wjert, 2024, Vol. 10, Issue 5, 336-345. *Review Article* **ISSN 2454-695X**

World Journal of Engineering Research and Technology
WJERT
www.wjert.org
SJIF Impact Factor: 7.029

MODERNIZED MUSIC RECOMMENDER APPLICATION USING MACHINE LEARNING

Hruthik G. R., *Yashas M. D., Jhenkar, Kiran Kumar N., Sindhu K. S. (Asst. Professor), Shruthi B. S. (Asst. Professor)

Information Science and Engineering, Malnad College of Engineering, Hassan, Karnataka, India.

Article Received on 21/04/2024 **Article Revised on 11/04/2024** **Article Accepted on 01/05/2024**

***Corresponding Author**
Yashas M. D.
Information Science and Engineering, Malnad College of Engineering, Hassan, Karnataka, India.

ABSTRACT

The study aims to develop an innovative music recommendation system to address specific concerns. It employs K Nearest Neighbour Classification and Random Forest Classifier algorithms. This novel approach utilizes Spotify API calls, implementing various machine learning algorithms such as K Nearest Neighbour Classification, Decision Tree Classifier, and Random Forest Classifier. The system recommends music to users based on genre, year range, and music features like acousticness, danceability, energy, tempo, and valence. The prevalence of digital music and mobile-accessible commercial music streaming services has made music more accessible. However, organizing vast digital music libraries can be time-consuming and lead to information overload. Therefore, a music recommender system that can automatically analyse music libraries and suggest suitable songs to consumers is highly beneficial. Such a system allows music providers to predict and offer relevant songs based on users' past music preferences, increasing user satisfaction and diversity in music consumption. The significance of a music recommendation system for music distributors cannot be overstated. It predicts songs that consumers would enjoy and promotes them, leading to increased user satisfaction and a wider music selection.

KEYWORDS: The system recommends music to users based on genre, year range, and music features like acousticness, danceability, energy, tempo, and valence.

I. INTRODUCTION

With the explosion of networks in recent decades, the internet has become the major source for retrieving multimedia information such as videos, books, and music. Music is considered an important aspect of people's lives, but they engage in listening to music infrequently.

Music service providers need an efficient way to manage songs and help their customers discover music by providing quality recommendations. A music recommender system learns from the user's past listening history and recommends songs they would likely enjoy in the future. By utilizing a music recommender system, providers can predict and offer appropriate songs to users based on the characteristics of the music they have previously heard.

There is a strong need for a good recommendation system, especially with the prevalence of music streaming services like Pandora, Spotify, etc., working on building high- precision commercial music recommendation systems. Amazon, Netflix, and many other companies are also using recommendation systems to enhance user experience.

Music recommendation systems have revolutionized how we discover and enjoy music in the digital era. Leveraging machine learning algorithms, these systems provide personalized recommendations, enhancing user experience on music streaming platforms. By analyzing user behaviour and preferences, they generate tailored suggestions that align with individual tastes, allowing users to explore diverse genres and artists. These systems address the challenge of information overload by curating personalized playlists and suggesting songs based on users' unique preferences, thus streamlining the music discovery process. Through continuous learning from user interactions, music recommendation systems improve the accuracy and relevance of suggestions over time, contributing to a more engaging and enjoyable music listening experience.

1. Collaborative approach

This type of suggestion is based on an analysis of both the behaviour of the audience and the behaviour of all other platform users. The fundamental concept is that the opinions of other users can be used to make a reliable prediction about another user's preferences for an item they have not yet rated: a user is provided recommendations based on other users who share their tastes. Indeed, for years, individuals have sought recommendations from friends, family, and colleagues when it came to music, restaurants, movies, and other forms of entertainment. This is the mechanism being attempted to be replicated here. This strategy (based on stars given by other users) was pioneered by Netflix, but it is now widely used, notably for Spotify's Discover Weekly.

Collaborative filtering makes suggestions based on the collective evaluation power of users. It is assumed that if various users rate music items similarly or exhibit similar behaviour, they would rate other music items similarly. The main issue in collaborative filtering methods is the sparse evaluation matrix, as most users only encounter a small portion of all music libraries, resulting in many evaluations being undecided. When a new user is added to the system, they will not initially have enough ratings for the system to find sufficiently similar users, thus limiting the accuracy of the predictions.

