Circles
155	unreleased demo	Arc and Sender	Rock	Maps of the Stars Homes
169	Boss of Goth	Argumentix	Rock	Boss of Goth
170	Nightmarcher	Argumentix	Rock	Industry Standard Massacre

Artist wise Recommendation

The table 2 represents Artist wise recommendation. For the particular artist's name different track title and track_id is displayed.

Table. 2 Artist wise recommendation

track id	Album Title	Artist Name	Track Title
34660	Zehu	AvantGarde|International|Blues|Jazz|	Hadri Ha'Kat
34661	Zehu	AvantGarde|International|Blues|Jazz|	Blender Tzivoni
34662	Zehu	AvantGarde|International|Blues|Jazz|	Naniah
34663	Zehu	AvantGarde|International|Blues|Jazz|	Yoter Miday

Mixed Attribute Recommendation

The table 3 represents Mixed wise recommendation. By considering all the attributes mixed feature recommendations are provided.

Table. 3 Mixed Attribute

track id	Album title	Artist Name	Genre	Track Title
6765	Cousin Mosquito 1999-2003	Dave Public	Electronic	tropical breaks
13815	Different World	Menhirs of Er Grah	Folk	No Simpler Now You're Gone
6770	Cousin Mosquito 1999-2003	Dave Public	Electronic	IMAGINE
12532	Color in a world of monochrome	Nils Hoffmann	Pop	Sweet Man Like Me
13673	The Opening Salvo	Melissa Welch	Electronic	Nano Moon Coral

8. CONCLUSION

The given model behaves overfitting because of the dataset and it can provide the limited amount of the suggestion. So, it depends on the dataset. If the dataset is of free from redundancy the same model behaves as a under fitting and the model performed well. Both K Means clustering and BIRCH algorithm provides the accuracy of about 71%. It successfully provides genre, artist, features and track based recommendations. Some proposed future enhancements include to execute the algorithms on a distributed system, to concurrently provide the computation, reduces the runtime of the recommender system. Combine all the methods and learn the for every method consistent with the dataset. Predominantly produce the suggestions based on relevant attribute.

REFERENCES

1. Shefali Garg (2014). Music Recommender System, IEEE Conference, Dept of CSE Indian Institute of Technology, Kanpur.
2. Anand Neil Arnold, and Vaira Muthu S(2019) Razzavi B. (2012), Music Recommendation using Collaborative Filtering and Deep Learning, International Journal of Innovative Technology and Exploring Engineers (IJITEE), ISSN: 2278-3075, Volume-8 Issue-7, May 2019
3. Brian McFee (2012), The Million Song Dataset Challenge, International World wide Web Conference Committee (IW3C2), SanDiego USA
4. Thomas Kugelstadt. (2005). Getting the most out of your instrumentation amplifier design, Analog Applications Journal. Texas Instruments Incorporated, Texas, USA, 25-30. https://www.ti.com/lit/pdf/slyt226
5. Diego, Sanchez-Moreno1, Inferring User Expertise from social Tagging in Music Recommender Systems for Streaming Services.
6. Shivani Shivananda1, Chatbot with Music and Movie Recommendation Based on Mood, International Journal of Engineering Research and Technology (IJERT) ISSN:2278-0181 published by NCAIT-2020 Conference Proceedings
7. Yucheng Jin Lenovo (2019), Music Bot: Evaluating Critiquing-Based Music Recommenders with Conversational Interaction research CIKM, Beijing, China
8. Yingjie Jin and Chunyan han*: A Music Recommendation Algorithm based on Clustering and Latent Factor Model. Software College, Northeaster University, Shenyang, Liaoning Province, China
9. Zhang,Tian,Raghu Ramakrishnan and Miron Livny. “BIRCH: an efficient data clustering method for very large databases, China
10. G. Fernando, Z. K.A. Baizal and R Dharayani(2021), “Music Recommendation using Conversational Recommender System with Explanation Facility” International Conference on Data Science and Its Applications, Bandung, Indonesia
11. A. Patel and R. Wadhvani, "A Comparative Study of Music Recommendation Systems," 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, 2018, pp. 1-4, doi: 10.1109/SCEECS.2018.8546852
12. V. P. Sharma, A. S. Gaded, D. Chaudhary, S. Kumar and S. Sharma, "Emotion-Based Music Recommendation System," 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2021, pp. 1-5, doi: 10.1109/ICRITO51393.2021.9596276
13. A. Nair, S. Pillai, G. S. Nair and A. T, "Emotion Based Music Playlist Recommendation System using Interactive Chatbot," 2021 6th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2021, pp. 1767-1772, doi: 10.1109/ICCES51350.2021.9489138
14. B. Kostek, "Listening to Live Music: Life Beyond Music Recommendation Systems," 2018 Joint Conference - Acoustics, Ustka, Poland, 2018, pp. 1-5, doi: 10.1109/ACOUSTICS.2018.8502385
15. H. Han, X. Luo, T. Yang and Y. Shi, "Music Recommendation Based on Feature Similarity," 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), Chongqing, China, 2018, pp. 650-654, doi: 10.1109/IICSPI.2018.8690510

Shubha Vibhu Malige received her BTech degree from VTU, Bellary, Karnataka, Bangalore in 2002 and received MTech degree from RV Engineering College, Bengaluru, Karnataka, India. She is currently pursuing her PHD from RV Engineering College, Bengaluru, Karnataka. Her areas of Interest include Computer Networking, Cyber Security, IOT and AI. She is currently working as Assistant Professor in MSRIT, Bangalore

Email: shubha.malige@msrit.edu

Shruti J R received his BTech Degree from Basaveshwar Engineering College, Bagalkot and completed her master’s degree in computer Science Engineering from Basaveshwar Engineering College, Bagalkot. Ms. Shruti is currently pursuing PHD from MSRIT, Bangalore. Her areas of interest include Block Chain Technology, ML, IOT. She is currently working as Assistant Professor in MSRIT, Bangalore, Karnataka.

Corresponding author Email: jrshruti@msrit.edu