2. Content Based approach

The content-based recommendation method focuses on analysing the content of items to suggest similar goods based on user preferences. It doesn't rely on user feedback, instead using sound similarity from previously heard songs. Items are located based on their characteristics, and suggestions are made by comparing them to other items in the dataset. Content-based filtering can suggest songs similar to those in a playlist but not on it, using properties like loudness and pace.

However, content-based filtering's drawback is its predictability, suggesting only similar songs and potentially disappointing new users with limited listening history. Yet, diverse information can lead to surprising connections between seemingly dissimilar songs. This method extracts audio elements and user preferences, using features like timbre and rhythm for similarity.

Spotify's rich track information helps compare songs. Despite limited user data, features like volume, mode, and tempo are used, although their effectiveness for describing recordings is uncertain. Content-based filtering groups items based on similar attributes and uses the user's history to recommend items with high resemblance. This approach allows for recommendations of both popular and unknown music, avoiding the cold start problem of new items in the system.

3. Context-based approach

The recommendation of music is influenced by various factors such as the time, emotional state, and specific conditions under which the music is listened to. These factors greatly impact how individuals perceive and enjoy music. While there are numerous applications for music recommendation, few examples exist in reality, such as tourist guide apps with adaptive ambient music.

The fundamental principle behind music recommendation is leveraging the opinions of other users to predict a user's preferences for items they have not yet rated. This collaborative and content- based screening generates personalized suggestions based on similarities in audio signals. The goal is to create a framework that helps users discover music that suits their tastes.

4. Hybrid approach

Developing a hybrid recommendation system involves combining complementary strategies such as collaborative and content-based filtering. This approach can address issues like cold start and sparsity by leveraging less common techniques like location-based recommendations. By combining multiple recommendation systems into one, a hybrid method can evaluate more criteria for each recommendation. Additionally, demographic filtering, a form of collaborative filtering, can group people with similar demographics together for better recommendations.

In our recommender system, we use both content- based and real-time data, making it a hybrid method. While real-time data doesn't directly solve content- based filtering challenges, it improves suggestions by customizing them based on the user's current condition. It's also possible to maintain distinct systems and assign weights to them, allowing for easy switching between systems. Moreover, outputs from one system can be used as input for another system, enhancing the overall recommendation process.

II. LITERATURE SURVEY

The scrutiny of a Music Suggestion Framework encompasses assessing the efficacy of the developed model and acquiring insights into its functionality. Here is a general overview of how the evaluation is typically executed. The majority of prior music recommendation frameworks extract characteristics from temporal correlations among consecutive listening histories, overlooking the exploitation of supplementary data, such as the singer and album of the music. In this manuscript, we concentrate on the music sequential recommendation task while taking into account the additional data, proposing a pioneering Graph- based

Attentive Sequential Model with Metadata (GASM). This model integrates metadata to enhance music depictions and proficiently unearth user listening behaviour patterns. The incorporation of metadata with a high frequency of omissions in GASM may lead to unintended consequences, highlighting the urgent requirement for a viable approach to absorb metadata adaptively and selectively. The application of GASM in other domains with copious metadata remains unexplored.[2]

This work introduces a deep music recommendation algorithm that utilizes dance motion analysis and evaluates it quantitatively. For this purpose, the algorithm employs an LSTM-AE based method to learn the connections between motion and music. The proposed approach achieves a recommendation accuracy of 91.3% by late fusion of joint and limb features. The main challenge addressed is the scarcity of deep learning-based music recommendation methods that incorporate motion analysis. Additionally, this work pioneers the use of quantitative measurement for evaluating both motion synthesis and music recommendation, focusing on the overall alignment between the music track and the entire dance motion sequences.[3]

This study aims to create a music recommendation framework based on audio signal features' similarity. It employs Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) for this purpose. The framework intends to provide customized recommendations that accurately reflect individual preferences. While the precision of the model after 10 epochs may be limited, further training can significantly enhance its performance. The focus remains on refining the model through training to enhance the accuracy of its predictions.[4]

This project utilizes a dataset of songs to establish user-song correlations, enabling the recommendation of new songs based on users' past preferences. It is implemented using libraries like NumPy and Pandas, leveraging techniques such as Cosine similarity and Count Vectorizer. The project includes a front end with Flask to display recommended songs based on a specific song selection. Designing a personalized music recommender is complex, as understanding and meeting users' needs is challenging. Future music recommenders should guide users to make informed music choices.[1]

The study aims to develop a music recommendation system using machine learning techniques such as K Nearest Neighbour Classification and Random Forest Classifier. It also proposes an alternative approach utilizing Spotify API calls and various machine learning algorithms, including Decision Tree Classifier. The system recommends music to users based on factors like genre, year range, and music features such as acousticness, danceability, energy, tempo, and valence.[5]

Digital music has become highly accessible through commercial streaming services on mobile devices. Managing vast music libraries can be overwhelming, leading to the need for music recommender systems. These systems can scan libraries and suggest songs to users based on their listening history, improving user satisfaction and diversifying music consumption. Music providers benefit from these systems by anticipating user preferences and promoting relevant songs, ultimately increasing user satisfaction and offering a wider range of music choices.[6]

III.PROPOSED SYSTEM

This research aims to determine the most effective data mining (DM) technique with the highest precision among various classification techniques. Additionally, it seeks to evaluate the impact of the train/test data ratio on the prediction accuracy. All paragraphs must be indented. All paragraphs must be justified, i.e. both left-justified and right-justified.

1. Data Collection and Understanding Process

The authentic dataset has been utilized for the research. We've acquired musical data comprising 2000 records and 15 attributes, encompassing categorical and numerical features. Each record in the musical dataset signifies singular musical information, with each attribute in the record representing a feature of that specific entity.

2. Data Preparation and Pre-processing

After the data collection process is finished, the process of preparing the data is performed. It is important to refine this data so that it can be suitable for the models and generate better results. In this phase, we performed tasks like cleaning, filling the missing data, and removing unwanted data. The data of Spotify had various attributes which were not relevant, i.e., was not giving any useful information, like Title, Artist, Top Genre, Energy, BPM, Liveness, etc.; hence these attributes are removed in this phase.

3. Feature Selection

Feature selection stands as a pivotal concept within the realms of Data Mining (DM) and

Machine Learning. This process involves the meticulous selection of pertinent variables from a dataset to enhance the outcomes of machine learning algorithms, thereby improving accuracy. Given the multitude of columns present in the predictor variable, calculating the correlation coefficient becomes imperative to identify the significant ones. These selected variables are then utilized in the training phase to extract the key factors influencing performance.

4. Test and Train Dataset

The test dataset contains all the necessary data for data prediction, while the training dataset contains all irrelevant data. This study splits the dataset into variable ratios to analyse prediction estimation. The main focus is on identifying the most important variables that could positively impact the accuracy of features in music performance prediction models, utilizing various feature selection algorithms.

IV. MODELING AND EXPERIMENTS

Prior to constructing the model and software framework, the initial phase involves data preprocessing and cleaning. This preparatory stage is crucial as the 'get important features' function relies on complete rows; any missing data in the dataset, indicated by NaN values, must be substituted with empty strings.

We can see the most important features selected in Table.

Sr.no	Attributes
1	Song Name
2	Singer/Artists
3	Genre
4	Album/Movie
5	User Rating

Here, three machine learning models are being used for the recommendation of music.

K-Nearest Neighbour

The k-Nearest-Neighbours (kNN) method is a simple yet effective classification strategy. However, it has limitations, including low efficiency due to its lazy learning nature, which makes it unsuitable for large repositories in dynamic web mining. Another drawback is its reliance on choosing an appropriate value for k. To address these issues, a novel kNN-based classification technique is proposed in this paper.

The method builds a kNN model directly from the data, reducing the reliance on k. The ideal number of k is automatically chosen based on the size of the genre data, enhancing classification accuracy and speed. The nearest neighbours k value is determined by the length of the genre_data being chosen, allowing for the selection of the nearest neighbours based on this value. The recommendation of the first was implemented using the k Nearest Neighbour Classification model.

Decision Tree Classifier

Decision Trees have surfaced as a model- centric approach for crafting recommender systems. Leveraging decision trees to construct recommendation models proffers myriad advantages, encompassing efficiency, interpretability, and adaptability in managing diverse input data formats. This model constructs a prognostic schema that links the input to a projected value grounded on the input's attributes. Each inner node within the tree corresponds to an attribute, while each linkage from a progenitor to a descendant node epitomizes a potential value or array of values of that attribute.

The process of tree formation initiates with a root node and the input set. An attribute is designated to the root, and linkages and sub-nodes are engendered for each array of values. The input set is subsequently partitioned by the values, guaranteeing that each descendant node solely acquires the portion of the input set congruent with the attribute value stipulated by the linkage to the descendant node. This process is then reiterated recursively for each descendant until further partitioning becomes unviable. Hence, the decision tree classifier is a pivotal instrument for crafting recommender systems. This model is operationalized for augmenting Spotify account page recommendations.

Random Forest Classifier

Random forest is indeed a highly accurate method suitable for large datasets. It can predict missing data with high precision, even without preprocessing, making it efficient. This technique combines bagging and random feature selection to create decision trees, which are then combined with individual learners. By using a random subset of training data, these trees are generated. Once the forest is trained, test rows are passed through it, and each tree produces an output class. The random forest classifier in a music recommendation system determines the output class by taking the mode of these classes. Because it's an ensemble learning technique, the random forest classifier is widely used in machine learning for its high accuracy and performance in recommendations.

V. REQUIREMENT ANALYSIS

Software: The operating systems used will be windows 7 & above. Programming languages used are Python, Java, Flutter, Bootstrap.

Hardware: To ensure optimal system performance, the primary memory should exceed 8 GB, allowing the entire program to reside in memory simultaneously. This eliminates the need to swap memory contents, enhancing operational efficiency. Additionally, a hard disk drive is necessary for permanent program storage, while a high-speed processor is essential for swift data processing. A Computer/Laptop is indispensable for enabling user interaction with the system while on the move.

The system begins by accepting a user- inputted song. Following this, ten similar songs are recommended based on five main features: Song name, Artist, Album/Movie, Genre, and User Rating. The system utilizes Angular distance and Euclidean distance for this process. To implement this, the Count Vectorizer class and cosine similarity method are employed. The Count Vectorizer class counts the number of terms in each feature, while cosine similarity calculates the similarity score using structured data. Prior to processing with the Count Vectorizer class, a function is utilized to merge the contents of all rows of the specified features. Any NaN values are replaced with an empty string to ensure smooth processing.

VI. CONCLUSION AND FUTURE SCOPE

Designing a personalized music recommender is a complex task, requiring a deep understanding of users' preferences and needs. As we look to future research, the focus will be on developing user-centric music recommender systems. A survey conducted among athletes revealed that individuals in sport and exercise environments often choose music somewhat arbitrarily, without fully considering its motivational characteristics. Therefore, future music recommendation systems should guide users towards making more informed choices. Ultimately, we hope that this study will help bridge the gap between isolated research in various disciplines.

VII. REFERENCES

1. Music Recommendation System Using Machine Learning Varsha Verma, Ninad Marathe, Parth Sanghavi, Dr. Prashant Nitnaware International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2021.
2. Graph-Based Attentive Sequential Model With Metadata for Music Recommendation HE

WENG, JIANJIANG CHEN, DONGJING WANG, (Member, IEEE), XIN ZHANG, AND DONGJIN YU, (Senior Member, IEEE) IEEE Journal-12 October 2022.

3. A Deep Music Recommendation Method Based on Human Motion Analysis WENJUAN GONG, (Member, IEEE), AND QINGSHUANG YU IEEE Journal- February 5, 2021.
4. AI based Music Recommendation system using Deep Learning Algorithms Anand R, Sabeenian R.S, Deepika Gurang, Kirthika R, Shaik Rubeena R Anand et al 2021 IOP Conf. S.
5. Music Recommendation System Using Machine Learning Dr. Sanjay Gandhi Gundabatini, R. Bindu Sneha, M. Vamsi, N. Yagna Yaswanth, K. Krishna Vamsi Issue 04, Apr 2023.
6. RECOMMENDATION SYSTEM FOR MUSIC BASED ON CONTENT AND POPULARITY RATINGS Timanshi Bhardwaj Aastha Jain Karan Choudhary Manav Rachna International Institute of Research and Studies, Faridabad, India, December 2021.
7. Music Content Recognition and Recommendation Technology Based on Deep Learning Gao Yang Qilu Normal University, Shandong published on 14 march 2022.